
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

# ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

## ’’Αποτελεσματική μείωση δεδομένων εκπαίδευσης με διαχωρισμό του χώρου των δεδομένων και χρήση των κεντροειδών κλάσεων’’

**Του φοιτητή**
**Μαστρομανώλη Θεόδωρου**
**Αρ. Μητρώου: 174937**

**Επιβλέπων**
**Ουγιάρογλου Στέφανος, Επ. Καθηγητής**
**Δέρβος Δημήτρης, Καθηγητής**

2 Οκτωβρίου 2021

Τίτλος Δ.Ε.: Αποτελεσματική μείωση δεδομένων εκπαίδευσης με διαχωρισμό του χώρου των δεδομένων και χρήση των κεντροειδών κλάσεων

Κωδικός Δ.Ε. 21163

Ονοματεπώνυμο φοιτητή/ών: Μαστρομανώλης Θεόδωρος

Ονοματεπώνυμο εισηγητή: Ουγιάρογλου Στέφανος

Ημερομηνία ανάληψης Δ.Ε.: 15-03-2021

Ημερομηνία περάτωσης Δ.Ε.: 14-09-2021

*Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.*



Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραιτήτως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

## Πρόλογος

Η εργασία η οποία πρόκειται να διαβάσετε, συντάχθηκε από τον φοιτητή Μαστρομανώλη Θεόδωρο υπό την εποπτεία του κ. Στέφανου Ουγιάρογλου. Αφόρα τον τομέα της εξόρυξης και της αναλυτικής των δεδομένων και το συγκεκριμένο γνωστικό αντικείμενο διδάσκεται στο Διεθνές Πανεπιστήμιο της Ελλάδος, στο τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων, στα πλαίσια του μαθήματος "Οργάνωση Δεδομένων και Εξόρυξη Πληροφορίας". Το συγκεκριμένο μάθημα, διδάσκει αρκετά από τα πράγματα και τις έννοιες τις οποίες αναφέρουμε σε αυτή την εργασία. Στόχος μας, είναι να προσθέσουμε ένα "λυθαράκι" σε αυτή την γνωστική περιοχή. Το συγκεκριμένο μάθημα, είναι ιδιαίτερα ενδιαφέρον και ελκυστικό για πολλούς φοιτητές, καθώς μας μαθαίνει έννοιες, αλγορίθμους και τεχνικές που μπορούν να χρησιμοποιηθούν σε πραγματικά δεδομένα. Τα στοιχεία αυτά εξετάζονται τόσο θεωρητικά, όσο και πρακτικά. Αυτά, δεν είναι άλλα από τεχνικές μείωσης δεδομένων, αλγορίθμους κατηγοριοποίησης, συσταδοποίηση και ομαδοποίηση στιγμιοτύπων κ.α. Η επιστήμη της εξόρυξης δεδομένων, έχει προσφέρει πάρα πολλά πράγματα σε διάφορους τομείς όπως η Ιατρική, η Διαφήμιση και το Μαρκετινγκ, η Πληροφορική, η Βιολογία κ.α. Τελειώνοντας, αν και υπάρχει ήδη πληθώρα αλγορίθμων κατηγοριοποίησης και μείωσης συνόλων δεδομένων, θα ήθελα και εγώ με τη σειρά μου μαζί με τον κ. Ουγιάρογλου να συνεισφέρουμε σε αυτό το έργο μέσω αυτής της διπλωματικής εργασίας.

## Περίληψη

Η συγκεκριμένη εργασία, δίνει έμφαση στους αλγόριθμους μείωσης δεδομένων και τους αλγορίθμους κατηγοριοποίησης στιγμιοτύπων. Στο παρελθόν, έχουν προταθεί διάφοροι αλγόριθμοι μείωσης δεδομένων οι οποίοι αν και αποδείχτηκαν αποτελεσματικοί, φαίνεται να έχουν περιθώρια για βελτίωση. Στόχος αυτών των αλγορίθμων, είναι η δημιουργία ενός σύνολου δεδομένων το οποίο είναι πολύ μικρότερο από το αρχικό, ενώ τα αποτελέσματα της κατηγοριοποίησης επηρεάζονται όσο το δυνατόν λιγότερο γίνεται. Αυτό το σύνολο δεδομένων, ονομάζεται συμπυκνωμένο σύνολο δεδομένων και τα πλεονεκτήματά του είναι το μικρό υπολογιστικό κόστος που απαιτείται για την εφαρμογή αλγορίθμων σε αυτό, ενώ ταυτόχρονα η ακρίβεια της κατηγοριοποίησης μένει σε υψηλά επίπεδα. Συχνά, για αυτή τη διαδικασία χρησιμοποιείται η μέθοδος της μείωσης των δεδομένων με διαχωρισμό του χώρου και ειδικότερα ο αλγόριθμος RSP3. Βασικό πρόβλημα του εν λόγω αλγορίθμου είναι το μεγάλο υπολογιστικό κόσοτος που απαιτεί για τη δημιουργία του συμπυκνωμένου συνόλου. Στη συγκεκριμένη εργασία, προτείνονται παραλλαγές του RSP3, οι οποίες έχουν ως σκοπό να μειώσουν το υπολογιστικό κόστος δημιουργίας του συμπυκνωμένου συνόλου καθώς επίσης να βελτιώσουν και άλλες μετρικές που παίρνουμε κατά τη πειραματική διαδικασία. Έτσι, στην παρούσα εργασία, αφού παρουσιαστούν κάποιες βασικές έννοιες σχετικά με την κατηγοριοποίηση, τον αλγόριθμο των κ εγγύτερων γειτόνων, τις τεχνικές μείωσης των δεδομένων και αφού περιγραφούν οι υπάρχοντες αλγόριθμοι μείωσης δεδομένων βάσει διαχωρισμού του χώρου των δεδομένων και τα προβλήματα τους, θα προταθούν παράλλαγες του αλγορίθμου RSP3, που έχουν ως στόχο την αντιμετώπιση των προβλημάτων που παρουσιάζουν οι υπάρχοντες αλγόριθμοι. Στο τέλος, η παρούσα διπλωματική εργασία παρουσιάζει μια εκτεταμένη πειραματική μελέτη, όπου οι προτεινόμενοι αλγόριθμοι συγκρίνονται τόσο μεταξύ τους όσο και με τους υπάρχοντες αλγορίθμους μείωσης δεδομένων εκτελώντας πειράματα σε έναν μεγάλο αριθμό συνόλων δεδομένων. Τα απότελέσματα της πειραματικής μελέτης αναδεικνυουν την υψηλή απόδοση και αποτελεσματικότητα των προτεινόμενων παραλλάγων.

«Data Reduction by Space Partitioning through the usage of Class Centroids»

«Theodoros Mastromanolis»

## Abstract

The focus of this paper, is Data Reduction methods and Classification Algorithms. In the past, there have been suggested algorithms for these causes, which even though were effective, there seem to have room for improvement. The aim of these algorithms, is the creation of a condensing set, a dataset which is much smaller in space than the starting one, while the results of the classification of the data are not influenced a lot in a negative way. This condensed dataset, needs a lot less computing data than initial dataset, while the classification accuracy stays at high levels. Often, the RSP3 algorithm is used for the process of data reduction through space partitioning.A drawback of this algorithm is the big computational cost needed for the creation of the condensed set. In this paper also, there are some alternations of the RSP3 that are being suggested, with the aim to improve the computational cost needed for the creation of the condensed set, while also improving the other metrics at the experimental stages. In this particular paper, after some basic ideas about classification, the k-NN algorithm, data reduction methods with their problems, and after the currently existing data reduction by space partitioning algorithms and their problems are presented, there will be some alternations of the RSP3 algorithm suggested, with the aim to deal with the drawbacks of the existing algorithms. Last but not least, this paper presents an extended experimental study, in which the suggested algorithms are compared to each other, while also compared to the existing algorithms by testing and experimenting with a great amount of datasets. The results of this experimental study, highlight the great performance of the suggested alternative algorithms.

## Ευχαριστίες

Ξεκινώντας, θα ήθελα να ευχαριστήσω τον κ. Στέφανο Ουγιάρογλου, πρώην μέλος ΕΔΙΠ του τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων και νυν Επίκουρο καθηγητή του Πανεπιστημίου Πελοποννήσου, ο οποίος είναι ο επιβλέπων καθηγητής της συγκεκριμένης διπλωματικής εργασίας. Με τη καθοδήγησή του και τη συνεχή προσοχή που μου προσέφερε, με βοήθησε να ξεπεράσω οποιαδήποτε θέματα είχαν προκύψει είτε κατά τη συγγραφή της εργασίας, είτε κατά την ανάπτυξη των αλγορίθμων που εμπεριέχονται σε αυτή. Τον ευχαριστώ για τον χρόνο τον οποίο διέθεσε και θα ήθελα να τονίσω πως κατά τη συννενόηση και την επεξήγηση διαφόρων θεμάτων που έιχα ήταν πλήρως κατανοητός. Έπειτα, θα ήθελα να ευχαριστήσω την οικογένειά μου, τους φίλους μου και τους δικούς μου ανθρώπους, καθώς μου στάθηκαν και κατανόησαν από τη πρώτη στιγμή το δύσκολο έργο που είχα να φέρω εις πέρας, δηλαδή την εκπόνηση της συγκεκριμένης διπλωματικής εργασίας. Χωρίς τη ψυχολογική στήριξη που μου προσέφεραν δε θα μπορούσα σε καμία περίπτωση να τα καταφέρω, και τους ευχαριστώ για αυτόν τον λόγο.

# Περιεχόμενα

## Κατάλογος Σχημάτων



## Κατάλογος Πινάκων

# Κεφάλαιο 1ο: Εισαγωγή

## 1.1 Κατηγοριοποίηση

Οι αλγόριθμοι εξόρυξης δεδομένων, όσον αφορά την αποτελεσματικότητα και την αποδοτικότητα τους, αποτελούν ένα σημαντικό επιστημονικό θέμα το οποίο έχει τραβήξει τόσο τη προσοχή των ακαδημαικών, όσο και των επαγγελματιών της πληροφορικής [1]. Η κατηγοριοποίηση των δεδομένων, αποτελεί ένα βασικό θέμα στην εξόρυξη δεδομένων. Η κατηγοριοποίηση, είναι η διαδικασία διαχωρισμού των δεδομένων σε μικρότερες ομάδες, βάση κάποιων χαρακτηριστικών που έχουν. Αυτό, είναι κάτι το οποίο γίνεται συνεχώς στον κόσμο μας, αν και πολλές φορές δε μπορούμε να το αντιληφθούμε. Για παράδειγμα, η έγκριση ενός δανείου ή η πρόβλεψη μιας οικονομικής απάτης βάση κριτηρίων που θέτουν οι τράπεζες, αποτελούν παραδείγματα κατηγοριοποίησης. Τέτοια παραδείγματα μπορούν να βρεθούν σε διάφορους τομείς πέρα από την οικονομία, όπως η ιατρική, η πληροφορική και το περιβάλλον.

Οι κατηγοριοποιητές, μπορούν να χωριστούν σε δύο κύριες κατηγορίες αλγορίθμων [2]. Στους πρόθυμους ή επαγωγικούς (eager) κατηγοριοποιητές και τους οκνυρούς (lazy) κατηγοριοποιητές. Σκοπός και των δύο είναι να η ακριβής πρόβλεψη της εκάστοτε κλάσης για ένα συγκεκριμένο στιγμιότυπο. Ο τρόπος, όμως, με τον οποίο το επιτυγχάνουν αυτό είναι διαφορετικός. Ένας πρόθυμος κατηγοριοποιητής προ-επεξεργάζεται τα δεδομένα και δημιουργεί ένα μοντέλο, το οποίο χρησιμοποιείται στη συνέχεια για τη κατηγοριοποίηση των νέων δεδομένων. Ένας οκνυρός κατηγοριοποιητής, δεν θα δημιουργήσει ένα νέο μοντέλο κατηγοριοποίησης, αλλά θα θεωρήσει πως το σύνολο δεδομένων της εκπαίδευσης είναι ένα μοντέλο κατηγοριοποίησης. Έτσι, θα κατηγοριοποιήσει ένα νέο στιγμιότυπο βάση του συνόλου εκπαίδευσης.

Συγκρίνοντας αυτές τις δύο κατηγορίες αλγορίθμων, ερχόμαστε στο συμπέρασμα ότι η κατηγοριοποίηση στους πρόθυμους κατηγοριοποιητές είναι πολύ γρηγορότερη σε σχέση με τους οκνυρούς, καθώς οι πρώτοι έχουν ήδη ένα μοντέλο κατηγοριοποίησης έτοιμο πριν φτάσει το στιγμιότυπο το οποίο θα κατηγοριοποιηθεί. Οι οκνυροί κατηγοριοποιητές, χρειάζονται πολύ χρόνο για τη κατηγοριοποίηση στιγμιοτύπων σε σχέση με τους πρόθυμους κατηγοριοποιητές καθώς χρησιμοποιούν για μοντέλο κατηγοριοποίησης όλο το σύνολο των δεδομένων εκπαίδευσης, γεγονός το οποίο αποτελεί μια πολύ χρονοβόρα και περίπλοκη διαδικασία για έναν αλγόριθμο.

Ένα βασικό μειονέκτημα των πρόθυμων κατηγοριοποιητών, είναι ότι βάσει της προ-επεξεργασίας των δεδομένων που κάνουν βγάζουν συμπεράσματα και κρατάνε συγκεκριμένα στοιχεία από ολόκληρο το σύνολο δεδομένων εκπαίδευσης. Αυτό, δεν είναι πάντα εφικτό να γίνεται, καθώς μπορεί πολλές φορές να επιδράσει αρνητικά στην ακρίβεια της πρόβλεψης, η οποία αποτελεί μια πολύ σημαντική μετρική για την εξόρυξη δεδομένων, καθώς και να αυξήσει τον χρόνο και τη πολληπλοκότητα που απαιτείται για τη προ-επεξεργασία των δεδομένων. Από την άλλη, οι οκνυροί κατηγοριοποιητές επειδή χρησιμοποιούν ολόκληρο το σύνολο των δεδομένων, μπορούν να κάνουν πιο περίπλοκες υποθέσεις για τα δεδομένα. Ως αποτέλεσμα αυτού, αυξάνεται και η ακρίβεια της πρόβλεψης. Μια τελευταία διαφορά που έχουν οι πρόθυμοι και οι οκνυροί κατηγοριοποιητές, είναι πως οι οκνυροί κατηγοριοποιητές χρειάζονται πολύ χώρο στη μνήμη, καθώς πρέπει όλο το σύνολο δεδομένων να υπάρχει αποθηκευμένο για να λειτουγήσει ο αλγόριθμος, ενώ οι πρόθυμοι αλγόριθμοι αφότου φτιάξουν το μοντέλο πρόβλεψης, μπορούν να ξεφορτωθούν από τη μνήμη όλο το σύνολο των δεδομένων και να κρατήσουν μόνο το μοντέλο που έχουν

δημιουργήσει.

Κάποια παραδείγματα πρόθυμων αλγορίθμων είναι τα δέντρα αποφάσεων, ο Naive Bayes και τα νευρωνικά δίκτυα, ενώ παραδείγματα οκνυρών αλγορίθμων αποτελούν ο αλγόριθμος των k-εγγύτερων γειτόνων και ο case-based reasoning.

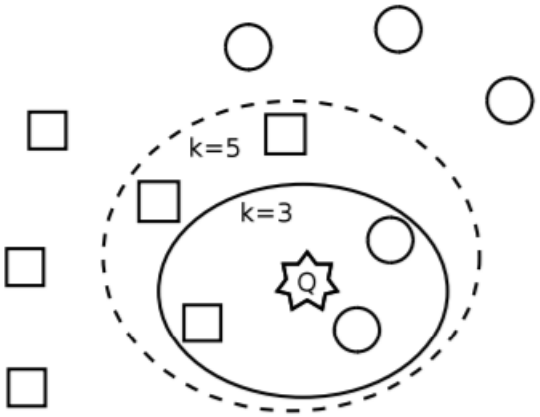
Τα δέντρα αποφάσεων [3], αποτελούν μια πολύ γνωστή υποκατηγορία των πρόθυμων αλγορίθμων. Μέσω των δέντρων αυτών, μοντελοποιείται μια σειριακή και ιεραρχική δομή αποφάσεων, η οποία μας οδηγεί σε ένα τελικό αποτέλεσμα. Η πολυπλοκότητα ενός δέντρου αποφάσεων σχετίζεται άμεσα με τις επιλογές που μπορούμε να πάρουμε σαν αποτελέσματα. Για παράδειγμα, αν θέλουμε να προβλέψουμε τον καιρό και δώσουμε σαν επιλογές στο δέντρο μία προς μία τις πιθανές τιμές που μπορεί να πάρει η θερμοκρασία, ή του δώσουμε μία προς μία τις τιμές που μπορεί να πάρει ο βαθμός της υγρασίας, τότε πιθανότατα το δέντρο μας να γίνει πολύ μεγάλο. Πρακτικά, δε μας ενδιαφέρει απαραίτητα η ακριβής τιμή της θερμοκρασίας και της υγρασίας, απλά μας ενδιαφέρει το αν μπορούμε να είμαστε έξω ή όχι σε μια συγκεκριμένη στιγμή της ημέρας. Η τελική ιδέα ενός δέντρου [4], είναι ο διαχωρισμός των δεδομένων σε περιοχές στις οποίες θα υπάρχουν μόνο ομογενή στιγμιότυπα. Ομοιογενής, είναι η περιοχή στην οποία τα στιγμιότυπα τα οποία βρίσκονται μέσα ανήκουν στην ίδια κλάση. Άλλοι πρόθυμοι κατηγοριοποιητές είναι τα νευρωνικά δίκτυα [5], τα οποία έχουν ως σκοπό να απεικονίσουν τον τρόπο λειτουργίας του ανθρώπινου εγκεφάλου. Ένα νευρωνικό δίκτυο πρώτα εκπαιδεύεται και στη συνέχεια κατηγοριοποιεί τα στιγμιότυπα, όπως αυτό κρίνει ότι πρέπει να κατηγοριοποιηθούν. Επιπλέον, υπάρχουν και οι πιθανοτικοί κατηγοριοποιητές, οι οποίοι χτίζουν ένα μοντέλο βασισμένο στις πιθανότητες. Ένα χαρακτηριστικό παράδειγμα πιθανοτικού κατηγοριοποιητή, αποτελεί ο Naive Bayes [6]. Ο Naive Bayes αλγόριθμος, είναι πολύ γνωστός και ευραίως χρησιμοποιημένος στην επιστήμη της εξόρυξης δεδομένων, καθώς είναι επεκτάσιμος και σχετικά απλός στην υλοποίηση του. Συνοπτικά, ο Naive Bayes δουλεύει ως εξής: δεδομένου ότι υπάρχει ένα στιγμιότυπο προς κατηγοριοποίηση το οποίο αναπαριστάται σε ένα διάνυσμα x και έχει κάποια n χαρακτηριστικά, του αναθέτει μια σειρά από πιθανότητες για τη κλάση στην οποία μπορεί να κατηγοριοποιηθεί. Ένα παράδειγμα προβλήματος κατηγοριοποίησης για τον κατηγοριοποιητή Naive Bayes αποτελέι η πρόβλεψη του καιρού για τη πρόταση: ”Οι παίκτες θα βγούνε στο γήπεδο αν δε βρέχει και έχει ζεστό καιρό”. Το πρόβλημα αυτό λύνεται δημιουργόντας πίνακες συχνοτήτων και πιθανοφάνειας και στη συνέχεια βρίσκοντας την πιθανότητα του να μη βρέχει και να έχει ζεστό καιρό, με τη χρήση υπολογιστικών πράξεων.

Στη συνέχεια, θα μελετήσουμε τον κατηγοριοποιητή των k-εγγύτερων γειτόνων, ο οποίος ανήκει στη κατηγορία των οκνυρών κατηγοριοποιητών και αποτελεί ένα από τα βασικά κομμάτια αυτής της εργασίας.

## 1.2 Ο αλγόριθμος κατηγοριοποίησης k εγγύτερων γειτόνων

Ο κατηγοριοποιητής των k-εγγύτερων γειτόνων [7], αποτελεί έναν αποτελεσματικό και πολυχρησιμοποιημένο οκνυρό αλγόριθμο κατηγοριοποίησης. Είναι εύκολος στην υλοποίηση και την ενσωμάτωσή του σε ένα σύστημα και μπορεί να χρησιμοποιηθεί σε ποικίλους τομείς και συστήματα. Ο συγκεκριμένος αλγόριθμος, μπορεί να επιλύσει τόσο προβλήματα κατηγοριοποίησης, όσο και παλινδρόμησης.

Ο κατηγοριοποιητής των k-εγγύτερων γειτόνων δε φτιάχνει κάποιο μοντέλο για τη κατηγοριοποίηση, καθώς κατατάσσεται στη κατηγορία των οκνυρών κατηγοριοποιητών. Ο αλγόριθμος, χρησιμοποιεί ολό-




Σχήμα 1.1: Παράδειγμα k-NN κατηγοριοποιητή για k=3 και k=5

κληρο το σύνολο των δεδομένων εκπαίδευσης όταν πρέπει να κατηγοριοποιήσει ένα νέο στιγμιότυπο σε μια κλάση. Πιο συγκεκριμένα, κατηγοριοποιεί ένα στιγμιότυπο x, ψάχνοντας όλο το σύνολο των δεδομένων εκπαίδευσης και παίρνοντας υπ'όψιν τους k εγγύτερους γείτονες του x, βάση μιας μετρικής που ορίζουμε στον αλγόριθμο για τις αποστάσεις. Αυτή η κλάση, λέμε ότι είναι η πλειοψηφούσα κλάση και αποφασίζεται από μια διαδικασία ψηφοφορίας μεταξύ των εγγύτερων γειτόνων. Αξίζει να ειπωθεί, πως για k=1, ο αλγόριθμος ονομάζεται 1-NN rule και στην ουσία μετατρέπεται σε αλγόριθμο εύρεσης της κλάσης του κοντινότερου γείτονα στο στιγμιότυπο x.

Στο σχήμα 1.1, απεικονίζεται ένα δυσδιάστατο παράδειγμα κατηγοριοποίησης ενός συνόλου δεδομένων. Πιο συγκεκριμένα, μας δείχνει πως σε ένα σύνολο δεδομένων με δύο κλάσεις, τα τετράγωνα και τους κύκλους κατηγοριοποιείται έναν νέο στιγμιότυπο που εισέρχεται εκείνη τη στιγμή στο σύνολο δεδομένων, το αστέρι. Αν υποθέσουμε ότι το k=3, τότε το αστέρι θα κατηγοριοποιηθεί στη κλάση των κύκλων, καθώς οι τρεις κοντινότεροί του γείτονες είναι κύκλοι, ενώ αν υποθέσουμε πως το k=5, τότε το αστέρι θα κατηγοριοποιηθεί στη κλάση των τετραγώνων, μιας και οι πέντε εγγύτεροί του γείτονες θα είναι τρία τετράγωνα και δύο κύκλοι.

Η απόδοση του κατηγοριοποιητή επηρεάζεται κυρίως από τον αριθμό που δίνουμε στη παράμετρο k. Το k το οποίο μας δίνει τη καλύτερη ακρίβεια πρόβλεψης δεν είναι ένας συγκεκριμένος αριθμός, αλλά διαφέρει αναλόγων με το σύνολο δεδομένων που χρησιμοποιείται. Ο μόνος τρόπος για να βρούμε ποιο k είναι αυτό που μας δίνει τη καλύτερη ακρίβεια, είναι να τρέξουμε τον αλγόριθμο κάθε φορά με διαφορετική παράμετρο k και να κρατήσουμε αυτό που μας δίνει τη μεγαλύτερη ακρίβεια, δηλαδή μέσω πολλαπλών δοκιμών. Σε σύνολα με πολύ θόρυβο στα δεδομένα τους θα χρειαστούν μεγάλα τιμή k, για να εξετάσουν μεγαλύτερες γειτονιές δεδομένων και να μη πάρουν λανθασμένες αποφάσεις λόγω θορύβου. Σε γενικές περιπτώσεις, η κατηγοριοποίση δε λειτουργεί σωστά για σύνολα που περιέχουν πολύ θόρυβο, καθώς επίσης και η βέλτιστη τιμή του k, ίσως να μην είναι πάντα η βέλτιστη. Αυτό, οφείλεται στ'ότι σε σύνολα δεδομένων με μεγάλο όγκο, μπορεί διαφορετικά k να είναι βέλτιστα για διαφορετικές περιοχές του συνόλου δεδομένων [8].

Σε περιπτώσεις δυαδικής κατηγοριοποίησης, το k πρέπει να έχει μονή τιμή (1-3-5 κτλ.), για να μην υπάρχουν ισσοπολίες κατά τη καταμέτρηση των στιγμιοτύπων σε ένα μέρος του συνόλου που εξετάζεται. Για όλες τις άλλες περιπτώσεις, το k μπορεί να πάρει οποιαδήποτε τιμή. Σε περιπτώσεις ισσοπαλίας κατά

τη ψηφοφορία, οι περισσότεροι αλγόριθμοι επιλέγουν τυχαία μια κλάση, ή επιλέγουν την κλάση του πιο κοντινού στιγμιότυπου.

Ένα άλλο σημαντικό θέμα, είναι ο τρόπος με τον οποίο υπολογίζεται η μετρική για τις αποστάσεις μεταξύ των στιγμιοτύπων. Αυτό, πρέπει να αποφασίζεται βάση του συνόλου δεδομένων το οποίο χρησιμοποιείται, καθώς κάθε σύνολο έχει διαφορετικά χαρακτηριστικά. Στις περιπτώσεις που έχουμε πραγματικούς και ακαίρεους αριθμούς, επικτρατεί η χρήση της Ευκλείδιας απόστασης σαν τη πιο διαδεδομένη μετρική. Ωστόσο, μπορούν να χρησιμοποιησθούν και άλλοι τρόποι για την εύρεση των αποστάσεων, όπως η Manhattan απόσταση, η Minkowski απόσταση, η Chebyshev απόσταση κλπ. [9]. Αν και έχουν βρεθεί πολλοί ακόμα τρόποι για την μέτρηση των απαστάσεων, όλα τα πειράματα σε αυτήν την εργασία έχουν γίνει με τη χρήση της Ευκλείδιας απόστασης, όπου χρειάστηκε. Επομένως, στιγμιότυπα τα οποία έχουν n χαρακτηριστικά, ορίζονται ως διανύσματα σε n διαστάσεις στο Ευκλείδιο χώρο και η απόσταση μεταξύ δύο σημείων p και q δίνεται από τη σχέση 1.1:

$$d(p,q) = d(q,p) = \sqrt{(q_1{-}p_1)^2 + (q_2{-}p_2)^2 + ... + (q_n{-}p_n)^2} = \sqrt{\sum_{i=1}^{n}(q_i{-}p_i)^2} \tag{1.1}$$

Απ' τη παραπάνω σχέση, προκύπτει πως διαφορετικό εύρος τιμών στα στιγμιότυπα μπορεί να δώσει διαφορετική τιμή για την απόσταση. Ακόμα και σε περιπτώσεις που όλα τα χαρακτηριστηκά των στιγμιοτύπων έχουν την ίδια βαρύτητα, στιγμιότυπα με πιο ευρύες τιμές μπορούν να έχουν μεγαλύτερη επιρροή στην απόσταση, απ'ότι στιγμιότυπα με πιο περιορισμένο εύρος τιμών. Ας υποθέσουμε τα χαρακτηριστικά θερμοκρασία και πληθυσμός μιας περιοχής. Αν υποθέσουμε πως η θερμοκρασία παίρνει τιμές από -10 μέχρι 40 βαθμούς, ενώ ο πληθυσμός μιας περιοχής μπορεί να είναι από μερικές χιλιάδες, ως και πολλά εκατομμύρια, τότε το δεύτερο χαρακτηριστικό μπορεί να επηρεάσει πολύ περισσότερο την απόσταση απ'ότι το πρώτο. Για τον λόγο αυτό, το εύρως των χαρακτηριστικών πρέπει να κανονικοποιείται, δηλαδή να έρχεται στο διάστημα [0,1]. Υποθέτοντας πως ένα σύνολο δεδομένων έχει n στιγμιότυπα και e χαρακτηριστικά για να κανονικοποιηθούν τα χαρακτηριστικά του, δηλαδή να πάνε στη μορφή 0 ή 1, πρέπει να εφαρμοστεί ο συγκεκριμένος τύπος για κάθε i-οστό στιγμιότυπο του συνόλου δεδομένων, όπως φαίνεται στη σχέση 1.2:

$$normalized(e_i) = \frac{e_i{-}Emin}{Emax{-}Emin} \tag{1.2}$$

Το Emin και Emax, αποτελούν την ελάχιστη και τη μέγιστη, αντίστοιχα τιμή, που μπορεί να πάρει κάθε στιγμιότυπο e. Η κανονικοποίηση των δεδομένων, είναι μια πολύ στάνταρ διαδικασία, την οποία τα περισσότερα λογισμικά εξόρυξης πληροφορίας τη κάνουν πλέον αυτομάτως.

Με το πέρασμα των χρόνων, έχουν προταθεί πολλές παραλλαγές για τον κατηγοριοποιητή των k-εγγύτερων γειτόνων. Η σημαντικότερη παραλλαγή, είναι ο αλγόριθμος distance-weigthed k-NN rule [10], ο οποίος χρησιμοποιεί μιας σταθμισμένης απόστασης συνάρτηση για να υπολογίσει καλύτερα την απόσταση με τους κοντινότερους γείτονες, σε σχέση με τους υπόλοιπους. Ο πιο κοντινός γείτονας, παίρνει για βάρος το 1, ενώ αυτός που βρίσκεται πιο μακριά το 0. Τα βάρη των υπολοίπων βρίσκονται μέσα σε αυτό το διάστημα. Έτσι ένα νέο στιγμιότυπο κατηγοριοποιείται στη κλάση η οποία έχει το μεγαλύτερο άθροισμα

βαρών.

## 1.3 Μειονεκτήματα του κατηγοριοποιητή k εγγύτερων γειτόνων

Αν και ο κατηγοριοποιητής k-εγγύτερων γειτώνων θεωρείται πως είναι ο πιο αποτελεσματικός σε σχέση με τους υπόλοιπους, έχει κάποια μειονεκτήματα τα οποία καθιστούν τη χρήση του άσκοπη σε μερικές περιπτώσεις. Ξεκινόντας, ο κατηγοριοποιητής των k-εγγύτερων γειτόνων έχει τεράστιο υπολογιστικό κόστος. Αυτό, ισχύει επειδή ο κατηγοριοποιητής πρέπει να υπολογίσει όλες τις αποστάσεις μεταξύ του νέου μη κατηγοριοποιημένου αντικείμενου που εισέρχεται εκείνη τη στιγμή στο σύνολο δεδομένων. Όσο μεγαλύτερο είναι το σύνολο δεδομένων στο οποίο τρέχουμε τον αλγόριθμο, τόσο μεγαλύτερος είναι και ο χρόνος που χρειάζεται για τη κατηγοριοποίηση. Ας υποθέσουμε, πως έχουμε ένα σύνολο δεδομένων με 30000 στιγμιότυπα στο σύνολο εκπαίδευσης και 20000 μη κατηγοριοποιημένα στιγμιότυπα. Το σύστημα, στο οποίο θα εκτελεστεί η διαδικασία κατηγοριοποίησης θα πρέπει να υπολογίσει εξακόσιες εκατομμύρια αποστάσεις, γεγονός απαράδεκτο, ασχέτως με τ'ότι τα σημερινά συστήματα είναι φτιαγμένα για να κάνουν περίπλοκους και πολλαπλούς υπολογισμούς σε ελάχιστο χρόνο. Πέρα όμως από τον αριθμό των αποστάσεων, θα πρέπει να αναφερθεί πως τα δεδομένα μπορεί να είναι και πολυδιάστατα. Αυτό, αλλάζει πολύ τη πολληπλοκότητα του προβλήματος και των υπολογισμών, καθώς για τον υπολογισμό των αποστάσεων μεταξύ των στιγμιοτύπων, θα χρειάζεται το σύστημα να συνυπολογίζει τις διάφορες διαστάσεις των δεδομένων.

Ακόμα ένα μειονέκτημα του κατηγοριοποιητή k-εγγύτερων γειτόνων, είναι ο μεγάλος αποθηκευτικός χώρος που απαιτείται για την αποθήκευση του συνόλου δεδομένων εκπαίδευσης, το οποίο χρησιμοποιεί ως μοντέλο για τη κατηγοριοποίηση. Οι πρόθυμοι αλγόριθμοι κατηγοριοποίησης, όπως προαναφέρθηκε σε προηγούμενη παράγραφο, αφότου φτιάξουν ένα μοντέλο κατηγοριοποίησης από το σύνολο δεδομένων εκπαίδευσης, αποδεσμεύουν τη μνήμη από αυτό. Οι οκνυροί αλγόριθμοι και συγκεκριμένα ο κατηγοριοποιητής των k-εγγύτερων γειτόνων, χρειάζεται να τρέξει σε σύστημα που να περιέχει μεγάλη χωρητικότητα μνήμης RAM, ώστε να έχει ανα πάσα στιγμή σε διαθεσιμότητα όλο το σύνολο των δεδομένων εκπαίδευσης.

Το τελευταίο μειονέκτημα του κατηγοριοποιητή των k-εγγύτυτερων γειτόνων, είναι η ευαισθησία που έχει στον θόρυβο των δεδομένων. Πιο συγκεκριμένα, η ακρίβεια πρόβλεψης στη κατηγοριοποίηση των δεδομένων, σχετίζεται άμεσα με τη ποιότητα του συνόλου των δεδομένων, καθώς και με το αν έχουν σωστές ή όχι ετικέτες. Επιπλέον, συγκαλήψεις περιοχών διαφορετικών περιοχών δεδομένων, οδηγούν σε κακή κατηγοριοποίησή. Για την επίλυση αυτού του προβλήματος, ενδείκνυται η χρήση μεγάλου k, γεγονός που αναγκάζει τον αλγόριθμο να εξετάζει μεγαλύτερες περιοχές δεδομένων. Αυτό όμως, δεν είναι κάτι απόλυτο, καθώς μπορεί πολλές φορές να μπερδέψει τον κατηγοριοποιητή και να τον οδηγήσει σε λανθασμένα συμπεράσματα, ανάλογα πάντα με το εκάστοτε σύνολο δεδομένων. Για τον λόγο αυτό, συνιστάται να δοκιμάζονται πολλές και διαφορετικές τιμές για τη παράμετρο k, κατά την εκτέλεση του αλγορίθμου.

Όλες αυτές οι αδυναμίες και τα μειονεκτήματα του κατηγοριοποιητή των k-εγγύτερων γειτόνων, τα οποία προαναφέρθηκαν, έχουν τραβήξει τόσο τη προσοχή των ακαδημαικών, όσο και των επαγγελματιών της πληροφορικής και αποτελούν ενεργό αντικείμενο έρευνας για αυτούς ακόμα και στις μέρες μας.

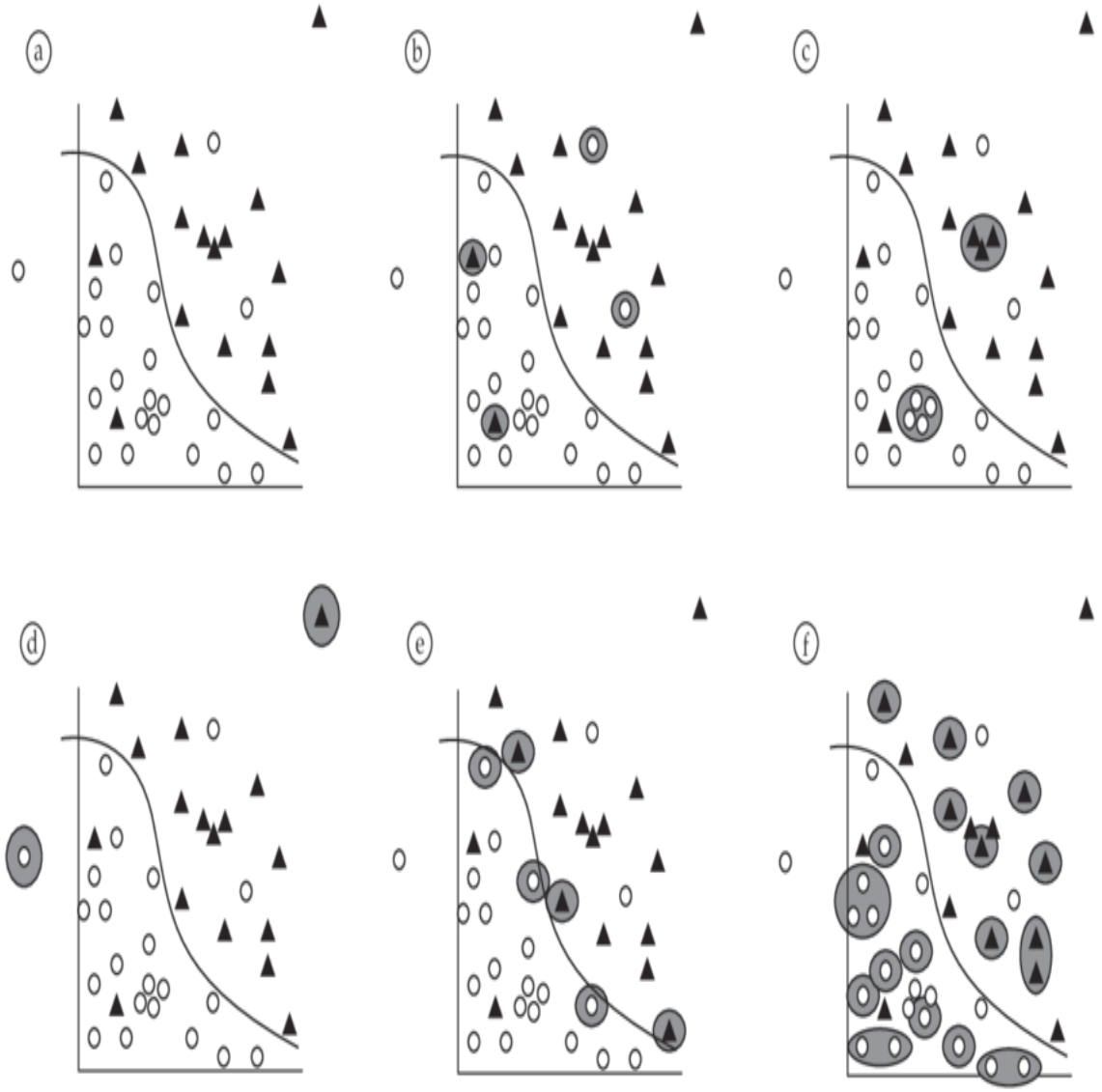
## 1.4 Τεχνικές μείωσης δεδομένων (DRT)

Οι τεχνικές μείωσης των δεδομένων που εφαρμόζονται πάνω σε σύνολα δεδομένων, δεν είναι άλλο από αλγόριθμους οι οποίοι έχουν ως σκοπό τη μείωση του όγκου των δεδομένων, ώστε οι κατηγοριοποιητές να μπορούν είτε να φτιάχνουν τα μοντέλα τους πιο εύκολα, αν είναι πρόθυμοι, είτε να απασχολούν μικρότερο ποσοστό της μνήμης RAM, αν πρόκειται για οκνυρούς. Αυτό, έχει νόημα αν και εφόσον τα δεδομένα δεν αλλοιώνονται σε βαθμό που το σύνολο δεδομένων χάνει κάποια από τα βασικά του χαρακτηριστικά και αν ο χρόνος και το υπολογιστικό κόστος που απαιτείται για τη διαδικασία της μείωσης του όγκου των δεδομένων δεν είναι τεράστιο.

Όσον αφορά τη κατηγοριοποίηση, τις τεχνικές μείωσεις των δεδομένων μπορεί να τις δει κάποιος από δύο διαφορετικές οπτικές, τη (i)μείωση του όγκου των δεδομένων και τη (i)μείωση των διαστάσεων των δεδομένων. Στα πλαίσια της συγκεκριμένης εργασίας, θα επικεντρωθούμε στη πρώτη κατηγορία αλγορίθμων. Οι τεχνικές μείωσης των δεδομένων, χωρίζονται σε δύο μεγάλες κατηγορίες. Η πρώτη κατηγορία, είναι οι αλγόριθμοι επιλογής στιγμιοτύπων(prototype selection algorithms), ενώ η δεύτερη κατηγορία είναι οι αλγόριθμοι παραγωγής στιγμιοτύπων ή αλγόριθμοι αφαίρεσης(prototype abstraction/generation algorithms) [11]. Οι αλγόριθμοι επιλογής στιγμιοτύπων, επιλέγουν συγκεκριμένα στιγμιότυπα με δικά τους κριτήρια, από το αρχικό σύνολο δεδομένων εκπαίδευσης, ενώ οι αλγόριθμοι παραγωγής στιγμιοτύπων δημιουργούν νέα στιγμιότυπα βάση αυτών που ήδη βρίσκονται στο σύνολο των δεδομένων εκπαίδευσης, τα οποία περιέχουν χαρακτηριστικά όλων των υπολοίπων. Σαν αποτέλεσμα αυτών, κάθε στιγμιότυπο αντιπροσωπεύει μια συγκεκριμένη περιοχή του πολυδιάστατου χώρου των δεδομένων.

Οι αλγόριθμοι επιλογής στιγμιοτύπων, μπορούν να χωριστούν σε δύο υποκατηγορίες, μπορεί είτε να είναι συμπυκνωτικοί(condensing), είτε επεξεργαστικοί(editing) αλγόριθμοι. Και οι συμπυκνωτικοί αλγόριθμοι και οι αλγόριθμοι παραγωγής στιγμιοτύπων έχουν ως σκοπό τη δημιουργία ενός μικρότερο συνόλου δεδομένων από το αρχικό, το συμπυκνωμένο σύνολο δεδομένων. Το συμπυκνωμένο σύνολο δεδομένων, θα χρειάζεται μικρότερο χώρο στη μνήμη RAM για την αποθήκευσή του και οι αλγόριθμοι κατηγοριοποίησης των στιγμιοτύπων θα έχουν χαμηλούς χρόνους εκτέλεσης, χωρίς να επηρεάζεται πολύ η ακρίβεια της πρόβλεψης για κάθε νέα κατηγοριοποίηση. Αντιθέτως, οι επεξεργαστικοί αλγόριθμοι έχουν ως στόχο να πετύχουν όσο το δυνατόν γίνεται μεγαλύτερη ακρίβεια στη πρόβλεψη της κατηγοριοποίησης, παρά να μειώσουν τον όγκο του συνόλου δεδομένων εκπαίδευσης. Το επιτυγχάνουν αυτό, βελτιώνοντας τα ήδη υπάρχοντα σύνολα δεδομένων, με μείωση του θορύβου τους και εξαλείφοντας οποιεσδήποτε επικαλύψεις υπάρχουν μεταξύ των στιγμιοτύπων στην επιφάνεια την οποία βρίσκονται, όπως φαίνεται στο Σχήμα 1.2. Αξίζει να σημειωθεί, πως κάποιοι συμπυκνωτικοί αλγόριθμοι, δανείζονται στοιχεία από τους επεξεργαστικούς αλγόριθμους, δημιουργόντας μια νέα υποκατηγορία αλγορίθμων επιλογής στιγμιοτύπων, τους υβριδικούς αλγορίθμους.

Στο σχήμα 1.2, φαίνεται οπτικά πως λειτουργούν οι αλγόριθμοι μείωσης των δεδομένων. Αρχικά, θα ήταν καλό να ειπωθεί πως οι λευκοί κύκλοι αναπαριστούν μία κλάση στιγμιοτύπων, ενώ τα μαύρα τρίγωνα μία άλλη. Η αφαίρεση του θορύβου από ένα σύνολο δεδομένων φαίνεται στο σχήμα b, όπου δύο τρίγωνα μαρκάρονται και αφαιρόνται από το σημείο του χώρου(ανάμεσα στους κύκλους) το οποίο βρίσκονται, πράγμα το οποίο γίνεται και με τους δύο κύκλους ανάμεσα στα τρίγωνα. Στο σχήμα c, αναπαριστάται το clustering των στιγμιοτύπων, το οποίο αν και δεν έχει άμεση σχέση με τη μείωση των δεδομένων, έχει με τη προεπεξεργασία και την ομαδοποίησή, πριν από τη μείωσή τους. Στη συνέχεια, στο σχήμα d, έχουμε



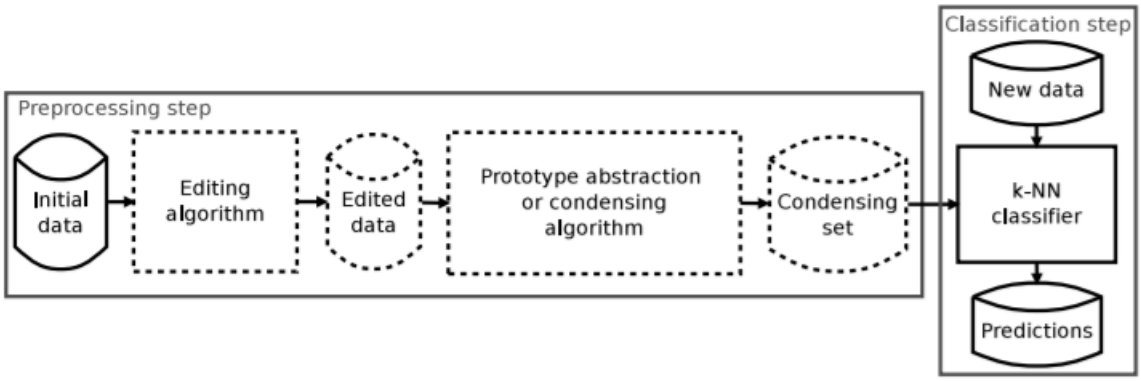

Σχήμα 1.2: Αφαίρεση θορύβου, εξάλειψη των επικαλύψεων και εξομάλυνση των ορίων μεταξύ των στιγμιοτύπων

την αφαίρεση δύο στιγμιοτύπων τα οποία θα μπορούσαν να χαρακτηριστούν άκυρα, όσον αφορά το ολικό σύνολο δεδομένων. Αυτό ισχύει, καθώς αυτά τα δύο στιγμιότυπα στις συγκεκριμένες θέσεις τις οποίες βρίσκονται δε μπορούν να μας δώσουν κάποια άξια αποτελέσματα όσον αφορά τα χαρακτηριστικά του συνόλου δεδομένων. Στο σχήμα e φαίνεται πως αντιμετωπίζονται τυχόν επικαλύψεις μεταξύ των ορίων που έχει θέσει ο εκάστοτε αλγόριθμος κατηγοριοποίησης, ενώ στο τελευταίο σχήμα έχουμε τη συνέχεια του σχήματος c, δηλαδή του clustering των στιγμιοτύπων.

Για να κρίνουμε το κατα πόσο ένας αλγόριθμος μείωσης δεδομένων λειτουργεί σωστά, μπορούμε να χρησιμοποιήσουμε ποικίλα κριτήρια. Στα πλαίσια της συγκεκριμένης εργασίας, χρησιμοποιούμε τη ποσοστιαία μείωση των δεδομένων(reduction rate), την ακρίβεια της κατηγοριοποίησης των στιγμιοτύπων(accuracy) και το υπολογιστικό κόστος που χρειάστηκε(computational cost). Η ποσοστιαία μείωση των δεδομένων, μας δείχνει το κατά πόσο μειώθηκε σε ποσοστό ένα σύνολο δεδομένων, βάση του αρχικού του μεγέθους. Μεγάλη ποσοστιαία μείωση των δεδομένων, μπορεί να σημαίνει μικρότερο υπολογιστικό κόστος για έναν αλγόριθμο, αλλά πιθανότατα θα έχει ένα αντίκτυκπο στην ακρίβεια της

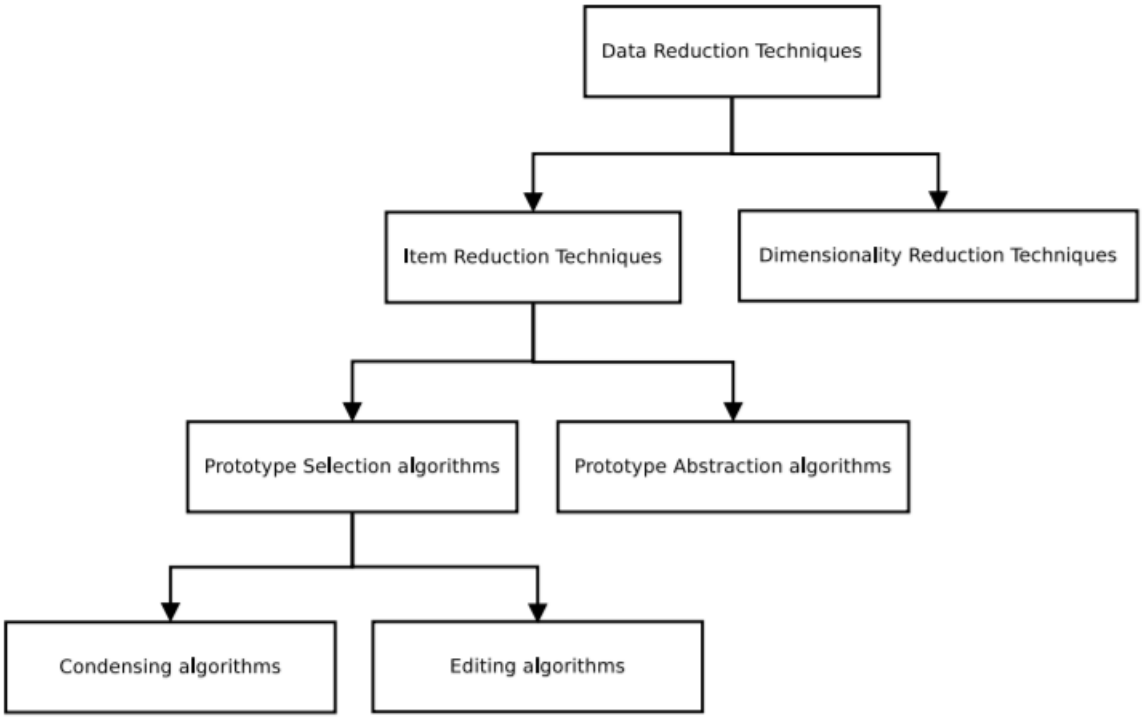
κατηγοριοποίησης. Η μεγάλη ακρίβεια κατηγοριοποίησης, είναι κάτι το ιδανικό για έναν αλγόριθμο κατηγοριοποίησης, αλλά τις περισσότερες φορές έρχεται σε αντίθεση με τον υπολογιστικό κόστος και τον χρόνο εκτέλεσης. Μικρό υπολογιστικό κόστος, οδηγεί σε μικρό χρόνο εκτέλεσης αλλά συνήθως έχει αρνητική επιρροή στην μείωση των δεδομένων και στην ακρίβεια της κατηγοριοποίησης. Για τους λόγους αυτούς, πειραματικά ψάχνουμε να βρούμε ποιες είναι οι ιδανικές παράμετροι που πρέπει να δώσουμε σε έναν αλγόριθμο ώστε να λειτουργήσει όσο το δυνατόν καλύτερα γίνεται, γεγονός που σημαίνει πως οι παράμετροι αυτοί δεν είναι συγκεκριμένοι, αλλά αλλάζουν ανάλογα με το εκάστοτε σύνολο δεδομένων. Στα πλάσια του συγκεκριμένου κειμένου, και τα τρία αυτά κριτήρια θεωρείται πως έχουν την ίδια βαρύτητα.

Η αφαίρεση του θορύβου, αποτελεί μια διαδικασία υψίστης σημασίας για τη κατηγοριοποίηση των δεδομένων. Πιο συγκεκριμένα, η ποσοστιαία μείωση σε ένα σύνολο δεδομένων έχει άμεση σχέση με τον θόρυβο των δεδομένων. Πολύς θόρυβος μεταξύ των δεδομένων, οδηγεί σε χαμηλή ποσοστιαία μείωση δεδομένων, κατά την εκτέλεση ενός συμπυκνωτικού αλγορίθμου, ή αλγορίθμου αφαίρεσης/παραγωγής στιμιοτύπων. Για τον λόγο αυτό, εκτελείται προτύτερα ένας επεξεργαστικός αλγόριθμος επιλογής στιγμιοτύπων, ο οποίος βοηθάει στην αύξηση της ακρίβειας και της ποσοστιαίας μείωσης του συμπυκνωμένου συνόλου δεδομένων [12, 13].




Σχήμα 1.3: Εκτέλεση του κατηγοριοποιητή k-NN αφότου έχει γίνει μείωση των δεδομένων

Στο σχήμα 1.3, συνοψίζεται η διαδικασία εκτέλεσης του κατηγοριοποιητή k-εγγύτερων γειτόνων σε μειωμένα δεδομένα. Η συγκεκριμένη διαδικασία, μπορεί να χωριστεί σε δύο φάσεις, την προεπεξεργασία των δεδομένων και την κατηγοριοποίηση. Αν και η προεπεξεργασία των δεδομένων είναι προαιρετική, υπάρχουν τέσσερεις πιθανοί περιπτώσεις: (i)να μην υπάρξει καμία προεπεξεργασία των δεδομένων, (ii)μόνο επεξεργασία(editing), (iii)μόνο συμπίκνωση(condensing) και (iv) επεξεργασία και συμπίκνωση των δεδομένων μαζί(condensing and editing). Αν το σύνολο δεδομένων εκπαίδευσης είναι μικρό και δεν έχει θόρυβο, τότε είμαστε στην περίπτωση (i). Όταν το μέγεθος του συνόλου δεδομένων εκπαίδευσης είναι μικρό, αλλά εμπεριέχει αρκετό θόρυβο, τότε βρισκόμαστε στην περίπτωση (ii). Σε περίπτωση που ο όγκος των δεδομένων είναι τεράστιος, αλλά δεν εμπεριέχεται θόρυβος μεταξύ των δεδομένων, τότε είμαστε στην περίπτωση (iii). Τέλος, αν έχουμε μεγάλα σύνολα δεδομένων εκπαίδευσης και πολύ θόρυβο, μια αρκετά συχνή περίπτωση, τότε βρισκόμαστε στην περίπτωση (iv), στην οποία πρέπει να κάνουμε και επεξεργασία και συμπίκνωση των δεδομένων ταυτόχρονα.




Σχήμα 1.4: Ιεραρχική κατηγοριοποίηση κατηγοριών DRT

Κλείνοντας το κεφάλαιο, στο Σχήμα 1.4 φαίνονται σχηματικά όλοι οι αλγόριθμοι τεχνικών μείωσης δεδομένων, οι οποίοι αναφέρθηκαν στο συγκεκριμένο κεφάλαιο.

## 1.5 Κίνητρο

Ο λόγος για τον οποίο γίνεται αυτή η εργασία, είναι η εξέλιξη και η βελτίωση των ήδη υπάρχοντων τεχνικών μείωσης των δεδομένων με διαχωρισμού του χώρου. Χαρκατηριστικό παράδειγμα αυτής της κατηγορίας αλγορίθμων είναι ο αλγόριθμος παραγωγής στιγμιοτύπων RSP3, ο οποίος περιληπτικά, χωρίζει τα δεδομένα σε μικρότερες ομογενής ομάδες δεδομένων, μέχρι να μη μπορούν να χωριστούν περεταίρω. Ο RSP3 όμως είναι αργός, καθώς σε κάθε επανάληψη του πρέπει να βρίσκει τα δύο πιο απομακρυσμένα στιμιότυπα του εκάστοτε συνόλου, έπειτα να υπολογίζει τις αποστάσεις του κάθε σημείου με αυτά τα δύο σημεία και στη συνέχεια να αποφασίζει σε ποιο από τα δύο βρίσκονται πιο κοντά, ώστε να τα κατατάσσει στην κατάλληλη ομάδα αντίστοιχα.

Αυτό, τον καθιστά ακατάλληλο για μεγάλα σύνολα δεδομένων, καθώς οι υπολογισμοί που χρειάζονται σε αυτές τις περιπτώσεις κάνουν το υπολογιστικό κόστος τεράστιο, όπως επίσης και τον χρόνο εκτέλεσης του αλγορίθμου, γεγονός το οποίο φαίνεται στα αποτελέσματα των πειραμάτων στο κεφάλαιο 4. Κάτι που αξίζει να σημειωθεί επιπλέον, είναι πως αν το σύνολο δεδομένων περιέχει θόρυβο, είναι πολύ πιθανόν ο RSP3 να μας οδηγήσει σε υποομάδες οι οποίες αποτελούνται από μόνο ένα στιγμιότυπο, το οποίο θα αποτελεί θόρυβο. Επομένως, αν το σύνολο δεδομένων έχει θόρυβο, κατά την εκτέλεση του RSP3 δε μπορεί να επιτευχθεί μεγάλη ποσιστιαία μείωση του στο συμπυκνωμένο σύνολο δεδομένων.

## 1.6 Συνεισφορά

Ο σκοπός μας, μέσω αυτής της εργασίας είναι να συνεισφέρουμε στη βελτίωση του RSP3, δημιουργώντας και αναλύοντας διάφορες παραλλαγές του, οι οποίες θα παρουσιαστούν στο κεφάλαιο 3. Οι συγκεκριμένες παραλλαγές, έχουν κάποιος μικροαλλαγές σε σχέση με τον κλασσικό RSP3. Για παράδειγμα ο RSP3-RND, επιλέγει με τυχαίο τρόπο τα δύο στιγμιότυπα τα οποία στον κλασσικό RSP3 επιλέγονται βάση της απόστασης. Άλλο ένα παράδειγμα είναι ο CC-RSP3, ο οποίος δημιουργεί τα δύο σημεία ως δύο τεχνητά σημεία τα οποία είναι ο μέσος όρος των δύο πολυπληθέστερων κλάσεων του εκάστοτε συνόλου δεδομένων, το οποίο αλλάζει σε κάθε επανάληψη.

Στόχος αυτών των παραλλαγών, είναι να αυξηθεί η ταχύτητα στην εκτέλεση, η ακρίβεια στην κατηγοριοποίηση και η αύξηση της ποσοστιαίας μείωσης του αρχικού συνόλου δεδομένων. Τέλος, οι παραλλαγές αυτές έχουν ως σκοπό να λύσουν το πρόβλημα το οποίο έχει ο κλασσικός RSP3 με τον θόρυβο, θεωρόντας ως θόρυβο στιμιότυπα τα οποία βρίσκονται μόνα τους σε μια υποομάδα.

## 1.7 Οργάνωση της διπλωματικής

Η συγκεκριμένη διπλωματική εργασία, αποτελείται από 5 κεφάλαια. Μέχρι εδώ, έχει γίνει μια σύντομη αναφορά στο τι θα ακολουθήσει, καθώς επίσης έχουν παρουσιαστεί μερικά βασικά πράγματα σχετικά με τη κατηγοριοποίηση και τις τεχνικές μείωσης δεδομένων.

Στο δεύτερο κεφάλαιο, θα αναφερθούν αναλυτικά κάποιες κλασσικές και ήδη υπάρχουσες τεχνικές μείωσης δεδομένων με βάση τον διαχωρισμό του χώρου τους. Πιο συγκεκριμένα, το κεφάλαιο αυτό θα περιγραφούν οι μέθοδοι CJA, RSP1, RSP2 και RSP3. Αυτοί οι αλγόριθμοι, αν και ήδη γνωστοί, είναι απαραίτητο να αναφερθούν καθώς αποτελούν του προγόνους των προτεινόμενων παραλλαγων που θα παρουσιαστούν στο τρίτο κεφάλαιου. Αξίζει να αναφερθεί ότι στο πλαίσιο των πειραμάτων ασχοληθήκαμε μόνο με τον RSP3, μεταξύ αυτών των τεσσάρων αλγορίθμων επειδή ο RSP3 είναι μόνος από αυτούς του αλγορίθμους που καθορίζουν το μέγεθος του συμπυκνωμένου συνόλου αυτόματα, χωρίς ο χρήστης να πρέπει να προσδιορισεί τιμή σε κάποια παράμετρο.

Στο τρίτο κεφάλαιο, επικεντρονώμαστε πάλι στις τεχνικές μείωσης δεδομένων. Αυτή τη φορά όμως παρουσιάζονται οι προτινόμενες, στα πλαίσια αυτής της εργασίας, τεχνικές μείωσης δεδομένων. Αυτές, περιλαμβάνουν τον ERSP3 αλγόριθμο στην υποενότητα 3.1, τους RSP3-RND και ERSP3-RND αλγορίθμους στην υποενότητα 3.2, τους CC-RSP3 και CC-ERSP στην υποενότητα 3.3 και τους CC2-RSP3 και CC2-ERSP3 στην υποενότητα 3.4. Επίσης, στο κεφάλαιο αυτό γίνεται η ανάλυση των παραπάνω αλγορίθμων τόσο περιγραφικά, όσο και με αλγορθμική μορφή (σε ψευδογλώσσα). Αξίζει να σημειωθεί, πως η ανάγνωση και κατανόηση του δεύτερου κεφαλαίου είναι απαραίτητη για την κατανόηση του συγκεκριμένου κεφαλαίου.

Στο τέταρτο κεφάλαιο, γίνεται η πειραματική μελέτη και η ανάλυση των αποτελεσμάτων για τα διάφορα σύνολα δεδομένων, πάνω στα οποία εκτελέστηκαν οι αλγόριθμοι του δεύτερου και τρίτου κεφαλαίου. Αρχικά, αναφέρεται ο τρόπος με τον οποίο οργανώσαμε το πειραματικό μας περιβάλλον, καθώς και τα εργαλεία που χρησιμοποιήσαμε. Γίνεται μια εκτενής περιγραφή των χαρακτηριστηκών του κάθε συνόλου δεδομένων πάνω στα οποία εκτελέστηκαν τα πειράματα. Επιπλέον, επεξηγούνται έννοιες όπως το cross-validation, η ευκλείδια απόσταση, το normalization και μετρικές όπως το accuracy και το reduction

rate. Αυτές όλες οι έννοιες, αποτελούν είτε εργαλεία που χρησιμοποιήσαμε για υπολογισμό αποστάσεων μεταξύ των στιγμιοτύπων, είτε εργαλεία που χρησιμοποιήθηκαν για τη κατηγοριοποίησή τους, είτε μετρικές για να αξιολογήσουν την απόδοσή των αλγορίθμων. Όλα αυτά, είναι απαραίτητο να ειπωθούν, καθώς με γνώμονα αυτές τις έννοιες θα μπορέσουμε να καταλάβουμε το κατά πόσο αποδοτικά ήταν τα πειράματά μας, όπως επίσης θα μπορέσει κάποιος στο μέλλον είτε να ξαναεκτελέσει τα πειράματα αυτά, είτε να πάρει τα δικά μας ευρήματα και να τα πάει στο επόμενο επίπεδο.

Στο πέμπτο, και τελευταίο κεφάλαιο, υπάρχουν τα συμπεράσματά μας από την εκπόνηση της συγκεκριμένης εργασίας. Τελειόνοντας, γίνονται προτάσεις και κατευθύνσεις για μελλοντική έρευνα προς οποιονδήποτε θέλει να ασχοληθεί με τη κατηγοριοποίηση και τις τεχνικές μείωσης δεδομένων, αλλά και σε όποιον θέλει να πάρει τα ευρήματα της συγκεκριμένης εργασίας και να τα εξελίξει.

# Κεφάλαιο 2ο: Τεχνικές Μείωσης Δεδομένων με βάση τον διαχωρισμό του χώρου δεδομένων

Οι ήδη γνωστές τεχνικές μείωσης των δεδομένων που βασίζονται σε διαχωρισμού του χώρου, είναι οι CJA, RSP1, RSP2 και RSP3. Αυτή η γενιά των αλγορίθμων διαιρεί επαναληπτικά τον πολυδιάστατο χώρο και δημιουργεί υποπεριοχές δεδομένων. Στο τέλος παράγεται ένα στιγμιότυπο που αντιπροσωπεύει την περιοχή. Η αρχική έμπνευση είναι του Chen και Jozwik [14].

Αυτοί οι αλγόριθμοι, συνδέονται μεταξύ τους καθώς ο ένας αποτελεί προηγούμενο του άλλου, με τον CJA να είναι ο παλιότερος. Ο RSP1, λύνει ένα από τα μειονεκτήματα του CJA, βελτιώνοντας την ακρίβειά του. Ο RSP2 αποτελεί βελτίωση του RSP1, και με τον RSP3 υιοθετείται έννοια της ομοιογένειας ή/και ανομοιογένειας μεταξύ των υποπεριοχών που βρίσκονται τα στιγμιότυπα. Ως αποτέλεσμα αυτού, ο RSP3 λύνει όλα τα προβλήματα του CJA.

## 2.1 Ο αλγόριθμος Chen και Jozwik (CJA)

Ο αρχικός αλγόριθμος ο οποίος πρότειναν οι Chen και Jozwik, είναι ο CJA [15]. Ο CJA δουλεύει, αρχικά επιλέγοντας τα δύο πιο απομακρυσμένα στιγμιότυπα $x$ και $y$ από το αρχικό σύνολο δεδομένων εκπαίδευσης. Η απόσταση που έχουν το $x$ με το $y$ είναι αυτή που μας δίνει τη διάμετρο του συνόλου δεδομένων. Στη συνέχεια, βάση των δύο αυτών στιγμιοτύπων, ο CJA χωρίζει το αρχικό σύνολο δεδομένων σε δύο υποσύνολα. Το πρώτο αποτελείται απο στιγμιότυπα που βρίσκονται πιο κοντά στο $x$, και είναι το υποσύνολο $S_y$, ενώ το δεύτερο αποτελείται απο στιγμιότυπα πιο κοντά στο $y$ και είναι το $S_y$. Ο CJA, συνεχίζει αυτή τη διαδικασία, χωρίζοντας κάθε φορά το μη ομειογενές υποσύνολο με τη μεγαλύτερη διάμετρο. Αν όλα τα υποσύνολα είναι ομοιγενή, ο CJA συνεχίζει και δημιουργεί νέα ομοιογενή υποσύνολα χρίζοντας κάθε φορά το ομοιογενές σύνολο με τη μεγαλύτερη διάμετρο. Αυτή η διαδικασία συνεχίζεται μέχρι έναν συγκεκριμένο αριθμό, μια παράμετρο που δίνει ο χρήστης. Στο τέλος, για κάθε υποσύνολο $S$, ο CJA υπολογίζει τον μέσο όρο των στιγμιοτύπων του κάθε υποσυνόλου και δημιουργεί ένα στιγμιότυπο, το οποίο είναι το μέσο στν στιγιοτύπων του υποσυνόλου και η κλάση του στιγμιοτύπου που δημιοργείται είναι η πλειοψηφούσα κλάση στο υποσύνολο. Το τελικό συμπυκνωμένο σύνολο δεδομένων, αποτελείται μόνο απο αυτά τα νέα στιγμιότυπα. Ένα για κάθε υποσύνολο.

Το μέσο στιγμιότυπο του κάθε υποσυνόλου $S$, υπολογίζεται βρίσκοντας τους μέσους όρους των N-οστών χαρακτηριστικών των στιγμιοτύπων $xi, i = 1, 2, ..., |S|$ που ανήκουν στο υποσύνολο $S$. Επομένως, τα N-οστά χαρακτηριστικά $m.dj$ του $m$, υπολογίζονται ως εξής:

$$m.d_j = \frac{1}{|S|} \sum_{x_i ES} (x_i.d_j, j = 1, 2, ..., t) \tag{2.1}$$

Ο αλγόριθμος που ακολουθεί(Αλγόριθμος 1), αποτελεί μια πιθανή υλοποίηση του CJA σε ψευδοκώδικα. Παίρνει, σαν παραμέτρους ένα αρχικό σύνολο δεδομένων εκπαίδευσης, το $TS$ και των αριθμό των στιγμιοτύπων $a$ που θα δημιουργηθούν.

Ο αλγόριθμος, χρησιμοποιεί μια δομή δεδομένων για να αποθηκεύσει τα υποσύνολα που δημιουργού-

**Algorithm 1** CJA

**Input:** $TS$, $a$
**Output:** $CS$

1: $S \leftarrow \varnothing$
2: add($S$, $TS$)
3: **for** $i = 2$ to $n$ **do**
4: $C \leftarrow$ επέλεξε το μη ομογενές υποσύνολο του αρχικού συνόλου με τη μεγαλύτερη διάμετρο
5: **if** $C == \varnothing$ {Όλα τα υποσύνολα είναι ομογενή} **then**
6: $C \leftarrow$ επέλεξε το ομογενή υποσύνλο του αρχικού συνόλου με τη μεγαλύτερη διάμετρο
7: **end if**
8: $(S_x, S_y) \leftarrow$ χώρισε το $C$ σε 2 υποσύνολα
9: add($S$, $S_x$)
10: add($S$, $S_y$)
11: remove($S$, $C$)
12: **end for**
13: $CS \leftarrow \varnothing$
14: **for** κάθε υποσύνολο $T \in S$ **do**
15: $r \leftarrow$ υπολόγισε το μέσο στιγμιότυπο μέσω των στιγμιοτύπων του $T$
16: $r.label \leftarrow$ βρες την επικρατέστερη κλάση στο $T$
17: $CS \leftarrow CS \cup \{r\}$
18: **end for**
19: **return** $CS$

νται κατά την εκτέλεσή του. Στις γραμμές 1 και 2, φαίνεται πως το $TS$, δηλαδή ολόκληρο το σύνολο δεδομένων, αποθηκεύεται στην αρχή ως ένα υποσύνολο. Στη συνέχεια, το μη ομογενή σύνολο με τη μεγαλύτερη διάμετρο χωρίζεται σε 2 υποσύνολα(γραμμές 4,8). Στις γραμμές 5-7, υπάρχει μια συνθήκη κατά την οποία αν όλα τα υποσύνολα είναι ομογενή, ο CJA χωρίζει το ομογενή υποσύνολο με τη μεγαλύτερη διάμετρο, σε 2 μικρότερα υποσύνολα. Και τα δύο υποσύνολα που δημιουργούνται, αποθηκεύονται στο $S$, καθώς το $C$ αφαιρείται, όπως φαίνεται στις γραμμές 9-11. Στη γραμμή 3, έχουμε τον αριθμό τον επαναλήψεων για τις οποίες εκτελείται ο αλγόριθμος, ο οποίος προκύπτει όταν έχουν δημιουργηθεί $a$ υποσύνολα. Τέλος, στις γραμμές 13-18, φαίνεται πως υπολογίζονται τα μέσα στιγμιότυπα για κάθε υποσύνολο και πως αυτά συμπεριλαμβάνονται στο $CS$.

Ο CJA, επιλέγει στη συνέχεια το επόμενο υποσύνολο το οποίο θα χωριστεί, εξετάζοντας τη διάμετρό του κάθε υποσυνόλου. Γενικά, ισχύει πως όσο μεγαλύτερο είναι ένα υποσύνολο σε διάμετρο, τόσο περισσότερα στιγμιότυπα θα περιέχει βάση πιθανοτήτων, το οποίο σημαίνει μεγαλύτερη ποσοστιαία μείωση του αρχικού συνόλου δεδομένων. Ο CJA δημιουργεί πάντα το ίδιο συμπυκνωμένο σύνολο δεδομένων, ασχέτως της σειράς διαχωρισμού των υποσυνόλων. Ωστόσο, έχει δύο μειονεκτήματα. Αρχικά, ο αλγόριθμος είναι παραμετρικός, που σημαίνει πως ο χρήστης πρέπει να εισάγει τον αριθμό των στιγμιοτύπων που θα δημιουργηθούν. Αυτό, αν και έχει τα θετικά του, καθώς ο χρήστης μέσω μιας διαδικασίας δοκιμών μπορεί να δημιουργήσει όσα υποσύνολα/στιμιότυπα θέλει καθώς και να βρει την παράμετρο που του δίνει τα καλύτερα αποτελέσματα, έχει και τα αρνητικά του, καθώς εμποδίζει τον αλγόριθμο στο να δημιουργήσει αυτός όσα υποσύνολα θέλει, βάσει των χαρακτηριστικών που έχουν τα δεδομένα. Η δεύτερη αδυναμία του, είναι πως τα στιγμιότυπα που δεν ανήκουν στη πολυπληθέστερη κλάση του εκάστοτε υποσυνόλου δεν φαίνονται στο συμπυκνωμένο σύνολο. Επομένως, όσα στιγμιότυπα σε ένα υποσύνολο δεν έχουν κλάση ίδια με αυτή της πολυπληθέστερης απλά αγνοούνται από τον αλγόριθμο και δεν

αντιπροσωπεύονται στο συμπυκνομένο σύνολο.

## 2.2 Ο αλγόριθμος RSP1

Η πρώτη από τις 3 παραλλαγές του CJA, ο RSP1 [16], δημιουργήθηκε για να λύσει το δεύτερο μειονέκτημα του CJA. Αυτό είναι το εξής: Ο CJA αγνοεί όλα τα στιγμιότυπα μέσα σε ένα υποσύνολο, τα οποία δεν ανήκουν στην πολυπληθέστερη κλάση του υποσυνόλου. Πιο συγκεκριμένα, ο RSP1 δημιουργεί τόσα μέσα στιγμιότυπα μέσα σε ένα υποσύνολο, όσα και ο αρθμός των διαφορετικών κλάσεων που υπάρχουν σε αυτό. Επομένως, υπολογίζει τον μέσο όρο για κάθε διαφορετική κλάση μέσα στο υποσύνολο, επιλύοντας έτσι τη δεύτερη αδυναμία του CJA. Αυτό, έχει θετική επίπτωση στην ακρίβεια, καθώς ο αλγόριθμος δεν αγνόει πλέον καμία κλάση μέσα στο υποσύνολο, εξαιτίας της ύπαρξης όλων των μέσων για τη κάθε κλάση ξεχωριστά. Άρα, όλα τα χαρακτηριστικά των δεδομένων του υποσυνόλου λαμβάνονται υπ'όψιν. Από την άλλη η μείωση των δεδομένων δεν είναι τόσο μεγάλη όσο στην περίπτση του CJA. Αυτό είναι απολύτως λογικό αφού ο RSP1, για τα μη ομοιγενεί υποσύνολα δημιουργεί περισσότερα από ενα στογμιότυπα για κάθε υπόσύνολο.

Ο RSP1 όμως, κρατάει πολλά κοινά στοιχεία με τον CJA, όσον αφορά τη λειτουργία του. Αρχικά, είναι και αυτός παραμετρικός, που σημαίνει ότι δέχεται είσοδο από τον χρήστη. Επιπλέον, επιλέγει το υποσύνολο που θα διαιρεθεί βάση του ποιο υποσύνολο έχει τη μεγαλύτερη διάμετρο. Αυτό, όπως έχει προαναφερθεί το κάνει για να επιτύχει όσο το δυνατόν γίνεται μεγαλύτερη ποσοστιαία μείωση σε σχέση με το αρχικό σύνολο, καθώς βάση πιθανοτήτων ένα υποσύνολο με μεγαλύτερη διάμετρο θα εμπεριέχει περισσότερα στιγμιότυπα.

## 2.3 Ο αλγόριθμος RSP2

Ο RSP1 διαφέρει με τον RSP2 στο πως επιλέγουν το επόμενο υποσύνολο που θα διασπαστεί. Όπως κάνει ο CJA, ο RSP1 χρησιμοποιεί τη διάμετρο του υποσυνόλου σαν κριτίριο διάσπασης, βασισμένο στην ιδέα ότι αν ένα υποσύνολο έχει μεγαλύτερη διάμετρο, κατά πάσα πιθανότητα θα περιέχει και περισσότερα στιγμιότυπα, που σημαίνει πως η ποσοστιαία μείωση του μεγέθους του υποσυνόλου θα είναι μεγαλύτερη. Σε αντίθεση όμως, ο RPS2 χρησιμοποιεί σαν κριτήριο διάσπασης τον βαθμό επικάλυψης. Βάση αυτού, τα στιγμιότυπα τα οποία ανήκουν στην ίδια κλάση θεωρείται πως βρίσκονται όσο το δυνατόν πιο κοντά γίνεται, ενώ στιγμιότυπα τα οποία ανήκουν σε διαφορετικές κλάσεις βρίσκοντα όσο πιο μακριά γινεται μεταξύ τους. Βάση της έρευνας [16], θεωρείται πιο αποδοτικό να διασπάται το υποσύνολο με τον μεγαλύτερο βαθμό επικάλυψης. Ο βαθμός επικάλυψης ενός υποσυνόλου, είναι ο λόγος των μέσων αποστάσεων μεταξύ 2 στιγμιοτύπων που ανήκουν σε διαφορετικές κλάσεις, προς τη μέση απόσταση 2 στιγμιοτύπων που ανήκουν στην ίδια κλάση.

## 2.4 Ο αλγόριθμος RSP3

Με τον αλγόριθμο RSP3, υιοθετείται η έννοια της ομοιογένειας. Ο αλγόριθμος, συνεχίζει να διασπάει τα υποσύνολα σε μικρότερα υποσύνολα έως ότου όλα τα υποσύνολα να είναι ομοιογενή, δηλαδή να περιέχουν στιγμιότυπα τα οποία ανήκουν σε μια και μόνο μια κλάση. Ο RSP3, μπορεί να χρησιμοποιήσει είτε τη μεγαλύρη διάμετρο, σαν εργαλείο για να αποφασίσει ποιο θα είναι το επόμενο υποσύνολο το

οποίο θα διασπαστεί, είτε τον βαθμό επικάλυψης. Βασικά, εφόσον όλα τα μη ομοιογενή υποσύνολα θα διασπαστούν, η επιλογή του κριτηρίου διάσπασης δεν εξυπηρετεί κανέναν σκοπό. Ο RSP3 αλγόριθμος, είναι ο μοναδικός RSP αλγόριθμος, ο οποίο αυτομάτως αποφασίζει ποιο θα είναι το μέγεθος του συμπυκνωμένου συνόλου δεδομένων, χωρίς να χρησιμοποιεί κάποια παράμετρο την οποία ο χρήστης περνάει χειροκίνητα. Ως αποτέλεσμα αυτού, ο RSP3 επιλύει και τα δύο βασικά προβλήματα του CJA. Αξίζει να ειπωθεί πως όπως και σε όλους τους προηγούμενους αλγορίθμους(τον CJA, RSP1, και RSP2), το συμπυκνωμένο σύνολο δεδομένων που δημιουργείται είναι ανεξάρτητο της σειράς που έχουν τα δεδομένα τα οποία βρισκόντουσαν στο αρχικό σύνολο δεδομένων εκπαίδευσης.

Μία πιθανή υλοποίηση για τον RSP3, είναι η εξής:

**Algorithm 2** RSP3

**Input:** $TS$
**Output:** $CS$

1: $S \leftarrow \varnothing$
2: add($S, TS$)
3: $CS \leftarrow \varnothing$
4: **repeat**
5: $C \leftarrow$ επέλεξε ένα υποσύνολο $\in S$
6: **if** $C$ είναι ομοιογενές υποσύνολο **then**
7: $r \leftarrow$ βρες το μέσο στιγμιότυπο υπολογίζοντας τα μέσα των στιγμιοτύπων του $C$
8: $r.label \leftarrow$ κλάση στιγμιοτύπων του $C$
9: $CS \leftarrow CS \cup \{r\}$
10: **else**
11: $(D_1, D_2) \leftarrow$ διάσπασε το $C$ σε 2 υποσύνολα
12: add($S$, $D_1$)
13: add($S$, $D_2$)
14: remove($S$, $C$)
15: **end if**
16: **until** IsEmpty($S$)
17: **return** $CS$

Ο προηγούμενος αλγόριθμος (Αλγόριθμος 2), αποτελεί τον ψευδοκώδικα του RSP3. Χρησιμοποιεί μια απλή δομή δεδομένων, το S, για να αποθηκεύει τα ανεπεξέργαστα σύνολα δεδομένων. Ξεκινόντας, όλο το σύνολο δεδομένων εκπαίδευσης($TS$) αποτελεί ένα ανεπεξέργαστο υποσύνολο δεδομένων και αποθηκεύεται στο $S$, όπως φαίνεται και στη γραμμή 2. Σε κάθε επανάληψη, ο RSP3 επιλέγει το υποσύνολο $C$ με το υψηλότερο κριτίριο διαχωρισμού (γραμμή 5) και ελέγχει αν το $C$ είναι ή δεν είναι ομογενή. Αν είναι ομογενή, το μέσο στιγμιότυπο υπολογίζεται βρίσκοντας τους μέσους όρους των στιγμιοτύπων του υποσυνόλου $C$ και τοποθετείται στο συμπυκνωμένο σύνολο δεδομένων ($CS$), όπως φαίνεται στις γραμμές 6-9. Αλλιώς, το $C$ διασπάται σε 2 υποσύνολα $D_1$ και $D_2$ (γραμμή 11), όπως γινόταν και στον CJA. Αυτά τα νέα υποσύνολα προστίθονται στο $S$ και το $C$ αφαιρείται από το $S$ (γραμμές 12-15). Αυτή η επαναληπτική διαδικασία συνεχίζεται έως ότου το σύνολο $S$ να μείνει κενό (γραμμή 16), που σημαίνει πως όλα τα υποσύνολα θα είναι ομογενή.

Σχετικά με τη λειτουργία του RSP3, παρατηρούμε ότι δημιουργεί λίγα στιγμιότυπα για την αναπαράσταση των περιοχών που δεν είναι κοντά στα άκρα της περιοχής μιας κλάσης, ενώ δημιουργεί πολλά στιγμιότυπα για την αναπαράσταση περιοχών κοντά στα άρκα των κλάσεων. Πιο συγκεκριμένα, η πο-

σοστιαία μέιωση στο σύνολο δεδομένων που επιτυγχάνεται από τον RSP3, εξαρτάται κατά πολύ από τον θόρυβο που υπάρχει στα δεδομένα. Όσο περισσότερος είναι ο θόρυβος στα δεδομένα, τόσο μικρότερα υποσύνολα δημιουργούνται, και σαν αποτέλεσμα αυτού, έχουμε μικρότερη ποσοστιαία μείωση στα δεδομένα. Αξίζει να σημειωθεί πως για να βρούμε την απόσταση μεταξύ των δύο μακρινότερων στιγμιοτύπων σε ένα υποσύνολο, πρέπει να βρούμε τις αποστάσεις μεταξύ όλων των στιγμιοτύπων μεταξύ τους και να κρατήσουμε τη μεγαλύρη από αυτές. Αυτές, είναι διαδικασίες που χρειάζονται πολύ χρόνο και έχουν μεγάλο υπολογιστικό κόστος. Έτσι, όταν τα σύνολα δεδομένων είναι μεγάλα, ο αλγόριθμος RSP3 καθιστάται μη ικανός για να μας δώσει τα αποτελέσματα που θα επιθυμούσαμε εξαιτείας του μεγάλου υπολογιστικού κόστους.

# Κεφάλαιο 3ο: Προτεινόμενες τεχνικές μείωσης δεδομένων με βάση τον διαχωρισμό του χώρου δεδομένων

Στο συγκεκριμένο κεφάλαιο, το οποίο είναι ίσως και το σημαντικότερο απ'όλα, θα παρουσιαστούν αναλυτικά οι τεχνικές μείωσης των δεδομένων οι οποίες προτείνονται στα πλαίσια αυτής της εργασίας. Αφιερώθηκε σημαντικός χρόνος στην ανάπτυξη των συγκεκριμένων αλγορίθμων, καθώς και στην εκτέλεση των πειραμάτων τα οποία θα ακολουθήσουν στο επόμενο κεφάλαιο.

Η γλώσσα που χρησιμοποιήθηκε για την ανάπτυξη των αλγορίθμων είναι η C++. Επιλέξαμε αυτή τη γλώσσα, καθώς μας δίνει πολλές επιλογές και δυνατότητες όσον αφορά τη διαχείρηση της μνήμης. Αυτό, γίνεται μέσω των pointers, που είναι μοναδικό χαρακτηριστικό αυτής της γλώσσας, και μας δίνουν τη δυνατότητα να έχουμε άμεση πρόσβαση στη μνήμη, γεγονός που μας δίνει καλύτερους χρόνους εκτέλεσης. Επίσης, εξαιτίας του μεγάλου όγκου των δεδομένων και συνόλων δεδομένων που χρησιμοποιούμε, κρίνεται απαραίτητη η ορθή διαχείρηση της μνήμης για την εξομάλυνση της εκτέλεσης του κάθε αλγορίθμου.

## 3.1 Ο αλγόριθμος ERSP3

Η πρώτη παραλλαγή η οποία προτείναμε, στα πλαίσια αυτής της εργασίας, είναι ο αλγόριθμος ERSP3. Ο συγκεκριμένος αλγόριθμος, αν και είναι σχεδόν πανομοιότυπος με τον κλασσικό RSP3, έχει μία σημαντική διαφορά: Αν βρεθεί ένα υποσύνολο μέσα στο οποίο βρίσκεται ένα και μόνο ένα στιγμιότυπο, τότε αυτό το υποσύνολο θεωρείται θόρυβος. Αυτό, σημαίνει ότι ο αλγόριθμος δεν προχωράει στους υπολογισμούς τους οποίους θα έκανε για τον υπολογισμό των μέσων για τα στιγμιότυπα του υποσυνόλου, αλλά θα το απομακρύνει από τα δεδομένα. Συνεπώς, για κάθε υποσύνολο με μόνο ένα στιγμιότυπο, ο ERSP3 δεν δημιουργεί στιγμιότυπο στο συμπυκνωμένο σύνολο.

Ο λόγος για τον οποίο δημιουργήθηκε ο ERSP3 αλγόριθμος, είναι αρχικά, η βελτίωση της ακρίβειας κατά τη κατηγορίοποίηση ενός στιγμιοτύπου. Αυτό, θα επιτευχθεί καθώς συνήθως όταν ένα στιγμιότυπο μίας κλάσης βρίσκεται μόνο του ανάμεσα σε άλλα στιγμιότυπα διαφορετικών κλάσεων, κατά πάσα πιθανότητα αποτελεί θόρυβο. Αφαιρώντας τον θόρυβο από ένα σύνολο/υποσύνολο δεδομένων, αυξάνουμε την ακρίβεια της κατηγοριοποίησης. Επίσης, αφαιρώντας ένα στιγμιότυπο που πιθανόν αποτελεί θόρυβο, μειώνουμε τον αριθμό τον αποστάσεων που χρειάζεται να υπολογίσει ο αλγόριθμος, καθώς δε χρειάζεται να υπολογίσει επιπλέον αποστάσεις, δηλαδή τις αποστάσεις ενός στιγμιοτύπου το οποίο είναι θόρυβος, και ο υπολογισμός τους δε μας οφελεί σε κάτι, αλλά αντίθετα επηρεάζει αρνητικά την ακρίβειά μας. Ο υπολογισμός λιγότερων αποστάσεων, έχει ως αποτέλεσμα την ελάττωση του CPU time, δηλαδή του χρόνου του CPU τον οποίο δεσμεύει ο αλγόριθμός μας. Βέβαια, για να ισχύουν όλα αυτά και για να έχουν κάποια ουσία, πρέπει ο αλγόριθμός αυτός που προτείνουμε να βελτιώνει την ακρίβεια του αλγόριθμου εξ΄αρχής, ώστε να έχουμε καλύτερη κατηγοριοποίηση των δεδομένων. Στη συνέχεια, θα παρουσιαστεί μια πιθανή υλοποίηση του ERSP3 και θα επεξηγηθεί αναλυτικά.

Μια πιθανή υλοποίηση του ERSP3, είναι η εξής:

**Algorithm 3** ERSP3

**Input:** $TS$
**Output:** $CS$

1: $S \leftarrow \varnothing$
2: add($S, TS$)
3: $CS \leftarrow \varnothing$
4: **repeat**
5: $C \leftarrow$ επέλεξε ένα υποσύνολο $\in S$
6: **if** ο αριθμός των στιγμιοτύπων του $C$ είναι > 1 **then**
7: **if** $C$ είναι ομοιογενή υποσύνολο **then**
8: $r \leftarrow$ βρες το μέσο στιγμιότυπο υπολογίζοντας τα μέσα των στιγμιοτύπων του $C$
9: $r.label \leftarrow$ κλάση στιγμιοτύπων του $C$
10: $CS \leftarrow CS \cup \{r\}$
11: **else**
12: $(D_1, D_2) \leftarrow$ διάσπασε το $C$ σε 2 υποσύνολα
13: add($S$, $D_1$)
14: add($S$, $D_2$)
15: remove($S$, $C$)
16: **end if**
17: **end if**
18: **until** IsEmpty($S$)
19: **return** $CS$

Όσον αφορά τη λειτουγία του ERSP3 αλγόριθμου, χρησιμοποιείται απλή δομή δεδομένων, το S, για να αποθηκεύει τα ανεπεξέργαστα σύνολα δεδομένων. Αρχικά,το σύνολο δεδομένων εκπαίδευσης(TS) αποτελεί ένα μη επεξεργασμένο υποσύνολο δεδομένων και αποθηκεύεται στο S(γραμμή 2). Ο RSP3 σε κάθε του επανάληψη επιλέγει το υποσύνολο C με το υψηλότερο κριτίριο διαχωρισμού (γραμμή 5). Στη συνέχεια, δηλαδή στη γραμμή 6, ελέγχεται αν ο αριθμός των στιγμιοτύπων στο υποσύνολο το οποίο επιλέχθηκε είναι μεγαλύτερος του 1. Αν είναι μεγαλύτερος του 1 τότε η διαδικασία εκτέλεσης του αλγορίθμου θα συνεχιστεί κανονικά, ενώ αν δεν είναι, τότε το συγκεκριμένο στιγμιότυπο/υποσύνολο θα θεωρηθεί θόρυβος. Έπειτα, γίνεται έλεγχος για το αν το C υποσύνολο είναι ή δεν είναι ομογενή. Αν είναι ομογενή, το μέσο στιγμιότυπο υπολογίζεται βρίσκοντας τους μέσους όρους των στιγμιοτύπων του υποσυνόλου C και τοποθετείται στο συμπυκνωμένο σύνολο δεδομένων (CS), όπως φαίνεται στιςγραμμές 7-10. Αλλιώς, το C διασπάται σε 2 υποσύνολα D1 και D2(γραμμή 12), όπως γινόταν και στον CJA. Αυτά τα νέα υποσύνολα προστίθονται στο S και το C αφαιρείται από το S(γραμμές 13-15). Αυτή η επαναληπτική διαδικασία συνεχίζεται έως ώτου το σύνολο S να μείνει κενό(γραμμή 18), που σημαίνει πως όλα τα υποσύνολα θα είναι ομογενή.

## 3.2 Οι αλγόριθμοι RSP3-RND και ERSP3-RND

Η δεύτερη παραλλαγή που προτείναμε, είναι ο αλγόριθμος RSP3-RND και ERSP3-RND αντίστοιχα. Θεωρήσαμε σωστό να τους συμπεριλάβουμε στο ίδιο κεφάλαιο, καθώς η μόνη τους διαφορά είναι ότι ο ERSP3-RND αν συναντήσει ένα υποσύνολο με ένα μοναδικό στιγμιότυπο θα το θεωρήσει θόρυβο, ενώ ο RSP3-RND θα φερθεί σε ένα τέτοιο υποσύνολο σαν ένα κανονικό υποσύνολο. Η παραλλαγή που προσφέρει ο RND αλγόριθμος, είναι πως αντί να υπολογίζει τις αποστάσεις όλων των στιγμιοτύπων

μεταξύ τους για να εντοπίσει τα 2 που έχουν τη μεγαλύτερη απόσταση μεταξύ τους όπως κάνει ο RSP3, βρίσκει τα 2 πρώτα στιγμιότυπα σε ένα υποσύνολο και τα θεωρεί ως τις δύο θέσεις με τις οποίες στη συνέχεια θα κάνει τους υπολογισμούς των αποστάσεων που ορίζει ο αλγόριθμος RSP3.

Ο λόγος για τον οποίο δημιουργήθηκε αυτός ο αλγόριθμος, δεν έχει να κάνει τόσο με την βελτίωση της ακρίβειας της κατηγοριοποίησης του αλγορίθμου RSP3, αλλά με τη δημιουργία αποτελεσμάτων πάνω στα οποία θα μπορέσουμε να συγκρίνουμε αποτελέσματα άλλων αντίστοιχων αλγορίθμων. Ωστόσο, οι αλγόριθμοι RSP3-RND και ERSP3-RND, αναμένεται πως θα βελτιώσουν το CPU time, καθώς όντας random ως προς την επιλογή των 2 πρώτων στιγμιοτύπων σε ένα υποσύνολο, περιμένουμε να χρειάζονται λιγότερο χρόνο κατά την εκτέλεση και τη δημιουργία του συμπυκνομένου συνόλου. Αυτό, όμως δεν είναι απόλυτο, καθώς μπορεί τα στιγμιότυπα που επιλέγονται για τους μεταξύ τους υπολογισμούς να οδηγούν σε περίπλοκες πράξεις ή να μην είναι τα ιδανικά (το οποίο είναι λογικό μιας και τα επιλέγουμε στη τύχη). Στην πράξη, αναμένουμε ότι οι RSP3-RND και ERSP3-RND θα μειώνουν στο ελάχιστο τον χρόνο εκτέλεσης για τη δημιουργία του συμπυκνομένου συνόλου αλλά θα οδηγεί σε χειρότερα ποσοστά μείωσης των δεδομένων. Στη συνέχεια, θα παρουσιαστούν και θα εξηγηθούν οι αλγόριθμοι RSP3-RND και ERSP3-RND.

Μια πιθανή υλοποίηση του RSP3-RND, είναι η εξής:

**Algorithm 4** RSP3-RND

**Input:** $TS$
**Output:** $CS$

1: $S \leftarrow \varnothing$
2: add($S, TS$)
3: $CS \leftarrow \varnothing$
4: **repeat**
5: $C \leftarrow$ επέλεξε ένα υποσύνολο $\in S$
6: Επέλεξε τα 2 πρώτα στιγμιότυπα του $C$ για τον υπολογισμό των αποστάσεων
7: **if** $C$ είναι ομοιογενή υποσύνολο **then**
8: $r \leftarrow$ βρες το μέσο στιγμιότυπο υπολογίζοντας τα μέσα των στιγμιοτύπων του $C$
9: $r.label \leftarrow$ κλάση στιγμιοτύπων του $C$
10: $CS \leftarrow CS \cup \{r\}$
11: **else**
12: $(D_1, D_2) \leftarrow$ διάσπασε το $C$ σε 2 υποσύνολα
13: add($S$, $D_1$)
14: add($S$, $D_2$)
15: remove($S$, $C$)
16: **end if**
17: **until** IsEmpty($S$)
18: **return** $CS$

Ο Αλγόριθμος 4, είναι ίδιος με τον Αλγόριθμο 2 του κεφαλαίου 2.4, δηλαδή με τον απλό RSP3, αλλά έχει μία διαφορά ως προς τα στιγμιότυπα τα οποία επιλέγει για τον υπολογισμό των μεταξύ τους αποστάσεων. Όπως προαναφέρθηκε, ο αλγόριθμος επιλέγει στη τύχη τα 2 πρώτα στιγμιότυπα του εκάστοτε υποσυνόλου, κάτι το οποίο φαίνεται στον αλγόριθμος στην γραμμή 6. Στη συνέχεια, ο αλγόριθμος συνεχίζει όπως τον ξέρουμε ήδη από το κεφάλαιο 2.4.

Όσον αφορά τον αλγόριθμο ERSP3-RND, μια πιθανή υλοποίηση του είναι η εξής:

**Algorithm 5** ERSP3-RND

**Input:** $TS$
**Output:** $CS$

```
1:  S ← ∅
2:  add(S, TS)
3:  CS ← ∅
4:  repeat
5:     C ← επέλεξε ένα υποσύνολο ∈ S
6:     Επέλεξε τα 2 πρώτα στιγμιότυπα του C για τον υπολογισμό των αποστάσεων
7:     if ο αριθμός των στιγμιοτύπων του C είναι > 1 then
8:        if C είναι ομοιογενή υποσύνολο then
9:           r ← βρες το μέσο στιγμιότυπο υπολογίζοντας τα μέσα των στιγμιοτύπων του C
10:          r.label ← κλάση στιγμιοτύπων του C
11:          CS ← CS ∪ {r}
12:       else
13:          (D1, D2) ← διάσπασε το C σε 2 υποσύνολα
14:          add(S, D1)
15:          add(S, D2)
16:          remove(S, C)
17:       end if
18:    end if
19: until IsEmpty(S)
20: return CS
```

Ο Αλγόριθμος 5, συνδυάζει τα ιδιαίτερα στοιχεία του ERSP3 και του RSP3-RND. Δηλαδή, επιλέγει στη τύχη τα 2 πρώτα στιγμιότυπα του εκάστοτε υποσυνόλου όπως κάνει ο RND (γραμμή 6), αλλά επίσης θεωρεί ως θόρυβο υποσύνολα τα οποία περιέχουν 1 μοναδικό στιγμιότυπο μέσα σε αυτά, όπως κάνει ο ERSP3 (γραμμή 7).

## 3.3 Οι αλγόριθμοι RSP3-CC και ERSP3-CC

Οι αλγόριθμοι RSP3-CC και ERSP3-CC, οι οποίοι είναι πανομοιότυποι, αποτελούν ένα από τα κύρια και σημαντικότερα μέρη αυτού του κεφαλαίου και της εργασίας γενικότερα. Κατά την εκτέλεση του αλγορίθμου RSP3-CC ισχύει ότι ισχύει για τον απλό RSP3, αλλά υπάρχουν μερικές διαφοροποιήσεις. Αρχικά, υπολογίζονται οι 2 πολυπληθέστερες κλάσεις σε ένα υποσύνολο, δηλαδή οι 2 κλάσεις οι οποίες σε αριθμό έχουν τον μεγαλύτερο αριθμό στιγμιοτύπων από τις υπόλοιπες. Αφού βρεθούνε οι 2 πολυπληθέστερες κλάσεις, τότε δημιουργούνται 2 νέα τεχνητά στιγμιότυπα από τον μέσο όρο των στιγμιοτύπων της πρώτης και της δεύτερης πολυπληθέστερης κλάσης. Στη συνέχεια, τα 2 αυτά τεχνητά στιγμιότυπα των 2 διαφορετικών κλάσεων χρησιμοποιούνται για τον υπολογισμό των αποστάσεων, όπως γίνεται και με τον RSP3 και η εκτέλεση του αλγορίθμου συνεχίζεται κανονικά.

Η ιδέα πίσω από αυτή την παραλλαγή είναι η εξής: Χωρίζοντας ένα μη-ομοιογενές υποσύνολο σε δυο υποσύνολα βάσει των κέντρων των πολυπληθέστερων κλάσεων στο υποσύνολο, είναι πιθανόν, να προκύψουν άμεσα ομοιογενή υποσύνολα και να μην απαιτηθεί να διαχωριστούν περαιτέρω. Έτσι ο αλγόριθμος μειώνει το υπολογιστικό κόστος δημιουργίας του συμπυκνομένου συνόλου. Παράλληλα αποφεύγεται ο

χρονοβόρος υπολογισμός όλων των αποστάσεων μεταξύ των στιγμιοτύπων του υποσυνόλου για την εύρεση των δύο πιο απομακτυσμένων στιγμιοτύπων.

Με αυτούς τους δύο αλγορίθμους, αποσκοπούμε στο να βελτιώσουμε μετρικές που έχουμε θέσει για την απόδοση ενός αλγορίθμου. Η ακρίβεια της κατηγοριοποίησης θεωρούμαι ότι θα παραμείνει σε υψηλά επίπεδα αφού οι μέσοι όροι των πολυπληθέστερων κλάσεων θα βοηθήσουν στον καλύτερο διαχωρισμό της επιφάνειας και συνεπώς των δεδομένων. Η ποσοστιαία μείωση των δεδομένων θα αυξηθεί αφού η χρήση των κέντρων των κλάσεων θα βοηθήσει στο να βρεθούν μεγάλύτερα ομοιογενή υποσύνολα. Τέλος, ο αριθμός των υπολογισμών των αποστάσεων θα μειωθεί επειδή αποφεύγεται ο χρονοβόρος υπολογισμός όλων των αποστάσεων μεταξύ των στιγμιοτύπων του υποσυνόλου για την εύρεση των δύο πιο απομακτυσμένων στιγμιοτύπων. Όλα αυτά, οδηγούν στο ότι και ο χρόνος εκτέλεσης της CPU (CPU time) θα μειωθεί αφού οι αποστάσεις που θα υπολογιστούν θα είναι λιγότερες. Στη συνέχεια, θα παρουσιαστούν και θα εξηγηθούν οι αλγόριθμοι RSP3-CC και ERSP3-CC.

Μια πιθανή υλοποίηση του RSP3-CC, είναι η εξής:

**Algorithm 6** RSP3-CC

**Input:** $TS$
**Output:** $CS$

1: $S \leftarrow \varnothing$
2: add($S, TS$)
3: $CS \leftarrow \varnothing$
4: **repeat**
5: $C \leftarrow$ επέλεξε ένα υποσύνολο $\in S$
6: **if** Ο αριθμός των κλάσεων του $C > 2$ **then**
7: Βρες τις 2 πολυπληθέστερες κλάσεις του $C$
8: Βρες τα μέσα αυτών των 2 κλάσεων, τα 2 νέα τεχνητά σημεία $p1$ και $p2$
9: **end if**
10: Χρησιμοποίησε τα 2 νέα σημεία για τον υπολογισμό των αποστάσεων
11: **if** $C$ είναι ομοιογενή υποσύνολο **then**
12: $r \leftarrow$ βρες το μέσο στιγμιότυπο υπολογίζοντας τα μέσα των στιγμιοτύπων του $C$
13: $r.label \leftarrow$ κλάση στιγμιοτύπων του $C$
14: $CS \leftarrow CS \cup \{r\}$
15: **else**
16: $(D_1, D_2) \leftarrow$ διάσπασε το $C$ σε 2 υποσύνολα
17: add($S$, $D_1$)
18: add($S$, $D_2$)
19: remove($S$, $C$)
20: **end if**
21: **until** IsEmpty($S$)
22: **return** $CS$

Στις γραμμές 6-10, είναι το σημείο στο οποίο επεμβαίνουμε στον πραγματικό RSP3 αλγόριθμο. Δηλαδή, ελέγχουμε τον αριθμό των κλάσεων, και αν αυτός είναι μεγαλύτερος του 2(γραμμή 6), τότε βρίσκουμε τις 2 πολυπληθέστερες κλάσεις(γραμμή 7) και στη συνέχεια υπολογίζουμε τα 2 νέα μέσα τα οποία είναι τα τεχνητά μας σημεία και τα οποία χρησιμοποιούμε στη συνέχεια του αλγορίθμου για τον υπολογισμό των αποστάσεων(γραμμή 8 και 10). Κατά τ'άλλα, ότι ακολουθεί δεν έχει καμία διαφορά με τον κανόνικο RSP3 τον οποίο ξέρουμε ήδη από το Κεφάλαιο 2.4 αυτής της εργασίας.

Όσον αφορά τον αλγόριθμο ERSP3-CC, μια πιθανή υλοποίηση του είναι η εξής:

**Algorithm 7** ERSP3-CC

**Input:** $TS$
**Output:** $CS$

1: $S \leftarrow \varnothing$
2: add($S$, $TS$)
3: $CS \leftarrow \varnothing$
4: **repeat**
5: $C \leftarrow$ επέλεξε ένα υποσύνολο $\in S$
6: **if** Ο αριθμός των κλάσεων του $C > 2$ **then**
7: Βρες τις 2 πολυπληθέστερες κλάσεις του $C$
8: Βρες τα μέσα αυτών των 2 κλάσεων, τα 2 νέα τεχνητά σημεία $p1$ και $p2$
9: **end if**
10: Χρησιμοποίησε τα 2 νέα σημεία για τον υπολογισμό των αποστάσεων
11: **if** ο αριθμός των στιγμιοτύπων του $C$ είναι $> 1$ **then**
12: **if** $C$ είναι ομοιογενή υποσύνολο **then**
13: $r \leftarrow$ βρες το μέσο στιγμιότυπο υπολογίζοντας τα μέσα των στιγμιοτύπων του $C$
14: $r.label \leftarrow$ κλάση στιγμιοτύπων του $C$
15: $CS \leftarrow CS \cup \{r\}$
16: **else**
17: $(D_1, D_2) \leftarrow$ διάσπασε το $C$ σε 2 υποσύνολα
18: add($S$, $D_1$)
19: add($S$, $D_2$)
20: remove($S$, $C$)
21: **end if**
22: **end if**
23: **until** IsEmpty($S$)
24: **return** $CS$

Ο Αλγόριθμος 7 που μόλις παρουσιάστηκε, είναι πανομοιότηπος με τον Αλγόριθμο 6, αλλά στη γραμμή 11 έχει τη συνθήκη για το αν ένα υποσύνολο έχει παραπάνω από ένα στιγμιότυπο, όπως ακριβώς γίνεται και στον απλό ERSP3. Επομένως, θα λέγαμε πως ο συγκεκριμένος αλγόριθμος αποτελεί έναν συνδιασμό των αλγορίθμων ERSP3 και RSP3-CC. Οπότε, περιμένουμε από τον συγκεκριμένο αλγόριθμο να συνδιάζει τα θετικά και των δύο, γεγονός που θα φανεί στο 4ο Κεφάλαιο, το κεφάλαιο στο οποίο παρουσιάζονται τα πειραματικά αποτελέσματα.

## 3.4 Οι αλγόριθμοι RSP3-CC2 και ERSP3-CC2

Οι αλγόριθμοι RSP3-CC2 και ERSP3-CC2, οι οποίοι αποτελούν παραλλαγές των RSP3-CC και ERSP3-CC2, έχουν ως σκοπό να βελτιώσουν την απόδοση των δύο τελευταίων, καθώς επίσης να αποτελέσουν και ένα μέτρο σύγκρισης για τους δύο τελευταίους αλγόριθμους. Οι αλγόριθμοι RSP3-CC2 και ERSP3-CC2, έχουν την ιδιαιτερότητα ότι αφού υπολογίσουν το τεχνητό μέσο σε ένα υποσύνολο, αντί να χρησιμοποιούν αυτό το τεχνητό μέσο για τον υπολογισμό αποστάσεων, όπως κάνουν οι RSP3-CC και ERSP3-CC, βρίσκουν τα δύο στιγμιότυπα τα οποία βρίσκονται σε θέση πιο κοντά στα δύο τεχνητά μέσα, με αποτέλεσμα τα στοιχεία τα οποία επιλέγονται να είναι τα δύο υπαρκτά σημεία.

Με τους δύο αυτούς αλγορίθμους, έχουμε σκοπό να δοκιμάσουμε ένα ελάχιστα διαφορετικό μονοπάτι σε

σχέση με τον RSP3-CC και ERSP3-CC. Πάραυτα όμως, επιδιώκεται η περεταίρω βελτίωση της ακρίβειας με τη χρήση των πραγματικών στιγμιοτύπων του συνόλου δεδομένων σε σχέση με τα τεχνητά σημεία. Στη συνέχεια, θα παρουσιαστούν και θα αναλυθούν οι αλγόριθμοι RSP3-CC2 και ERSP3-CC2.

Μια πιθανή υλοποίηση του RSP3-CC2, είναι η εξής:

**Algorithm 8** RSP3-CC2

**Input:** $TS$
**Output:** $CS$

1: $S \leftarrow \varnothing$
2: add($S$, $TS$)
3: $CS \leftarrow \varnothing$
4: **repeat**
5: $C \leftarrow$ επέλεξε ένα υποσύνολο $\in S$
6: **if** Ο αριθμός των κλάσεων του $C > 2$ **then**
7: Βρες τις 2 πολυπληθέστερες κλάσεις του $C$
8: Βρες τα μέσα αυτών των 2 κλάσεων, τα 2 νέα τεχνητά σημεία $a1$ και $a2$
9: **end if**
10: Βρες τα 2 πλησιέστερα σημεία $p1$ και $p2$ στα 2 σημεία $a1$ και $a2$ αντίστοιχα
11: Χρησιμοποίησε τα 2 νέα σημεία για τον υπολογισμό των αποστάσεων
12: **if** $C$ είναι ομοιογενή υποσύνολο **then**
13: $r \leftarrow$ βρες το μέσο στιγμιότυπο υπολογίζοντας τα μέσα των στιγμιοτύπων του $C$
14: $r.label \leftarrow$ κλάση στιγμιοτύπων του $C$
15: $CS \leftarrow CS \cup \{r\}$
16: **else**
17: $(D_1, D_2) \leftarrow$ διάσπασε το $C$ σε 2 υποσύνολα
18: add($S$, $D_1$)
19: add($S$, $D_2$)
20: remove($S$, $C$)
21: **end if**
22: **until** IsEmpty($S$)
23: **return** $CS$

Ο Αλγόριθμος 8, αποτελεί έναν συνδιασμό των αλγορίθμων RSP3-CC και RSP3, με την ιδιαιτερότητα ότι στη γραμμή 10 επιλέγουμε υπαρκτά σημεία για τον υπολογισμό αποστάσεων. Τα υπαρκτά αυτά σημεία, είναι τα σημεία τα οποία βρίσκονται πλησιέστερα στα 2 τεχνητά σημεία 1 και 2, τα οποία τα υπολογίζουμε στη γραμμή 8. Με τον τρόπο αυτόν, δε χρησιμοποιούμε μη υπαρκτά σημεία που προκύπτουν απλά από το σύνολο δεδομένων μας, αλλά έχουμε σημεία τα οποία υπάρχουν ήδη σαν στιγμιότυπα μέσα στο σύνολο δεδομένων μας. Αυτό, ίσως και να μας οφελήσει, αν και αυτό θα φανεί καλύτερα στο Κεφάλαιο 4, στο οποίο θα παρουσιαστούν τα αποτελέσματα από τα πειράματά μας πάνω σε πραγματικά δεδομένα, με τρόπο αναλυτικό.

Όσον αφορά τον αλγόριθμο ERSP3-CC2, μια πιθανή υλοποίηση του είναι η εξής:

**Algorithm 9** ERSP3-CC2

**Input:** $TS$
**Output:** $CS$

1: $S \leftarrow \varnothing$
2: add($S$, $TS$)
3: $CS \leftarrow \varnothing$
4: **repeat**
5: $C \leftarrow$ επέλεξε ένα υποσύνολο $\in S$
6: **if** Ο αριθμός των κλάσεων του $C > 2$ **then**
7: Βρες τις 2 πολυπληθέστερες κλάσεις του $C$
8: Βρες τα μέσα αυτών των 2 κλάσεων, τα 2 νέα τεχνητά σημεία $a1$ και $a2$
9: **end if**
10: Βρες τα 2 πλησιέστερα σημεία $p1$ και $p2$ στα 2 σημεία $a1$ και $a2$ αντίστοιχα
11: Χρησιμοποίησε τα 2 νέα σημεία για τον υπολογισμό των αποστάσεων
12: **if** ο αριθμός των στιγμιοτύπων του $C$ είναι $> 1$ **then**
13: **if** $C$ είναι ομοιογενή υποσύνολο **then**
14: $r \leftarrow$ βρες το μέσο στιγμιότυπο υπολογίζοντας τα μέσα των στιγμιοτύπων του $C$
15: $r.label \leftarrow$ κλάση στιγμιοτύπων του $C$
16: $CS \leftarrow CS \cup \{r\}$
17: **else**
18: $(D_1, D_2) \leftarrow$ διάσπασε το $C$ σε 2 υποσύνολα
19: add($S$, $D_1$)
20: add($S$, $D_2$)
21: remove($S$, $C$)
22: **end if**
23: **end if**
24: **until** IsEmpty($S$)
25: **return** $CS$

Η μοναδική διαφορά που έχει ο Αλγόριθμος 9 από τον προηγούμενό του, δηλαδή τον αλγόριθμο RSP3-CC2, είναι η γραμμή 12 στην οποία ελέγχεται το αν ο αριθμός των στιγμιοτύπων σε ένα υποσύνολο είναι μεγαλύτερος από 1.

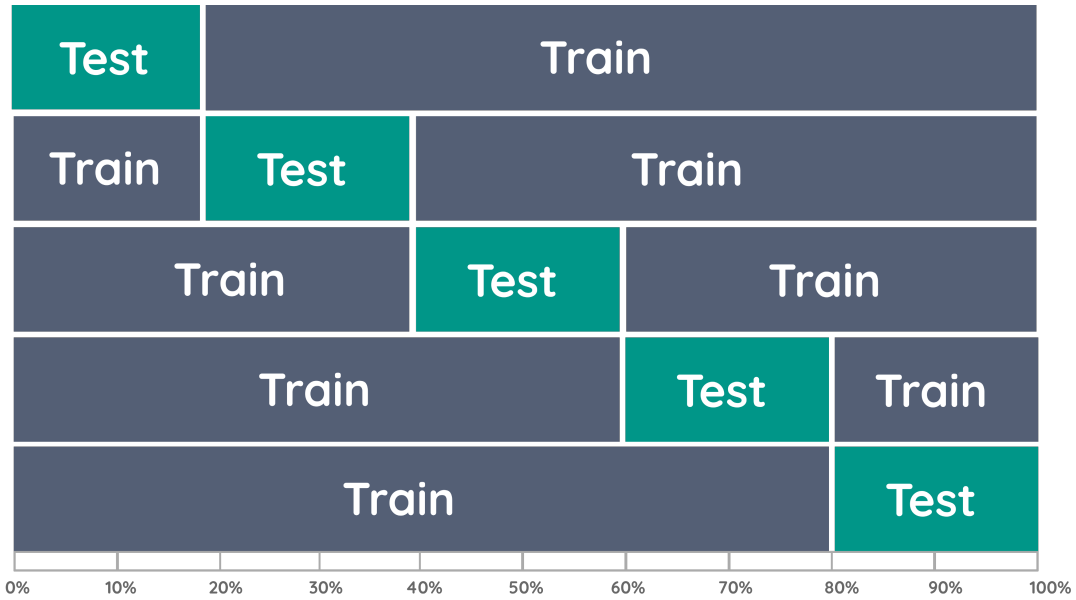
Συνολικά, αυτοί ήταν οι αλγόριθμοι οι οποίοι προτείνονται στα πλαίσια αυτής της διπλωματικής εργασίας και η επίδοση αυτών θα φανεί στο Κεφάλαιο 4, εκεί που όλοι θα συγκριθούν μεταξύ τους ως προς τις μετρικές που έχουμε θέσει σε πραγματικά δεδομένα.

## Κεφάλαιο 4ο: Πειραματική Μελέτη

Για τον έλεγχο της απόδοσης των προτεινόμενων αλγορίθμων, χρησιμοποιήσαμε ένα πλήθος από διαφορετικά σύνολα δεδομένων, καθένα από τα οποία έχουν διαφορετικές ιδιότητες. Με αυτόν τον τρόπο, πιστεύουμε πως τα αποτελέσματα τα οποία θα πάρουμε θα είναι όσο το δυνατόν πιο αληθή και ορθά τεκμηριωμένα. Στη πληθώρα αυτή των συνόλων δεδομένων, επιλέξαμε να χρησιμοποιήσουμε κάποιες μετρικές όπως την ακρίβεια(accuracy), τη ποσοστιαία μείωση του συνόλου δεδομένων(reduction rate), τον αριθμό των αποστάσεων που υπολογίστηκαν(computed distances) και τον χρόνο εκτέλεσης της CPU(CPU time).

Καταρχάς, να σημειωθεί πως στα πλαίσια αυτής της εργασίας δίνουμε το ίδιο βάρος και στις 4 αυτές μετρικές. Η ακρίβεια στη κατηγοριοποίηση, αποτελεί το ποσοστό των επιτυχημένων κατηγοριοποιήσεων από το σύνολο δεδομένων εκπαίδευσης, στο τελικό σύνολο δεδομένων. Η ποσοστιαία μείωση του συνόλου των δεδομένων, μας δίνει το επι τις 100 ποστό μείωσης του όγκου των δεδομένων, από το αρχικό σύνολο. Ο αριθμός των αποστάσεων που υπολογίστηκαν, μας δίνει τον αριθμό των υπολογισμών μεταξύ των στιγμιοτύπων που χρειάστηκε για γίνουν για να εκτελεστεί ο εκάστωτε αλγόριθμος. Τέλος, ο χρόνος εκτέλεσης της CPU, μας δίνει το χρόνο για τον οποίο ο αλγόριθμος χρειάστηκε τη CPU, δηλαδή τον χρόνο για τον οποίο κρατούσε τη CPU δεσμευμένη.

Για την εκτέλεση των πειραμάτων, χρησιμοποιήθηκε το 5-folds cross-validation, με σκοπό να αποφευχθεί το overfitting των μοντέλων. Το k-cross-validation, το οποίο ανήκει στην οικογένεια των μεθόδων Monte Carlo, έχει ως σκοπό το διαχωρισμό του συνόλου δεδομένων σε k (5 στη δική μας περίπτωση) ίσα μέρη, από τα οποία το 1 αποτελεί πάντα το σύνολο δοκιμών ενώ τα υπόλοιπα, k-1 μέρη, αποτελούν πάντα τα σύνολα δεδομένων εκπαίδευσης. Αυτή η εναλλαγή, γίνεται για k επαναλήψεις και σε κάθε επανάληψη έχουμε ένα διαφορετικό υποσύνολο το οποίο λειτουργεί ως σύνολο δοκιμών[Cross-Validation]. Ο μέσος όρος της απόδοσης των k επαναλήψεων, αποτελεί τη μέση cross-validated απόδοση του συνόλου δεδομένων. Στο Σχήμα 4.1, φαίνεται πως λειτουργεί η μέθοδος cross-validation στη πράξη.




Σχήμα 4.1: Γραφική αναπαράσταση της μεθόδου cross-validation

Για τον υπολογισμό των αποστάσεων μεταξύ των στιγμιοτύπων, χρησιμοποιήθηκε η Ευκλείδια απόσταση. Η σχέση της ευκλείδιας απόστασης, είναι ένας ήδη γνωστός και πολυχρησιμοποιημένος μαθηματικός τύπος, ο οποίος φαίνεται στη σχέση 4.1:

$$d(p,q) = d(q,p) = \sqrt{(q_1-p_1)^2 + (q_2-p_2)^2 + ... + (q_n-p_n)^2} = \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2} \tag{4.1}$$

Η Ευκλείδια απόσταση, επιλέχθηκε έναντι της Hamming ή της Manhattan απόστασης, καθώς έχει αποδειχθεί πως η Ευκλείδια απόσταση είναι πιο αποτελεσματική από τις υπόλοιπες, όταν πρόκειται για τον υπολογισμό αποστάσεων μεταξύ γραμμών δεδομένων τα οποία έχουν είτε πραγματικές αριθμιτικές τιμές, είτε ακέραιες αριθμιτικές τιμές [17]. Επίσης, οι περισσότεροι κατηγοριοποιητές που έχουν ως στόχο τη κατηγοριοποίηση στιγμιοτύπων χρησιμοποιούν την Ευκλείδια απόσταση, ως μετρική της απόστασης.

Αν οι γραμμές και οι στήλες ενός συνόλου δεδομένων χρησιμοποιούν διαφορετικές κλίμακες, είναι λογικό να γίνεται μια εξομάλυνση των δεδομένων, γνωστή και ως normalization, πριν τον υπολογισμό των αποστάσεων. Το normalization, έχει ως σκοπό τη προετοιμασία των δεδομένων, έτσι ώστε αν μια στήλη έχει τόσο μεγαλύτερες τιμές από τις υπόλοιπες, να μη δημιουργείται υποσκίαση των υπόλοιπων αριθμιτικών δεδομένων και αποστάσεων. Εδώ, θα ήταν ορθό να σημειωθεί πως το normalization δεν είναι απαραίτητο για κάθε σύνολο δεδομένων, αλλά γίνεται μόνο όταν οι τιμές των στιγμιοτύπων έχουν πολύ μεγάλο εύρος στη κλίμακά τους.

Για παράδειγμα, ας σκεφτούμε ένα σύνολο δεδομένων με δύο διαφορετικά ήδη δεδομένων. Το πρώτο, θα είναι η ηλικία του εκάστοτε στιγμιοτύπου, ένω το δεύτερο θα είναι το εισόδημά του. Η ηλικία, μπορεί να έχει ένα εύρος, ας πούμε 0-100 έτη, ένω το εισόδημα μπορεί να έχει ένα εύρος από 0-200000, ίσως και παραπάνω. Είναι σαφές πως, αν αυτές τις 2 διαφορετικές σε εύρος, αλλά αριθμιτικές τιμές τις περάσουμε από έναν αλγόριθμο, κατά πάσα πιθανότητα τα αποτελέσματα θα επηρεαστούν περισσότερο από το εισόδημα και λιγότερο από τα έτη, λόγω του μεγαλύτερου εύρους τιμών που έχει σαν ιδιότητα. Για τον λόγο αυτόν λοιπόν, επιλέγουμε να κάνουμε normalization στα σύνολα δεδομένων που κρίνουμε ότι χρειάζεται.

Όσον αφορά την εκτέλεση των πειραμάτων σε πραγματικό χρόνο, επιλέχθηκε ένας εξυπηρετητής της σχολής μας, το koraki, καθώς η υπολογιστική ισχύς και ο χρόνος που χρειάστηκε, κατέστησαν τη χρήση του ως μια πολύ καλή λύση στο πρόβλημά μας. Το koraki, είναι ενεργό όλο το 24ωρο της ημέρας, κάθε μέρα, γεγονός που μας φάνηκε άκρως απαραίτητο, καθώς κάποια σύνολα δεδομένων χρειάστηκαν παραπάνω από 8-10 ώρες, ανάλογα και τον αλγόριθμο που χρησιμοποιήθηκε. Ο εξυπηρετητής αυτός, έχει μία CPU 12 πυρήνων και μνήμη RAM 64GB. Οι διαφορετικοί πυρήνες του συστήματος χρησιμοποιήθηκαν μέσω του περιβάλλοντος linux, με τη χρήση εντολών από ένα απομακρυσμένο terminal. Η μνήμη του συστήματος βοήθησε, καθώς ο προσωπικός μου υπολογιστής που χρησιμοποιώ έχει μνήμη RAM 4GB, με την οποία δε θα μπορούσα σε καμία περίπτωση να ολοκληρώσω την εκτέλεση όλων των πειραμάτων. Για την ανάπτυξη του κώδικα, χρησιμοποιήθηκε το περιβάλλον C-Lion, το οποίο είναι ένας editor με compiler για τη γλώσσα C++.

## 4.1 Experimental Setup

Στα πλαίσια της διπλωματικής αυτής εργασίας, χρησιμοποιήσαμε κάποια σύνολα δεδομένων τα οποία περιέχουν αριθμητικές τιμές, είτε ακέραιες είτε πραγματικές σαν ιδιότητες στις πρώτες στήλες τους, ενώ η τελευταία στήλη τους αποτελεί τη κλάση του κάθε στιγμιοτύπου. Υπάρχουν σύνολα δεδομένων που έχουν 1 ή 2 κλάσεις, τα διαδυκά σύνολα, ενώ άλλα που έχουν παραπάνω από αυτές, τα σύνολα δεδομένων πολλαπλών κλάσεων. Σκοπός μας είναι η κατηγοριοποίηση αυτών των στιγμιοτύπων με όσο το δυνατόν γίνεται πιο σωστό τρόπο.

Στις επόμενες υποενότητες, περιγράφονται με τρόπο αναλυτικό τα σύνολα δεδομένων που χρησιμοποιήθηκαν, καθώς και τα ιδιαίτερα χαρακτηριστικά αυτών.

### 4.1.1 Σύνολο Δεδομένων BL

Το σύνολο δεδομένων BL, είναι μία οπτικο-ακουστική συλλογή από 238 προτάσεις οι οποίες αρθρώνονται από 17 διαφορετικούς ομιλητές. Για την ακρίβεια, κάθε ομιλητής μιλάει για περίπου 20 λεπτά και οι ηχογραφίσεις γίνονται σε 2 διαφορετικές συνεδρίες. Για τη πρώτη συνεδρία, χρησιμοποιείται μια έγχρωμη μπροστινή κάμερα και 2 μικρόφωνα, ενώ για τη δεύτερη συνεδρία, έχουμε 2 βαθμονομημένες κάμερες, 1 κάμερα βάθους και 2 μικρόφωνα. Η γλώσσα που χρησιμοποιούν οι ομιλητές είναι τα Γαλλικά.

Όπως ειπώθηκε νωρίτερα, τα δεδομένα για το σύνολο δεδομένων BL συλλέχθηκαν από 2 συνεδρίες. Τα δεδομένα από τη πρώτη συνεδρία χρησιμοποιούνται για την ανάλυση των 2D δεδομένων του στόματος και τα δεδομένα που συλλέχθηκαν από τη δεύτερη συνεδρία χρησιμοποιήθηκαν για την 3D ανάλυση των δεδομένων.

Πιο αναλυτικά, στη πρώτη συνεδρία έχουμε 238 εκφράσεις από 4 άνδρες και 4 γυναίκες. Ο ήχος συλλέχθηκε από 2 μικρόφωνα, με κάθε αρχείο ήχουν να αποτελεί μία έκφραση ενός ομιλητή, ενώ η εικόνα με τη μορφή βίντεο συλλέχθηκε από 1 μπροστινή κάμερα, με κάθε αρχείο να αποτελεί μία έκφραση ενός ομιλητή. Για τον ήχο και για το βίντεο, έχει γίνει η κατάλληλη μέριμνα ώστε να συμπίπτουν στον χρόνο, ένω όσον αφορά το προφίλ του ομιλητή, αναφέρεται μόνο η ηλικία του και το φύλο του.

Για τη δεύτερη συνεδρία, στην οποία έχουμε επίσης 238 εκφράσεις αλλά από 5 άνδρες και 4 γυναίκες, ο ήχος ηχογραφήθηκε πάλι από 2 μικρόφωνα, με κάθε αρχείο ήχου να αποτελεί μία έκφραση ενός ομιλητή. Η εικόνα με τη μορφή βίντεο, συλλέχθηκε από 2 βαθμονομημένες κάμερες και 1 κάμερα βάθους, όπου επίσης κάθε αρχείο βίντεο αντιπροσωπεύει μία πρόταση ενός ομιλητή. Το βίντεο και ο ήχος συμπίπτουν στον χρόνο, ενώ τα δεδομένα έχουν βαθμονομηθεί. Για τους ομιλητές, αναφέρεται μόνο η ηλικία και το φύλο τους.

Συνολικά όμως, το σύνολο δεδομένων BL έχει δεδομένα από 17 ομιλητές, περίπου 340 λεπτά για την 2D ανάλυση των δεδομένων, ενώ έχουν 9 ομιλητές, περίπου 180 λεπτά για τη 3D ανάλυση των δεδομένων.

Στη παρακάτω εικόνα, απεικονίζεται ο τρόπος με τον οποίο τα 33 φωνήματα της Γαλλικής γλώσσας, μεταφράζονται σε συχνότητες στο σύνολο δεδομένων BL, αλλά και στο σύνολο δεδομένων ESTER [18], το οποίο είναι παραπλήσιο του BL.

Ο σκοπός της δημιουργίας αυτής της βάσης δεδομένων, είναι η ύπαρξη οπτικο-ακουστικών δεδομένων

| IPA symbol | Occurrence | Occurrence frequency | |
|---|---|---|---|
| | | BL-Database | ESTER |
| a | 464 | 7,97 | 8,12 |
| l | 388 | 6,66 | 6,35 |
| ʁ | 370 | 6,35 | 8,20 |
| i | 326 | 5,60 | 5,74 |
| ø | 296 | 5,08 | 4,49 |
| t | 280 | 4,81 | 5,29 |
| e | 240 | 4,12 | 5,47 |
| s | 232 | 3,98 | 6,08 |
| ɛ | 222 | 3,81 | 4,77 |
| d | 206 | 3,54 | 5,13 |
| y | 196 | 3,37 | 2.08 |
| ɛ̃ | 171 | 2,94 | 1,58 |
| ɑ̃ | 165 | 2,83 | 3,28 |
| o | 164 | 2,82 | 1,11 |
| p | 163 | 2,80 | 3,52 |
| n | 162 | 2,78 | 3,00 |
| k | 162 | 2,78 | 4,04 |
| u | 154 | 2,64 | 1,73 |
| b | 154 | 2,64 | 1,17 |
| m | 147 | 2,52 | 3,01 |
| j | 132 | 2,27 | 2,07 |
| g | 126 | 2,16 | 0,62 |
| ʒ | 120 | 2,06 | 1,12 |
| v | 119 | 2,04 | 1,83 |
| ɔ̃ | 98 | 1,68 | 2,10 |
| f | 95 | 1,63 | 1,38 |
| z | 93 | 1,60 | 1,62 |
| ʃ | 81 | 1,39 | 0,50 |
| ɔ | 69 | 1,18 | 2,56 |
| w | 67 | 1,15 | 0,81 |
| œ | 55 | 0,94 | 0,51 |
| ɳ | 39 | 0,67 | 0,11 |
| ɥ | 38 | 0,65 | 0,43 |

Σχήμα 4.2: Τα 33 φωνήματα της Γαλλικής και οι συχνότητές τους

τα οποία θα μπορούν να χρησιμοποιηθούν για επιστημονικούς σκοπούς. Πέρα από τον ήχο που ηχογραφείται από τα μικρόφωνα, υπάρχουν δεδομένα και για τις κινήσεις του στόματος των ατόμων. Τα δεδομένα αυτά μάλιστα, έχουν ληφθεί από ίσο αριθμό ανδρών και γυναικών, ώστε να είναι το δυνατόν όσο πιο ορθά γίνεται και να ανταποκρίνονται στη πραγματικότητα, μιας και άντρες και γυναίκες έχουν κάποιες διαφορές όσον αφορά τα χαρακτηριστικά της φωνής και το σχήμα του προσώπου.

### 4.1.2 Σύνολο Δεδομένων KDD

Από το 1999, το σύνολο δεδομένων KDD'99 [19] είναι το πιο χρησιμοποιημένο σύνολο για την αξιολόγηση μεθόδων που σχετίζονται με την εύρεση ανωμαλιών σε μεθόδους και συστήματα. Τα δεδομένα αυτά, έχουν συλλεχθεί από τον Sholfo et al. και είναι φτιαγμένα μέσω του προγράμματος αξιολόγησης της βάσης δεδομένων DARPA98 [20]. Η βάση δεδομένων DARPA98, είναι περίπου 4GB σε μέγεθος και αποτελείται από 7 εβδομάδων δυαδικά tcpdump δεδομένα συλλεγμένα από τη κίνηση του δικτύου. Αυτό, αποτελεί την επεξεργασμένη κίνηση 5 εκατομμυρίων συνδέσεων, με μέσο όρο 100 byte τη καθεμία σε όγκο κίνησης. Το σύνολο δεδομένων KDD περιέχει περίπου 4.900.000 διανύσματα, το καθένα από τα οποία περιέχει 41 χαρακτηριστικά και έχει την ετικέτα normal ή attack, με τον τύπο της επίθεση(attack), να είναι μια συγκεκριμένη επίθεση. Οι κατηγορίες επίθεσης χωρίζονται σε 4 κατηγορίες:

1. **Denial of Service Attack (DoS):** Ο επιτιθέμενος δεσμεύει τους πόρους ενός εξυπηρετητή ή υπολογιστικού συστήματος, με αποτέλεσμα αυτό να μην είναι ικανό να εκτελέσει τις λειτουργίες τις οποίες πρέπει.
2. **User to Root Attack(U2R):** Ο επιτιθέμενος αποκτάει root δικαιώματα σε ένα σύστημα ξεκινόντας από ένας απλός χρήστης, με τη χρήση εντολών σε ένα τερματικό.
3. **Remote to Local Attack (R2L):** Ο επιτιθέμενος στέλνει πακέτα μέσω του δικτύου σε ένα σύστημα και μέσω αυτών αποκτάει πρόσβαση σε αυτό.
4. **Probing Attack:** Εδώ, ο επιτιθέμενος προσπαθεί να αποσπάσει όσες περισσότερες πληροφορίες μπορεί για ένα σύστημα , με σκοπό να μάθει τις αδυναμίες του και να τις εκμεταλλευτεί σε μεταγενέστερο χρόνο.

Αξίζει να ειπωθεί, πως τα δεδομένα του συνόλου δεδομένων επικύρωσης, δε περιέχουν αποκλειστικά ένα από αυτά τα 4 είδη επιθέσεων, αλλά περιέχουν και διάφορα άλλα τα οποία δεν εμπεριέχονται στο σύνολο δεδομένων εκπαίδευσης, ώστε αυτό να ανταποκρίνεται όσο το δυνατόν καλύτερα γίνεται σε πραγματικά δεδομένα. Πολλοί από τους ειδικούς στην ασφάλεια πληροφοριακών συστημάτων, πιστεύουν πως οι περισσότερες από τις επιθέσης που γίνονται σε συστήματα αποτελούν συνδιασμό μερικών βασικών επιθέσεων. Για τον λόγο αυτόν, επιλέχθηκαν οι 4 επιθέσεις που προαναφέρθηκαν. Τα σύνολα δεδομένων περιέχουν συνολικά 24 διαφορετικούς τύπους επιθέσεων, με 14 επιπλέον νέους μόνο για το σύνολο δεδομένων επικύρωσης.

Τα χαρακτηριστικά του KDD, μπορούν να χωριστούν σε 3 διαφορετικές κατηγορίες:

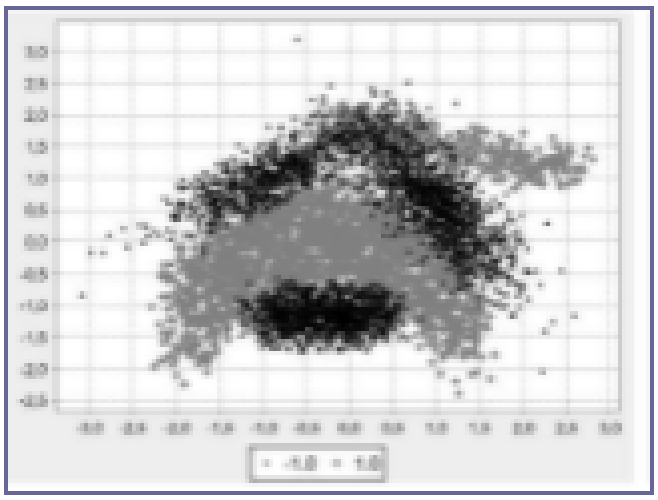
1. **Βασικά χαρακτηριστικά:** Σε αυτή τη κατηγορία, εμπεριέχονται οποιεσδήποτε πληροφορίες μπορεί κάποιος να εξάγει από μία TCP/IP σύνδεση. Τα περισσότερα από αυτά τα χαρακτηριστικά μιας τέτοιας σύνδεσης, είναι πολύ χρονοβόρα στην εύρεσή τους.
2. **Χαρακτηριστικά κίνησης:** Αυτά τα χαρακτηριστικά, συλλέγονται σε ένα συγκεκριμένο περιθώριο χρόνου που επαναλαμβάνεται και χωρίζονται σε 2 υποκατηγορίες. Η πρώτη, είναι η "same host", η οποία εξετάζει τη σύνδεση τα 2 τελευταία δευτερόλεπτα και συλλέγει στατιστικά δεδομένα για 2 συνδέσεις που έχουν τον ίδιο προορισμό. Η δεύτερη, η "same service", εξετάζει μόνο τις συνδέσεις που χρειάζονται την ίδια υπηρεσία(service), τα τελευταία 2 δευτερόλεπτα.

3. **Χαρακτηριστικά περιεχομένου:** Για τον εντοπισμό των R2L και U2R επιθέσεων, στις οποίες η επίθεση γίνεται μέσω κάποιων δεδομένων, είναι απαραίτητα κάποια χαρακτηριστικά για την αναγνώριση περίεργων προτύπων και χαρακτηριστικών μέσα στα ίδια τα πακέτα των δεδομένων, όπως είναι ο αριθμός των αποτυχημένων προσπαθειών ταυτοποίησης ενός χρήστη.

Το συγκεκριμένο σύνολο, αποτελεί το μεγαλύτερο σύνολο δεδομένων που χρησιμοποιείται στα πλαίσια αυτής της διπλωματικής, γεγονός το οποίο θα φανεί στα επόμενα κεφάλαια στους χρόνους χρήσης της CPU στο σύστημά μας.

### 4.1.3 Σύνολο Δεδομένων BN

Το σύνολο δεδομένων BN ή αλλιώς banana [21], είναι ένα σύνολο δεδομένων το οποίο χρησιμοποιείται ευρέως σε προβλήματα κατηγοριοποίησης δεδομένων. Πρόκειται για ένα τεχνητό σύνολο, στο οποίο τα δεδομένα ανήκουν σε συστάδες των οποίων το σχήμα μοιάζει με αυτό μιας μπανάνας. Στο Σχήμα 4.3, φαίνονται σχηματικά οι συστάδες των δεδομένων του συνόλου δεδομένων BN:



Σχήμα 4.3: Γραφική αναπαράσταση του BN συνόλου δεδομένων

Όσον αφορά τα δεδομένα του συγκεκριμένου υποσυνόλου, αυτά για τη πρώτη παράμετρο (At1), παίρνουν τιμές από [-3.09,2.81], ενώ για τη δεύτερη παράμετρο (At2), παίρνουν τιμές μεταξύ των [-2.39,3.19]. Οι δύο αυτές παράμετροι αποτελούν τα δεδομένα εισόδου και παριστάνουν μια θέση στον άξονα x και y αντίστοιχα. Σαν έξοδο, παίρνουμε την ετικέτα της κλάσης του εκάστοτε στιγμιοτύπου, η οποία είναι είτε -1 είτε 1.

Αξίζει να σημειωθεί, πως το συγκεκριμένο σύνολο δεδομένων είναι φτιαγμένο από επιστήμονες του χώρου της αναλυτικής των δεδομένων και πως επιλέγεται πολλές φορές από προγραμματιστές ή και άτομα του συγκεκριμένου χώρου, όταν αυτοί χρειάζονται ένα μικρό και ευέλικτο σύνολο δεδομένων(περιέχει μόνο 5300 στιγμιότυπα) για πειραματικές δοκιμές σε αλγορίθμους τους οποίους εξελίσσουν.

### 4.1.4 Σύνολο Δεδομένων LIR

Το σύνολο δεδομένων LIR (Letter Image Recognition), δημιουργήθηκε αρχικά από τον David J. Slate τον Ιανουάριο του 1991 [22]. Αρχικά, είχε χρησιμοποιηθεί για την αναγνώριση της Ολλανδικής γλώσσας, στη συνέχεια όμως εντάχθηκε σε αυτό η Αγγλική, γεγονός πολύ σημαντικό για την εξέλιξη και την αναγνώριση του συγκεκριμένου συνόλου δεδομένων. Όταν στο LIR χρησιμοποιούταν καθαρά η Ολλανδική γλώσσα, το καλύτερο ποσοστό ακρίβειας το οποίο είχε επιτευχθει σε κατηγοριοποίηση ήταν το 80 τις εκατό. Τα πρώτα 16000 στιγμιότυπα χρησιμοποιούνταν για την εκπαίδευση και τα υπόλοιπα 4000 για την επικύρωση.

Τα χαρακτηριστικά που έχει το συγκεκριμένο σύνολο δεδομένων, είναι 17 σε αριθμό και αναφέρονται εδώ:

1. lettr, κεφαλαία γράμματα (26 διαφορετικά γράμματα της αγγλικής αλφαβήτου)
2. x-box, οριζόντια θέση που έχει στο πλαίσιο (ακέραιος)
3. y-box, κάθετη θέση που έχει στο πλαίσιο (ακέραιος)
4. width, το πλάτος του πλαισίου (ακέραιος)
5. high, το ύψος του πλαισίου (ακέραιος)
6. onpix, ο συνολικός αριθμός των pixels (ακέραιος)
7. x-bar, τα μέσα x των pixels του πλαισίου (ακέραιος)
8. y-bar, τα μέσα y των pixels του πλαισίου (ακέραιος)
9. x2bar, τα μέσα βάρη των x (ακέραιος)
10. y2bar, τα μέσα βάρη των y (ακέραιος)
11. xybar, ο μέσος όρος ομοιότητας των x και y (ακέραιος)
12. x2ybr, ο μέσος όρος του x*x*y (ακέραιος)
13. xy2br, ο μέσος όρος του x*y*y (ακέραιος)
14. x-ege, η μέση απόσταση των ακμών από αριστερά στα δεξιά (ακέραιος)
15. xegvy, η ομοιότητα του x-ege και του y (ακέραιος)
16. y-ege, η μέση απόσταση των ακμών από κάτω προς τα πάνω (ακέραιος)
17. yegvx, η ομοιότητα του y-ege και του x (ακέραιος)

Για παράδειγμα, στην οριζόντια κλίμακα, μετράμε pixels από την αριστερή ακμή της εικόνας, από το κέντρο του πιο μικρού ορθογόνιου κουτιού που μπορεί να ζωγραφιστεί με όλα τα pixels που αυτό έχει μέσα, ενώ στη κάθετη κλίμακα, μετράμε pixels από κάτω προς τα πάνω. Όλα τα χαρακτηριστικά είναι ακέραιοι

αριθμοί (1-15). Τα γράμματα είναι 26, δηλαδή ξεκινάνε από το Αγγλικό γράμμα Α και τελειώνουν στο Αγγλικό γράμμα Z.

Πρόκειται για ένα σύνολο δεδομένων το οποίο είναι πολύ γνωστό στην επιστημονική κοινότητα, λόγω της ευελιξίας και της συνοχής που προσφέρει με τα δεδομένα του. Αν και σχετικά παλιό, φτιαγμένο το 1991, αποτελεί ένα από τα σύνολα τα οποία χρησιμοποιούσαν και θα συνεχίσουν να χρησιμοποιούνται στην επιστημονική κοινότητα για αρκετά χρόνια ακόμα.

### 4.1.5 Σύνολο Δεδομένων LS

Στο συγκεκριμένο σύνολο δεδομένων, εμπεριέχονται πολυ-φασματικές τιμές από pixels σε 3x3 γειτονιές σε μια εικόνα που είναι τραβηγμένη από δορυφόρο. Η κατηγοριοποίηση, σχετίζεται με την εύρεση του κεντρικού pixel σε κάθε γειτονιά. Ο στόχος είναι να επιτευχθεί αυτό, δεδομένου ότι οι τιμές είναι πολυ-φασματικές. Στη βάση που χρησιμοποιείται σε αυτή τη διπλωματική, η κλάση του κάθε pixel ορίζεται ως ένας αριθμός [23].

Τα δεδομένα του δορυφόρου, είναι μια από τις πολλές πηγές δεδομένων που υπάρχουν για μια σκηνή. Η ερμηνεία της σκηνής μέσω της ενσωμάτωσης δεδομένων διαφόρων ειδών και αναλύσεων, όπως και πολυ-φασματικών δεδομένων, δεδομένων από radar, τοπολογικά δεδομένα κλπ, περιμένουμε να έχει τεράστια σημασία στην κατανόηση χαρακτηριστικών και δεδομένων που προέρχονται από απομακρυσμένους αισθητήρες(για παράδειγμα, τα δεδομένα του συστήματος παρακολούθησης της Γης από τη NASA). Οι ήδη υπάρχουσες στατιστικοί μέθοδοι, δεν είναι κατάλληλα φτιαγμένες για την ανάλυση τέτοιων περίπλοκων δομών δεδομένων. Ας σημειωθεί πως τα δεδομένα τα οποία χρησιμοποιούνται στη συγκεκριμένη διπλωματική, είναι απλοποιημένα ώστε να είναι επεξεργάσιμα με όσο το δυνατόν καλύτερο και ευκολότερο τρόπο γίνεται. Εξαιτίας αυτού, θα είχε ενδιαφέρον η σύγκριση της απόδοσης αυτών των δεδομένων, με τα δεδομένα που προέρχονται από άλλες στατιστικές μελέτες και οπτικές.

Μία εικόνα από έναν δορυφόρο της NASA, αποτελείται από 4 ψηφιακές λήψεις της ίδιας σκηνής/εικόνας, αλλά σε διαφορετικές δέσμες συχνοτήτων. Οι 2 από αυτές, βρίσκονται σε περιοχή ορατών συχνοτήτων για το ανθρώπινο μάτι, ενώ οι άλλες 2 περιέχουν άλλου είδους δεδομένα για την εικόνα. Κάθε pixel είναι μια λέξη από 8-bit, με το 0 να είναι το μαύρο χρώμα και το 255 το άσπρο. Η χωρική ανάλυση κάθε pixel είναι 80m x 80m. Κάθε εικόνα περιέχει 2340 x 3380 τέτοια pixels.

Η βάση δεδομένων, είναι ένα μικρό υποσύνολο της συνολικής εικόνας, η οποία περιέχει 82 x 100 pixels. Κάθε γραμμή δεδομένων αντιστοιχεί σε μια γειτονιά 3x3 από pixels τα οποία εμπεριέχονται στην ευρύτερη περιοχή, δηλαδή την 82 x 100. Κάθε γραμμή επίσης, περιέχει τις τιμές των pixel και από τις 4 δέσμες συχνοτήτων (μετά από μετατροπή σε ASCII), για κάθε ένα από τα 9 pixels των 3x3 γειτονιών, καθώς και έναν αριθμό που υποδηλώνει την ετικέτα του κεντρικού pixel. Ο αριθμός είναι ένας κωδικός, ο οποίος αντιστοιχεί σε μία κλάση, όπως φαίνεται στη λίστα που ακολουθεί:

Αριθμός κλάσης:

1. red soil
2. cotton crop

3. grey soil
4. damp grey soil
5. soil with vegetation stubble
6. mixture class (all types present)
7. very damp grey soil

Να σημειωθεί, πως στο συγκεκριμένο σύνολο δεδομένων δεν υπάρχουν παραδείγματα της κλάσης 6. Αυτό, γίνεται ώστε να μη μπορέσει κάποιος να ξαναχτίσει την εικόνα ή ένα μέρος της μέσω αυτού του συνόλου που του δίνεται για εκπαιδευτικούς/ερευνητικούς σκοπούς.

Για κάθε γραμμή, τα δεδομένα για τις 4 δέσμες για κάθε pixel δίνονται από πάνω αριστερά, προς τα δεξιά και στη συνέχεια προς τα κάτω. Επομένως, τα δεδομένα που δίνονται από την ιδιότητα με αριθμό 17,18,19 και 20, αντιστοιχούν στο κεντρικό pixel. Αν επιθυμεί κάποιος, έχει τη δυνατότητα να χρησιμοποιήσει μόνο αυτές τις 4 ιδιότητες, που στην ουσία είναι αυτές του κεντρικού pixel στο οποίο εστιάζουμε κιόλας. Αυτό, επιλύει και πολλά προβλήματα τα οποία προκύπτουν με την οριοθέτηση των 3x3 γειτονιών των pixels.

Τελειώνοντας, το συγκεκριμένο σύνολο δεδομένων αξίζει να ειπωθεί πως είναι πολύ γνωστό στην επιστημονική κοινότητα, καθώς έχει χρησιμοποιηθεί και έχει συνηφέρει σε πολλές έρευνες στο παρελθόν, πράγμα το οποίο θα συνεχίσει να γίνεται και στο μέλλον.

### 4.1.6 Σύνολο Δεδομένων MGT

Το σύνολο δεδομένων MGT, ή αλλιώς Magic Gamma Telescope Data Set [23], είναι ένα σύνολο από δεδομένα τα οποία δημιουργούνται για να προσομοιώσουν υψηλής ενέργειας ακτίνες γ από ένα τηλεσκόπειο Cherenkov το οποίο βρίσκεται στο έδαφος. Τα Cherenkov τηλεσκόπια μπορούν να παρατηρήσουν υψηλής ενέργειας ακτίνες γ, μέσω της ακρινοβολίας που βγάζουν τα σωματίδια που που δημιουργούνται μέσα σε ηλεκτρομαγνητικά πεδία που δημιουργούνται και αναπαράγονται στην ατμόσφαιρα. Όλη αυτή η διαθέσιμη πληροφορία που δημιουργείται, μένει μέσω φωτονίων στη κάμερα του τηλεσκόπιου. Ανάλογα με την ενέργεια και τη ποσότητά της, μερικές χιλιάδες φωτόνια συλλέγονται από το τηλεσκόπιο σε μοτίβα, τα οποία υποδηλώνουν με κάποιο τρόπο αν τα φωτόνια αυτά προέρχονται από ακτίνες γ, δηλαδή είναι σήματα, ή αν προέρχονται από το παρασκήνιο της εικόνας.

Συνήθως, η εικόνα μετά από μια προεπεξεργασία μπορεί να αποτελέσει μια επιμηκυνσμένη συστάδα. Ο μακρύς της άξονας είναι προσανατολισμένος προς το κέντρο της κάμερας αν ο άξονας της είναι παράλληλος με τον οπτικό άξονα του τηλεσκοπίου. Μια ανάλυση των συνιστώσων γίνεται στη κάμερα του τηλεσκοπίου, μέσω της οποίας συσχετίζεται η παραλληλία των αξόνων του τηλεσκοπίου και του άξονα της εικόνας. Λόγω όλων αυτών, δημιουργούνται διάφορα ιδιαίτερα χαρακτηριστικά όσον αφορά την εικόνα και τους άξονές της, τα οποία μπορούν να αποτελέσουν κάλλιστα ένα σύνολο δεδομένων για περεταίρω ανάλυση.

Το συγκεκριμένο σύνολο δεδομένων, δημιουργήθηκε από ένα πρόγραμμα που βασίζεται σε μεθόδους Monte Carlo το 1998. Τα χαρακτηριστικά του είναι τα εξής:

Πληροφορίες χαρακτηριστικών:

1. fLength: continuous, major axis of ellipse [mm]
2. fWidth: continuous, minor axis of ellipse [mm]
3. fSize: continuous, 10-log of sum of content of all pixels [in phot]
4. fConc: continuous, ratio of sum of two highest pixels over fSize [ratio]
5. fConc1: continuous, ratio of highest pixel over fSize [ratio]
6. fAsym: continuous, distance from highest pixel to center, projected onto major axis [mm]
7. fM3Long: continuous, 3rd root of third moment along major axis [mm]
8. fM3Trans: continuous, 3rd root of third moment along minor axis [mm]
9. fAlpha: continuous, angle of major axis with vector to origin [deg]
10. fDist: continuous, distance from origin to center of ellipse [mm]
11. class: g,h, gamma (signal), hadron (background)

με g = gamma(σήμα): 12332 και h = hardon(παρασκήνιο): 6688

Για τεχνικούς λόγους, ο αριθμός των h είναι μικρότερος απ' ότι στη πραγματικότητα, καθώς σε πραγματικές συνθήκες αποτελεί τη πλειοψηφία των στιγμιοτύπων. Μια απλή κατηγοριοποίηση και η εύρεση της ακρίβειάς της, δεν έχει κάποια σημασία για το συγκεκριμένο σύνολο δεδομένων, καθώς η κατηγοριοποίηση ενός δεδομένου του παρασκηνίου ως σήμα είναι χειρότερο από τη κατηγοριοποίηση ενός σήματος ως παρασκήνιο. Για τη σύγκριση διαφορετικών κατηγοριοποιητών πρέπει να χρησιμοποιηθεί μια καμπύλη ROC. Η πιθανότητα αυτή η καμπύλη να δεχτεί ένα σήμα είναι κάτω από ένα από τα συγκεκριμένα όρια: 0.01, 0.02, 0.05, 0.2, τιμή η οποία εξαρτάται από τη ποιότητα του δείγματος και τις διαφορετικές πειραματικές συνθήκες.

### 4.1.7 Σύνολο Δεδομένων MNK

Το σύνολο δεδομένων MNK, προέρχεται από ένα σετ προβλημάτων σύγκρισης γνωστικών αλγορίθμων, γνωστά στην επιστημονική κοινότητα ως τα προβλήματα του MONK. Τα αποτελέσματα αυτής τη έρευνας συνοψίζονται στην εργασία: "The MONK's Problems - A Performance Comparison of Different Learning algorithms" [24].

Ένα σημαντικό χαρακτηριστικό της σύγκρισης αυτής που έγινε, είναι πως σε αυτή συμμετείχαν πολλοί ερευνητές, καθένας απ' τους οποίους ήταν υπεύθυνος για τον αλγόριθμο τον οποίο τέσταρε (πολλές φορές κάποιοι από αυτούς ήταν και οι ίδιοι οι δημιουργοί των αλγορίθμων αυτών). Με αυτό τον τρόπο, τα αποτελέσματα θα ήταν αμερόληπτα σε σχέση με το αν ένα άτομο ήταν υπεύθυνο για έναν συγκεκριμένο αλγόριθμο μάθησης, γεγονός το οποίο επιβεβαιώνει και το αποτέλεσμα της γενικευμένης μάθησης και των τεχνικών που χρησιμοποιούνται από τους χρήστες.

Τα προβλήματα του MONK είναι 3. Ο κλάδος των προβλημάτων είναι ίδιος και για τα 3 προβλήματα. Ένα από τα προβλήματα του MONK περιέχει θόρυβο. Για κάθε πρόβλημα, το σύνολο δεδομένων έχει διασπαστεί σε σύνολο δεδομένων εκπαίδευσης και σύνολο δεδομένων επικύρωσης.

Το σύνολο δεδομένων MNK έχει ώς εξής:

1. class: 0, 1
2. a1: 1, 2, 3
3. a2: 1, 2, 3
4. a3: 1, 2
5. a4: 1, 2, 3
6. a5: 1, 2, 3, 4
7. a6: 1, 2
8. Id: (A unique symbol for each instance)

Το συγκεκριμένο σύνολο δεδομένων, έχει επεξηγηθεί πλήρως στις έρευνες [25] και [26] και έχει χρησιμοποιηθεί σε ποικίλες άλλες έρευνες και πειράματα. Πρόκειται για ένα σύνολο το οποίο χρησιμοποιείται στην επιστημονική κοινότητα και θα συνεχίσει να χρησιμοποιείται στο μέλλον.

### 4.1.8 Σύνολο Δεδομένων PD

Το συγκεκριμένο σύνολο δεδομένων, ονομάζεται pendigits και αποτελείται από 250 χειρόγραφα δείγματα ψηφίων από 44 διαφορετικούς ανθρώπους. Τα 30 δείγματα από το σύνολο των 44 χρησιμοποιείται για εκπαίδευση, cross-validation και για ελεγχόμενο testing, ενώ τα άλλα 14 που απομένουν χρησιμοποιούνται για επικύρωση του συνόλου εκπαίδευσης. Η συγκεκριμένη βάση δεδομένων, υπάρχει επίσης σε UNIPEN φορματ [23].

Χρησιμοποιείται ένα WACOM PL-100V tablet, με ιδιαίτερη ευαισθησία στην αφή και με ενσωματωμένη LCD οθόνη και στυλό αφής. Το σημείο για γραφή και η οθόνη βρίσκονται στο ίδιο σημείο. Αυτά, έχουν ενσωματωμένο έναν Intel 486 PC, το οποίο μας επιτρέπει να συλλέγουμε οτιδήποτε χειρόγραφο εισάγεται στην οθόνη. Το tablet στέλνει συντεταγμένες τύπου x και y και επίπεδα πίεσης σε ένα συγκεκριμένο εύρος τιμών και χρονικά περιθώρια των 100ms.

Οι άνθρωποι που γράφουν πάνω στο tablet, έχουν ως εργασία να γράψουν 250 ψηφία σε τυχαία σειρά μέσα σε πλαίσια 500 x 500 tablet pixels. Αυτοί, παρακολουθούνται μόνο κατά τις πρώτες στιγμές του έργου τους. Κάθε οθόνη περιέχει 5 πλαίσια και τα ψηφία που πρέπει να γράψουν φαίνονται από πάνω. Στη συνέχεια, του λένε να γράψουν μόνο μέσα στα πλαίσια και πουθενά αλλού. Αν δε τους αρέσει κάτι που γράψαν, έχουν την επιλογή με ένα κουμπί να διαγράψουν οτιδήποτε βρίσκεται μέσα στο πλαίσιο. Τα πρώτα 10 ψηφία δεν λαμβάνονται υπ'όψιν, καθώς οι περισσότεροι χρήστες δεν είναι εξοικειωμένοι με τέτοιες συσκευές, οπότε χρειάζονται λίγο χρόνο να τις συνηθίσουν.

Στο σύνολο που χρησιμοποιείται για τη παρούσα εργασία, χρησιμοποιείται το x και y σαν συντεταγμένες θέσεων. Οι τιμές της πίεσης του στυλό στην οθόνη δε λαμβάνονται υπ'όψιν. Πρώτα, κανονικοποιείται το σύνολο των ανεπεξέργαστων δεδομένων, για να μην επηρεάζεται από αλλαγές σε κλίμακα και μεταβολές στην εικόνα. Τα ακατέργαστα δεδομένα που λαμβάνονται από το tablet περιέχουν τιμές ακεραίων μεταξύ του 0 και του 500(σε tablet pixels). Οι νέες συντεταγμένες έχουν τιμές, τέτοιες ώστε να βρίσκονται μεταξύ του 0 και το 100. Συνήθως, το x μένει σε αυτό το εύρος, καθώς οι περισσότερες τιμές των ψηφίων δεν είναι πολύ μεγάλες σε πλάτος.

Για να εκπαιδευτούν και να επικυρωθούν οι κατηγοριοποιητές μας, χρειάζεται να αναπαριστήσουμε τα ψηφία σαν συνεχές τιμές διανυσμάτων. Μια τεχνική που χρησιμοποιείται συχνά και έχει συνήθως καλά αποτελέσματα είναι η επαναδειγματοληψία των x και y σημείων. Η χρονική επαναδειγματοληψία ή η χωρική επαναδειγματοληψία μπορούν να χρησιμοποιηθούν για αυτόν τον σκοπό. Τα ανεπεξέργαστα δεδομένα, είναι ήδη στοιχισμένα σωστά ως προς τον χρόνο, αλλά ως προς την απόσταση μεταξύ τους υπάρχουν διακυμάνσεις. Προηγούμενες έρευνες, έχουν δείξει πως η χωρική επαναδειγματοληψία έχει δείξει πολύ καλύτερα αποτελέσματα, καθώς τα σημεία είναι διαχωρισμένα μεταξύ τους με καλύτερο τρόπο. Ο αλγόριθμος επαναδειγματοληψίας του συγκεκριμένου συνόλου, χρησιμοποιεί απλή γραμμική παρεμβολή μεταξύ ένος ζεύγους σημείων. Τα ψηφία αυτά, αναπαριστώνται ως μία σειρά από T σημεία x και y, με καθορισμένο κενό ως προς την απόσταση των τόξων τους.

Τα χαρακτηριστικά του συγκεκριμένου συνόλου δεδομένων είναι τα εξής: Όλα τα στιγμιότυπα εισόδου είναι ακέραιοι στο διάστημα 0 με 100 και το τελευταίο χαρακτηριστικό κάθε στιγμιοτύπου είναι ένας αριθμός μεταξύ του 0 και 9, ο οποίος υποδηλώνει κωδικό κλάσης.

### 4.1.9 Σύνολο Δεδομένων PH

Το σύνολο δεδομένων PH ή αλλιώς phoneme, χρησιμοποιείται σε προβλήματα κατηγοριοποίησης και τα στιγμιότυπά του έχουν 5 χαρακτηριστικά, ενώ στον αριθμό τους είναι 5404. Γενικά στο ίντερνετ, υπάρχουν πολλά σύνολα δεδομένων τα οποία βασίζονται σε φωνήματα. Στα πλαίσια όμως αυτής της εργασίας, ασχοληθήκαμε με το συγκεκριμένο, το οποίο βρίσκεται στο keel dataset repository [21]. Το συγκεκριμένο σύνολο δεδομένων, έχει 2 κλάσεις σαν στόχους, ενώ τα χαρακτηριστικά του είναι πραγματικοί αριθμοί. Ο πίνακας των χαρακτηριστικών φαίνεται στο Σχήμα 4.4.

Οι ήχοι, όπως παρατηρείται αντιστοιχούν σε ένα εύρος τιμών ανάλογα με το χαρακτηριστικό το οποίο αντιστοιχούν. Τα φωνήματα του συνόλου PH, έχουν ώς εξής: sh για το she, dcl για το dark, iy για το φωνήεν του she, aa για ττο φωνήεν του dark και ao για το πρώτο φωνήεν του water.

Ο στόχος του συγκεκριμένου συνόλου δεδομένων, είναι η αναγνώριση των ρινικών και τον στοματικών ήχων. Οι πρώτοι αντιστοιχούν στη κλάση 0, ενώ οι δεύτεροι στη κλάση 1. Ο διαμοιρασμός των στιγμιοτύπων είναι 3.818 για τη κλάση 0 και 1586 για τη κλάση 2.

### 4.1.10 Σύνολο Δεδομένων SH

Το σύνολο δεδομένων Statlog(Shuttle), δημιουργήθηκε αρχικά με σκοπό να φτιαχτούν καθολικοί κανόνες για την απόφαση που χρειάζεται να πάρει ένα σύστημα αυτόματης προσγείωσης για το αν πρέπει να

| Attribute | Domain |
|---|---|
| Aa | [-1.7, 4.107] |
| Ao | [-1.327, 4.378] |
| Dcl | [-1.823, 3.199] |
| Iy | [-1.581, 2.826] |
| Sh | [-1.284, 2.719] |
| Class | {0, 1} |

Σχήμα 4.4: Η κατανομή των κλάσεων του PD συνόλου δεδομένων στον χώρο

το προσγειώσει το αεροσκάφος ο πιλότος ή το σύστημα μόνο του. Για να το αποφασίσει αυτό, πρέπει να ξέρει ποιο είναι το είδος του σκάφους για να ξέρει πως να διαχειριστεί τη κατάσταση [21]. Το shuttle σύνολο δεδομένων περιέχει 9 χαρακτηριστικά τα οποία είναι αριθμητικά. Επίσης, έχει 7 πιθανές τιμές για κάθε μία από τις 7 πιθανές κλάσεις που μπορεί να έχει ένα στιγμιότυπο. Περίπου το 80 τις εκατό των δεδομένων ανήκει στη κλάση 1. Έτσι προκύπτει πως η ακρίβεια από προεπιλογή είναι περίπου 80 τις εκατό. Ο στόχος, είναι να πιάσουμε το 99 με 99.9 τις εκατό [23]. Τα παραδείγματα στο αυθεντικό σύνολο δεδομένων ήταν κατανεμημένα σε χρονική σειρά, η οποία θα μπορούσε να έχει σχέση με τη κατηγοριοποίηση. Ωστόσο, από το αυθεντικό σύνολο διαλέχθηκαν στη τύχη δεδομένα για σκοπούς εκπαίδευσης και για επικύρωση.

Τα χαρακτηριστικά του SH:

1. : Rad Flow
2. : Fpv Close
3. : Fpv Open
4. : High
5. : Bypass
6. : Bpv Close
7. : Bpv Open

Ενώ οι πιθανές 7 κλάσεις που μπορεί να έχει ένα στιγμιότυπο είναι οι εξής:

| Attribute | Domain |
|---|---|
| A1 | [27,126] |
| A2 | [-4821,5075] |
| A3 | [21,149] |
| A4 | [-3939,3830] |
| A5 | [-188,436] |
| A6 | [-26739,15164] |
| A7 | [-48,105] |
| A8 | [-353,270] |
| A9 | [-356,266] |
| Class | {1, 2, 3, 4, 5, 6, 7} |

Σχήμα 4.5: Η κατανομή των κλάσεων του PD συνόλου δεδομένων στον χώρο

### 4.1.11 Σύνολο Δεδομένων TXR

Το σύνολο δεδομένων Texture, χρησιμοποιείται κυρίως σε προβλήματα κατηγοριοποιήσης και αποτελείται από 40 χαρακτηριστικά τα οποία είναι πραγματικοί αριθμοί. Έχει 5500 στιγμιότυπα τα οποία ανήκουν σε 11 κλάσεις. Ο στόχος αυτού του συνόλου δεδομένων είναι ο διαχωρισμός 11 διαφορετικών επιφανειών (Grass lawn, Pressed calf leather, Handmade paper, Raffia looped to a high pile, Cotton canvas, κ.α.), με κάθε pixel να χαρακτηρίζεται από 40 χαρακτητιστικά τα οποία υπολογίστηκαν από 4 σημεία του ορίζοντα: στις 0 μοίρες, στις 45,90 και 135 μοίρες [21].

Στο Σχήμα 4.6, φαίνονται αναλυτικά τα χαρακτηριστικά των στιγμιοτύπων του TXR, όπως και οι τιμές που μπορούν να πάρουν αυτά για τη κλάση τους. Τα χαρακτηριστικά, είναι ονομασμένα από το A1 μέχρι το A40 και οι τιμές που παίρνουν είναι μεγαλύτερες του -1 και μικρότερες του 2. Η τιμή που παίρνει η κλάση είναι μεγαλύτερη του 2 και μικρότερη του 14, χωρίς να εμπεριέχονται όλοι οι ενδιάμεσοι αριθμοί. Πιο αναλυτικά, φαίνονται λεπτομερώς τα χαρακτηριστικά με τις τιμές που παίρνουν, στο σχήμα που ακολουθεί:

| Attribute | Domain | Attribute | Domain | Attribute | Domain |
|---|---|---|---|---|---|
| A1 | [-1.45, 0.774] | A15 | [-0.943, 0.164] | A28 | [-1.154, 0.422] |
| A2 | [-1.2, 0.33] | A16 | [-0.994, 0.036] | A29 | [-1.132, 0.392] |
| A3 | [-1.31, 0.344] | A17 | [-1.172, 0.02] | A30 | [-1.422, 0.472] |
| A4 | [-1.11, 0.588] | A18 | [-1.017, 0.115] | A31 | [-1.45, 0.774] |
| A5 | [-1.053, 0.439] | A19 | [-1.004, 0.083] | A32 | [-1.179, 0.565] |
| A6 | [-1.003, 0.452] | A20 | [-1.18, 0.439] | A33 | [-1.147, 0.675] |
| A7 | [-1.208, 0.525] | A21 | [-1.45, 0.774] | A34 | [-1.123, 0.313] |
| A8 | [-1.08, 0.398] | A22 | [-1.228, 0.596] | A35 | [-1.015, 0.34] |
| A9 | [-1.057, 0.437] | A23 | [-1.341, 0.446] | A36 | [-1.03, 0.156] |
| A10 | [-1.258, 0.355] | A24 | [-1.177, 0.688] | A37 | [-1.253, 0.09] |
| A11 | [-1.45, 0.774] | A25 | [-1.137, 0.41] | A38 | [-1.097, 0.194] |
| A12 | [-1.083, 0.372] | A26 | [-1.11, 0.373] | A39 | [-1.076, 0.202] |
| A13 | [-1.119, 0.635] | A27 | [-1.239, 0.612] | A40 | [-1.215, 0.465] |
| A14 | [-1.018, 0.157] | Class | {2, 3, 4, 9, 10, 7, 6, 8, 12, 13, 14} | | |

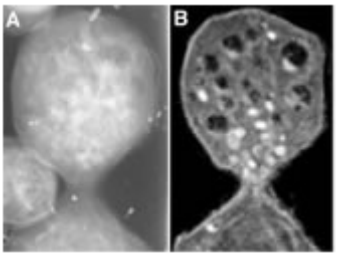
Σχήμα 4.6: Τα χαρακτηριστικά των στιγμιοτύπων του συνόλου TXR

### 4.1.12 Σύνολο Δεδομένων YS

Το σύνολο δεδομένων Yeast, περιέχει πληροφορίες σχετικά με τα κύτταρα κάποιων ζυμομυκήτων. Ο στόχος σε αυτή τη βάση δεδομένων, είναι να βρεθεί το είδος του μύκητα που βρίσκεται σε ένα συγκεκριμένο μέρος του κυττάρου, μεταξύ 10 διαφορετικών επιλογών.

Οι ζυμομύκητες, είναι μονοκύτταροι μύκητες και το επιστημονικό τους όνομα είναι ”Saccharonyces cerevisiae”. Οι οργανισμοί αυτοί συχνά χρησιμοποιούνται για τη ζύμωση της ζάχαρης για τη δημιουργία εθανόλης, όπως επίσης και στη ζύμωση του σίττου για τη παρασκευή μπίρας και άλλων αλκοολούχων ποτών [27]. Θεωρείται, επιπλέον συμπλήρωμα διατροφής από πολλούς καθώς η σύστασή του είναι πλούσια σε πρωτείνες και σε μια ποικιλία βιταμινών.

Η συγκεκριμένη συλλογή δεδομένων βρίσκεται συνεχώς σε εξέλιξη, και αποτελεί απόγονο μια παλαιότερης βάσης, η οποία περιέχει πρωτείνες και λέγεται SWISS-PROT. Το συγκεκριμένο σύνολο δεδομένων αποτελείται από 1484 στιγμιότυπα και κάθε στιγμιότυπό του έχει 8 στήλες οι οποίες είναι πραγματικοί αριθμοί [21]. Στο σχήμα 4.7 που ακολουθεί, φαίνεται η εικόνα 2 ζυμομυκήτων όπως αυτοί απεικονίζονται στα δεδομένα μας, ενώ στη συνέχεια αναλύονται τα χαρακτηριστικά του συνόλου δεδομένων YS.



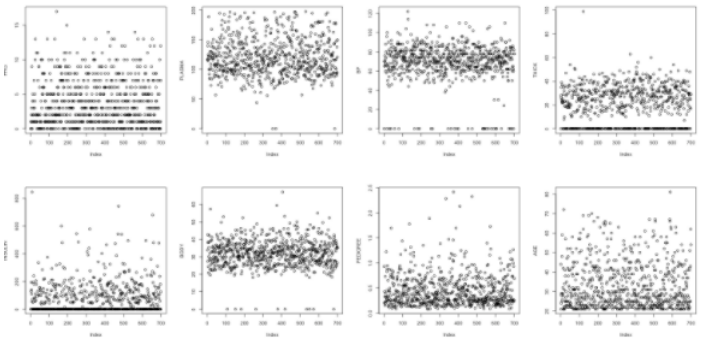

Σχήμα 4.7: Ζυμομύκητες του συνόλου δεδομένων YS

Τα χαρακτηριστικά του συνόλου YS είναι τα εξής [23]:

1. Sequence Name: Αριθμός για τη συσχέτιση με τη SWISS-PROT βάση δεδομένων.
2. mcg: Η μέθοδος McGeoch’s για την αναγνώριση σειρών από σήματα.
3. gvh: Η μέθοδος von Heijne’s για την αναγνώριση σειρών από σήματα.
4. alm: Η βαθμολογία του προγράμματος πρόβλεψης του διαστήματος της ALOM μεμβράνης
5. mit: Η βαθμολογία της ανάλυσης των περιεχομένων των αμινοξέων της N-terminal περιοχής της μιτοχόνδριας και μη μιτοχόνδριας πρωτείνης
6. erl: Η παρουσία του ”HDEL” συμβολοσειράς (λειτουργεί σαν σήμα για τη διαφύλαξη του ενδοκυτταρικού ιστού). Δυαδική τιμή.
7. pox: Σήματα της πρωτείνης(Peroxisomal protein) στο C-terminus.
8. vac: Η βαθμολογία της ανάλυσης του περιεχομένου των αμινοξέων 2 συγκεκριμένων πρωτεινών (vacuolar and extracellular proteins)
9. nuc: Η βαθμολογία της ανάλυσης των σημάτων από νουκλεικές και μη νουκλεικές πρωτείνες.

### 4.1.13 Σύνολο Δεδομένων PM

Το Pima Indians Diabetes σύνολο δεδομένων, δημιουργήθηκε από το Εθνικό Ινστιτούτο για τον Διαβήτη, Πεπτικών και Νεφρικών παθήσεων. Ο στόχος του συγκεκριμένου σύνολου είναι η πρόβλεψη της διάγνωσησης σχετικά με το αν ένας ασθενής έχει ή δεν έχει διαβήτη, βάση κάποιων διαγνωστικών γενικότερων στοιχείων τα οποία περιέχονται στο σύνολο δεδομένων [23]. Για τη συλλογή αυτών τον δεδομένων, χρησιμοποιήθηκαν μερικές παράμετροι πιο συγκεκριμένοι, οι οποία αναφέρονται στο σύνολο. Για παράδειγμα, όλοι οι ασθενείς είναι γυναίκες ηλικίας άνω των 21 και είναι απόγονοι του Ινδικού λαού Pima. Το PM σύνολο δεδομένων περιέχει 8 χαρακτηριστικά, τα οποία είναι πραγματικοί αριθμοί ενώ τα στιγμιότυπα του σε αριθμό είναι 768 και οι κλάσεις είναι 2(καθώς κάποιος ή θα νοσεί από διαβήτη ή οχι).



Σχήμα 4.8: Ο διαχωρισμός των χαρακτηριστικών του PM συνόλου δεδομένων στον χώρο

Στο Σχήμα 4.9, φαίνεται ο τρόπος με τον οποίο τα χαρακτηριστικά του συγκεκριμένου συνόλου δεδομένων, κατανέμονται στον χώρο. Στη συνέχεια, θα αναλυθούν τα χαρακτηριστικά που έχει το σύνολο δεδομένων PM [21].

1. Preg = Πόσες φορές έχει γεννήσει.
2. Plas = Συγκέντρωση Γλυκόζης σε 2ωρο στοματικό τεστ.
3. Pres = Πίεση αίματος (mm Hg)
4. Skin = Πάχος τρικέφαλου (mm)
5. Insu = 2-ωρών ινσουλήνη στο αίμα (mu U/ml)
6. Mass = Δείκτης μάζας σώματος (βάρος σε κιλά/(ύψος σε m στο τετράγωνο)
7. Pedi = Συνάρτηστη pedigree διαβήτη.
8. Age = Ηλικία (έτη)

### 4.1.14 Σύνολο Δεδομένων TN

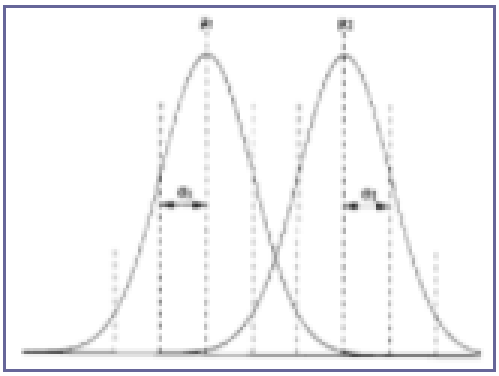
Το σύνολο δεδομένων TN, ή αλλιώς twonorm dataset, είναι μια υλοποίηση του twonorm παραδείγματος του Leo Breinman [28]. Πρόκειται για ένα πρόβλημα 20 διαστάσεων, με 2 κλάσεις προς κατηγοριοποίηση. Κάθε κλάση απεικονίζεται σχηματικά με κανονική κατανομή σε στυλ γραφήματος. Κάθε στιγμιότυπο, έχει 20 χαρακτηριστικά τα οποία είναι πραγματικοί αριθμοί, ενώ σε αριθμό τα στιγμιότυπα είναι

7400 και οι κλάσεις 2. Στο σχήμα 4.10, φαίνονται τα χαρακτηριστικά του TN συνόλου δεδομένων καθώς και οι τιμές που μπορούν να πάρουν [21]:

| Attribute | Domain | Attribute | Domain |
|---|---|---|---|
| A1 | [-4.2497, 3.5782] | A11 | [-3.9204, 4.0953] |
| A2 | [-3.9606, 4.3451] | A12 | [-3.8262, 4.3025] |
| A3 | [-4.4798, 4.4039] | A13 | [-3.9451, 3.7817] |
| A4 | [-3.8891, 4.2589] | A14 | [-3.8516, 4.1238] |
| A5 | [-3.9099, 5.6395] | A15 | [-3.6978, 3.5485] |
| A6 | [-4.0095, 3.9901] | A16 | [-3.6976, 4.2439] |
| A7 | [-3.7334, 4.4522] | A17 | [-3.9428, 3.7233] |
| A8 | [-4.5744, 3.8648] | A18 | [-4.0653, 4.0557] |
| A9 | [-3.8136, 4.254] | A19 | [-4.1218, 4.3238] |
| A10 | [-4.1999, 4.0829] | A20 | [-3.7831, 4.279] |
| Class | {0, 1} | | |

Σχήμα 4.9: Τα χαρακτηριστικά του TN dataset

Η κλάση 1, έχει σαν μέσο όρο το (a,a,..a) ενώ η κλαση 2 έχει σαν μέσο όρο το (-a,-a,..-a), όπου το a = 2/sqrt(20). Η αναφορά που δημοσίευεσε ο Breiman αναφέρει πως σε θεωρητικό επίπεδο η λανθασμένη κατηγοριοποίηση συμβαίνει κατα 2.3 τις εκατό. Πρακτικά όμως, σε 300 παραδείγματα εκπαίδευσης με CART βρήκε ένα σφάλμα της τάξης του 22.1%. Τέλος, στο Σχήμα 4.11 που ακολουθεί φαίνεται η κατανομή των 2 κλάσεων του συνόλου σε στυλ γραφήματος.

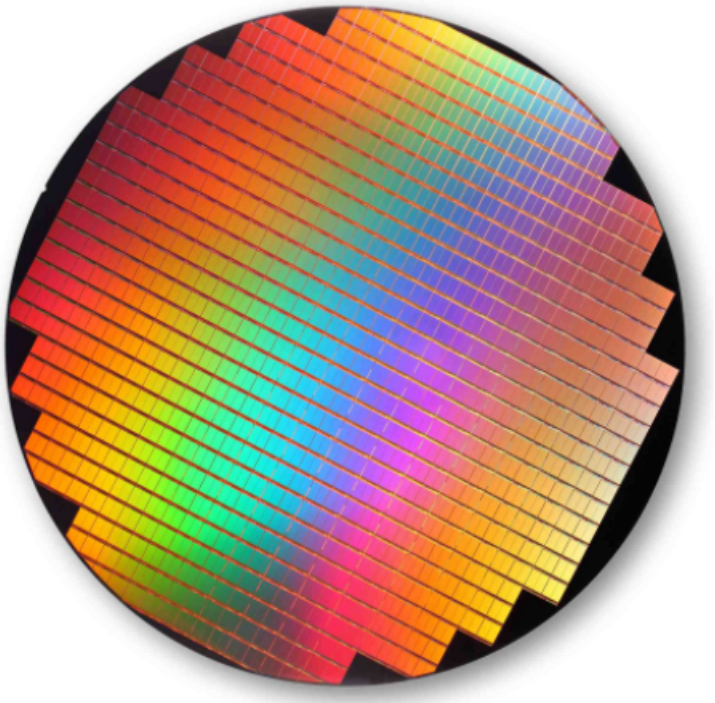


Σχήμα 4.10: Η κατανομή των 2 κλάσεων του TN σε στυλ γραφήματος

### 4.1.15 Σύνολο Δεδομένων WF

Το σύνολο δεδομένων WF, ή αλλιώς Wafer dataset, δημιουργήθηκε από τον R. Olszewski στα πλαίσια της εργασίας του ”Generalized feature extraction for structural pattern recognition in time-series data at Carnegie Mellon University, 2001” [29]. Τα wafer data, στα Ελληνικά είναι δεδομένα σε σχήμα ”γκοφρέτας”, ονομάζονται έτσι λόγου του κυψελοειδούς σχήματος που έχουν όταν αναπαριστώνται σαν εικόνα. Αυτά, σχετίζονται με τη μικροηλεκτρική και πιο συγκεκριμένα με τη δημιουργία ημιαγωγών. Μια συλλογή από εσωτερικές μετρήσεις από διάφορους αισθητήρες κατά την χρήση των ημιαγωγών αποτελούν το wafer σύνολο δεδομένων. Κάθε εγγραφή στο συγκεκριμένο σύνολο δεδομένων περιέχει τις μετρήσεις που συλλέχθηκαν από εναν αισθητήρα μέσω ενός εργαλείου σε ένα μικρό μέρος της κυψέλης τη φορά. Οι 2 κλάσεις είναι η ”normal” και ”abnormal”. Υπάρχει μια μεγάλη ανομοιομορφία μεταξύ ”normal” και ”abnormal” (10.7 τις εκατό είναι τα ”abnormal” στο σύνολο εκπαίδευσση και 12.1 τις εκατό στο σύνολο επικύρωσης).

Το συνολικό μέγεθος του συνόλου δεδομένων εκπαίδευσης είναι 1000 στιγμιότυπα, ενώ το σύνολο δεδομένων επικύρωσης είναι 6164 στιγμιότυπα. Το μέγεθος των στιγμιοτύπων είναι 152 σε μήκος. Στο Σχήμα 4.12 που ακολουθεί, απεικονίζονται τα δεδομένα σε στυλ ”γκοφρέτας” (wafer), τα οποία στην ουσία αναπαριστούνε τα δεδομένα που έχουν συλλεχθεί από τους ημιαγωγούς. Αξίζει να σημειωθεί, πως η μέγιστη ακρίβεια που μπορεί να επιτευχθεί στο συγκεκριμένο σύνολο δεδομένων είναι 99.98 τις εκατό, και αυτό οφείλεται στην κατασκευή του.



Σχήμα 4.11: Το σύνολο δεδομένων WF

### 4.1.16 Σύνολο Δεδομένων EEG

Η κατηγοριοποίηση στο σύνολο δεδομένων EEG (EEG Eye State Data Set) βασίζεται συχνά σε μια χρονική σειρά κατά την οποία φτάνει η πληροφορία. Συστήματα και αλγόριθμοι αναγνώρισεις προτύπων χρησιμοποιούνται ευραίως για τα EEG σύνολα δεδομένων [30]. Για παράδειγμα, ο Osamu Fukuda et al., χρησιμοποίησε μία μίξη μεταξύ ενός Γκαουσιανού συστήματος και ενός νευρωνικού δικτύου για να κάνει τη κατηγοριοποίηση σε ένα EEG σύνολο δεδομένων [31]. Επίσης, ο Yeo et al. χρησιμοποίησε με επιτυχία SVMs (Support Vector Machines), για την αναγνώριση της υπνηλίας του οδηγού μέσω του ματιού του [32]. Αυτές, και άλλες δουλείες αποδεικνύουν σε μας πως η μηχανική μάθηση και οι στατιστικές μέθοδοι είναι απαραίτητοι για την επίλυση προβλημάτων κατηγοριοποίησης EEG συνόλων δεδομένων.

Το EEG σύνολο δεδομένων, αποτελέιται από 14980 στιγμιότυπα και έχει 15 χαρακτηριστικά, 14 από τα οποία είναι συγκεκριμένες τιμές για το EEG σύνολο, ενώ η 1 τελευταία μας δείχνει τη κατάσταση του ματιού. Όλες οι τιμές αυτές είναι πραγματικοί ή ακέραιοι αριθμοί [23]. Όλα αυτά τα δεδομένα προέρχονται από μια συνεχόμενη μέτρηση EEG μέσω του Emotiv EEG Neuroheadset. Η διάρκεια της μέτρησης ήταν 117 δευτερόλεπτα. Η κατάσταση του ματιού αναλυόταν μέσω μιας κάμερας κατά τη διάρκεια της μέτρησης, και προστέθηκε στη συνέχεια στα αρχεία, έπειτα από την ανάλυση των πλάνων του βίντεο. Το ’1’ μας δείχνει πως το μάτι ήταν κλειστό και το ’0’ πως το μάτι είναι ανοιχτό. Όλες οι τιμές είναι σε χρονολογική μορφή με τη πρώτη μέτρηση, δηλαδή αυτή που βρίσκεται πάνω πάνω στο σύνολο των δεδομένων.

## 4.2 Πειραματικά αποτελέσματα

Τα αποτελέσματα των πειραμάτων τα οποία θα δείτε στη συνέχεια, αποτελούν ένα σύνολο από διαδοχικές εκτελέσεις των αλγορίθμων των οποίων δημιουργήσαμε και προτείναμε στα πλαίσια αυτής της εργασίας. Αυτή η διαδικασία, επαναλαμβάνεται για κάθε ένα από τα σύνολα δεδομένων τα οποία έχουμε αναλύσει, με σκοπό να μπορέσουμε να φτάσουμε σε ανιδιοτελή συμπεράσματα σχετικά με την αποτελεσματικότητα και τη χρηστικότητα των αλγορίθμων μας.

Έτσι, για κάθε μία από τις υποενότητες που θα ακολουθήσουν, θα παρουσιάζονται αρχικά τα αποτελέσματα του κλασσικού KNN αλγόριθμου και έπειτα των RSP3 και ERSP3 αλγορίθμων σε τρεις διαφορετικούς πίνακες. Αυτό, θα γίνει για να μπορέσουμε να εξετάσουμε με πιο εύκολο τρόπο τις διαφορές των αλγορίθμων βάση των μετρικών μας (ACC = ακρίβεια, RR = ποσοστιαία μείωση δεδομένων, DIST = αριθμός υπολογισμένων αποστάσεων, CPU = χρόνος εκτέλεσης CPU).

Στη συνέχεια, θα υπάρχει επημέρους σχολιασμός για το κάθε σύνολο δεδομένων ξεχωριστά όσον αφορά τα αποτελέσματά του και τέλος θα γίνει μια σύγκριση όλων των αποτελεσμάτων από όλα τα σύνολο στα Κεφάλαιο 4.3.

### 4.2.1 Πειραματικά αποτελέσματα Συνόλου Δεδομένων BL

Στους πίνακα 4.1 που ακολουθεί, φαίνονται τα αποτελέσματα του KNN στο σύνολο δεδομένων BL. Όπως θα φανεί στη συνέχεια, ο KNN θα έχει καλύτερη ακρίβεια από όλους τους RSP3 αλγορίθμους, αλλά δε θα ισχύει το ίδιο για τους ERSP3 αλγορίθμους. Όσον αφορά τις αποστάσεις που υπολογίστηκαν, συνολικά φαίνεται να είναι πολύ λιγότερες από αυτές των RSP3 και ERSP3, αλλά να είναι και αρκετά περισσότερες από αυτές των RSP3-RND, RSP3-CC, RSP3-CC2 όπως και των αντίστοιχων ERSP3 αλγορίθμων. Τέλος, το CPU time του KNN είναι πολύ χαμηλότερο από αυτά όλων των RSP3 και ERSP3 αλγορίθμων.

Στον πίνακα 4.2, απεικονίζονται τα πειραματικά αποτελέσματα των RSP3 αλγορίθμων. Το παράδοξο εδώ, είναι πως την καλύτερη ακρίβεια την πέτυχε ο RSP3-RND, ο οποίος όμως έχει πολύ χαμηλό ποστοστό μείωσης δεδομένων σε σχέση με τους άλλους τρεις αλγορίθμους. Όσον αφορά το CPU time και τις αποστάσεις που υπολογίστηκαν, η λογική λέει πως στον απλό RSP3 οι αριθμοί θα είναι πολύ μεγαλύτεροι από τους υπόλοιπους αλγορίθμους, καθώς ο απλός RSP3 υπολογίζει τις αποστάσεις μεταξύ όλων των στιγμιοτύπων του συνόλου δεδομένων. Ο RSP3-RND, αν και έχει τη καλύτερη ακρίβεια, πέρα από το ποστοστό μείωσης δεδομένων, υστερεί και σε λιγότερες υπολογισμένες αποστάσεις και CPU time σε σχέση με τους RSP3-CC και RSP3-CC2.

Στον πίνακα 4.3, τα αποτελέσματα όσον αφορά την ακρίβεια για όλους τους αλγορίθμους πέρα του CC-ERSP3, είναι αρκετά καλύτερα σε σχέση με όλους τους υπόλοιπους αλγορίθμους. Αυτό, πιθανότατα θα συμβαίνει γιατί το BL σύνολο δεδομένων μπορεί να έχει πολύ θόρυβο μέσα του. Το δυνατό σημείο των ERSP3 αλγορίθμων είναι η αφαίρεση του θορύβου, το οποίο γίνεται όταν συναντάται υποσύνολο με ένα μόνο στιγμιότυπο, το οποίο αντί να συμπεριληφθεί στους υπολογισμούς, θεωρείται θόρυβος. Όσων αφορά το ποστοστό μείωσης δεδομένων, οι ERSP3 αλγόριθμοι απέδωσαν πολύ καλύτερα απ'όλους τους RSP3, όπως επίσης έγινε και με το CPU time.

Πίνακας 4.1: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων BL

| - | KNN |
|---|---|
| ACC | 80.606 (%) |
| RR | - |
| DIST | 62300 |
| CPU | 28.087 (ms) |

Πίνακας 4.2: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων BL

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 72.754(%) | 74.365(%) | 66.191(%) | 69.719(%) |
| RR | 63.2(%) | 58.6(%) | 83.76(%) | 70.760(%) |
| DIST | 254187 | 7137.6 | 4928 | 6586.800 |
| CPU | 197.748(ms) | 86.991(ms) | 38.999(ms) | 68.054(ms) |

Πίνακας 4.3: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων BL

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 86.859(%) | 86.857(%) | 71.800(%) | 82.212(%) |
| RR | 86.400(%) | 86.440(%) | 91.480(%) | 89.440(%) |
| DIST | 254187.000 | 7137.600 | 4928.000 | 6586.800 |
| CPU | 148.629(ms) | 62.018(ms) | 42.101(ms) | 88.932(ms) |

### 4.2.2 Πειραματικά αποτελέσματα Συνόλου Δεδομένων KDD

Αρχικά, σχετικά με το σύνολο δεδομένων KDD, θα ήταν καλό να ειπωθεί πως αποτελεί το μεγαλύτερο σύνολο δεδομένων από αυτά που χρησιμοποιούμε στα πλαίσια αυτής της εργασίας. Για τον λόγο αυτό, οι αποστάσεις που υπολογίστηκαν και ο χρόνος εκτέλεσης CPU είναι τόσο μεγάλα σε τιμές σε σχέση με τα υπόλοιπα σύνολα δεδομένων. Στον πίνακα 4.4, ο KNN φαίνεται ως προς την ακρίβεια να παρουσίαζει τα καλύτερα αποτελέσματα σε σχέση με όλους τους υπόλοιπους αλγορίθμους, αλλά υστερεί πολύ σε λιγότερες υπολογισμένες αποστάσεις και CPU time.

Στον πίνακα 4.5, ως προς την ακρίβεια ο καλύτερος αλγόριθμος είναι ο RSP3. Όσον αφορά όμως το ποστοστό μείωσης δεδομένων, λιγότερες υπολογισμένες αποστάσεις και CPU, οι αλγόριθμοι RSP3-CC και RSP3-CC2 φαίνεται να παρουσιάζουν καλύτερα αποτελέσματα, γεγονός λογικό καθώς η διαδικασία με τον υπολογισμό των μέσων φαίνεται πως αποδίδει όσον αφορά τη ταχύτητα και την μείωση της δέσμευσης της CPU από τον αλγόριθμο.

Στον πίνακα 4.6, τα αποτελέσματα των ERSP3 αλγορίθμων ως προς την ακρίβεια, φαίνεται να είναι καλύτερα από αυτά των RSP3 αλγορίθμων ως προς το σύνολό τους. Επίσης και το ποστοστό μείωσης δεδομένων και το CPU time είναι καλύτερο σε σχέση με τους αλγορίθμους του πίνακα 4.5. Από τους ERSP3 αλγόριθμους, αν τους συγκρίνουμε μεταξύ τους, τη καλύτερη ακρίβεια πετυχαίνει ο ERSP3, ενώ το καλύτερο ποστοστό μείωσης δεδομένων και CPU time πετυχαίνει ο CC-ERSP3.

Πίνακας 4.4: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων KDD

| - | KNN |
|---|---|
| ACC | 99.714 (%) |
| RR | - |
| DIST | 3689348813649597440.000 |
| CPU | 5398796.304 (ms) |

Πίνακας 4.5: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων KDD

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 99.601(%) | 99.457(%) | 98.909(%) | 98.868(%) |
| RR | 98.544(%) | 97.760(%) | 99.297(%) | 99.091(%) |
| DIST | 20278726656.000 | 2137128.500 | 1833565.250 | 1788416.375 |
| CPU | 42388497.673 (ms) | 703945.319(ms) | 163361.676(ms) | 169966.282(ms) |

Πίνακας 4.6: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων KDD

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 99.618(%) | 99.517(%) | 98.786(%) | 99.047(%) |
| RR | 99.061(%) | 98.532(%) | 99.577(%) | 99.435(%) |
| DIST | 20278726656.000 | 2137128.500 | 1833565.250 | 1788416.375 |
| CPU | 49080418.141(ms) | 670802.280 (ms) | 153683.552 (ms) | 177098.785 (ms) |

### 4.2.3 Πειραματικά αποτελέσματα Συνόλου Δεδομένων BN

Από τον πίνακα 4.7, ο οποίος αποτελεί τα αποτελέσματα του KNN, φαίνεται πως ο συγκεκριμένος αλγόριθμος πετυχαίνει στο σύνολό του καλύτερη ακρίβεια από όλους τους RSP3, αλγορίθμους, αλλά αυτό δεν ισχύει για το σύνολο των ERSP3 αλγορίθμων. Οι αποστάσεις που υπολογίζει όμως, όπως και το CPU time του δείχνει να είναι πολύ μικρότερο σε σχέση με τους RSP3 και ERSP3 αλγορίθμους.

Στον πίνακα 4.8, ο αλγόριθμος που πετυχαίνει την καλύτερη ακρίβεια είναι ο κλασσικός RSP3, με 84.6% και ακολουθεί ο CC2-RSP3 με 84.46%. Ο CC2-RSP3 όμως, πετυχαίνει πολύ καλύτερο ποστοστό μείωσης δεδομένων, λιγότερες υπολογισμένες αποστάσεις και CPU time από τον RSP3. Γενικά, το καλύτερο ποστοστό μείωσης δεδομένων το επιτυγχάνει ο CC-RSP3 και ίσως και αυτός να είναι ο λόγος που η ακρίβειά του είναι τόσο μικρότερη από τους άλλους 3 αλγορίθμους. Δηλαδή, λόγω μεγάλης μείωσης δεδομένων θα μπορούσε να χάσει δεδομένα τα οποία θα του χρειαζόντουσαν για τη σωστή κατηγοριοποίηση στιγμιοτύπων.

Στον πίνακα 4.9, ο ERSP3 πετυχαίνει τη καλύτερη ακρίβεια, αλλά υστερεί σε λιγότερες υπολογισμένες αποστάσεις και CPU time σε σχέση με τους CC-ERSP3 και CC2-ERSP3. Τη καλύτερη ποσοστιαία μείωση δεδομένων, τη πετυχαίνει ο CC-ERSP3, ο οποίος μάλιστα έχει και το χαμηλότερο CPU time. Στο σύνολό τους, οι ERSP3 αλγόριθμοι επιτυγχάνουν τα καλύτερα αποτελέσματα σε σχέση με όλους τους υπόλοιπους για το σύνολο δεδομένων BN.

Πίνακας 4.7: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων BN

| - | KNN |
|---|---|
| ACC | 86.922 (%) |
| RR | - |
| DIST | 4492704.000 |
| CPU | 956.472 (ms) |

Πίνακας 4.8: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων BN

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 84.601(%) | 84.431(%) | 80.613(%) | 84.469(%) |
| RR | 75.090(%) | 73.703(%) | 82.393(%) | 77.962(%) |
| DIST | 18877824.000 | 78446.398 | 65920.797 | 76751.602 |
| CPU | 16956.437 (ms) | 10608.814(ms) | 8781.140(ms) | 9207.170(ms) |

Πίνακας 4.9: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων BN

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 88.356(%) | 88.300(%) | 87.809(%) | 87.507(%) |
| RR | 90.425(%) | 89.830(%) | 91.726(%) | 90.434(%) |
| DIST | 18877824.000 | 78446.398 | 65920.797 | 76751.602 |
| CPU | 15224.438 (ms) | 10192.489(ms) | 8790.142 (ms) | 8963.796(ms) |

### 4.2.4 Πειραματικά αποτελέσματα Συνόλου Δεδομένων LIR

Για το συγκεκριμένο σύνολο δεδομένων, ο KNN, όσον αφορά την ακρίβειά του, φαίνεται να είναι αρκετά καλύτερος σε σχέση με τους RSP3 και ERSP3 αλγορίθμους. Οι μόνοι που ίσως είναι κάπως κοντά του, είναι οι οι RSP3, RSP3-RND και ο CC2-RSP3, όπως φαίνεται στον πίνακα 4.11. Ο KNN, επιπλέον πετυχαίνει εντυπωσιακά μικρότερο CPU time από όλους τους υπόλοιπους αλγορίθμους των πινάκων 4.11 και 4.12.

Σχετικά με τον πίνακα 4.11, τη μεγαλύτερη ακρίβεια πετυχαίνουν κατά τύχη, υποθέτω, ο RSP3 και RSP3-RND. Γενικά, ως προς την ακρίβεια οι RSP3 αλγόριθμοι φαίνεται να είναι πολύ καλύτεροι από το σύνολο των ERSP3 αλγορίθμων. Η ποσοστιαία μείωση των δεδομένων όμως που επιτυγχάνουν οι RSP3 αλγόριθμοι, είναι πολύ χειρότερη από αυτή των ERSP3 αλγορίθμων.

Οι ERSP3 αλγόριθμοι του πίνακα 4.12, πετυχαίνουν τις χειρότερες τιμές σε μετρικές από όλους τους άλλους αλγορίθμους, εξαιρόντας το ποστοστό μείωσης δεδομένων. Αυτή η μεγάλη ποσοστιαία μείωση δεδομένων η οποία πετυχαίνουν, κατά πάσα πιθανότητα είναι αυτή που ευθύνεται για τη κακή ακρίβεια και το κακό CPU time, μιας και δεδομένα τα οποία είναι απαραίτητα για τη κατηγοριοποίηση μπορεί να χάθηκαν στη πορεία, ή να θεωρήθηκαν ως θόρυβος λανθασμένα.

Πίνακας 4.10: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων LIR

| - | KNN |
|---|---|
| ACC | 95.745 (%) |
| RR | - |
| DIST | 63993600.000 |
| CPU | 50374.683 (ms) |

Πίνακας 4.11: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων LIR

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 95.560(%) | 95.560(%) | 92.055(%) | 94.695(%) |
| RR | 61.884(%) | 54.309(%) | 82.740(%) | 71.624(%) |
| DIST | 329750624.000 | 481638.000 | 385453.594 | 439937.188 |
| CPU | 1065812.334 (ms) | 1277767.953 (ms) | 199851.154 (ms) | 539536.196(ms) |

Πίνακας 4.12: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων LIR

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 91.865(%) | 90.975(%) | 89.819(%) | 92.025(%) |
| RR | 84.075(%) | 83.891(%) | 91.296(%) | 87.664(%) |
| DIST | 329750624.000 | 481638.000 | 385453.594 | 439937.188 |
| CPU | 1175709.848 (ms) | 1417977.095 (ms) | 189918.930 (ms) | 541855.024 (ms) |

### 4.2.5 Πειραματικά αποτελέσματα Συνόλου Δεδομένων LS

Για ακόμη μια φορά, παρατηρείται πως ο KNN πετυχαίνει τα καλύτερα αποτελέσματα όσον αφορά την ακρίβεια, σε σχέση με το σύνολο των αλγορίθμων των RSP3 και ERSP3, εξαιρόντας τον RSP3-RND, ο οποίος δεν υπολογίζεται και τόσο, καθώς πρόκειται για απλή τύχη στη προκειμένη περίπτωση. Ο KNN

όμως, υπολόγισε πολλές περισσότερες αποστάσεις απ'όλους τους αλγορίθμους, με εξαίρεση τους CC-RSP3, CC2-RSP3, CC-ERSP3 και CC2-RSP3 και πέτυχε πολύ χαμηλότερο CPU time, με εξαίρεση πάλι τους 4 αλγορίθμους που προαναφέρθηκαν.

Όσον αφορά τον πίνακα 4.14, ο RSP3-RND, αν και στη τύχη πετυχαίνει τη καλύτερη ακρίβεια, υστερεί κατά πολύ σε ποστοστό μείωσης δεδομένων και σε CPU time, σε σχέση με τον CC2-RSP3 ο οποίος μάλιστα είναι αρκετά κοντά με αυτόν στην ακρίβεια. Το χαμηλότερο CPU time το επιτυγχάνει ο CC-RSP3, ο οποίος φαίνεται να χάνει σε ακρίβεια, έχοντας πιθανός αφαιρέσει στιγμιότυπα που δεν έπρεπε. Έτσι, έχει πολύ μεγάλο ποστοστό μείωσης δεδομένων, αλλά η ακρίβειά του είναι σημαντικά χαμηλότερη σε σχέση με τους υπόλοιπους 3 αλγορίθμους.

Σχετικά με τους ERSP3 αλγορίθμους, αυτοί φαίνεται στο σύνολό τους να παρουσιάζουν καλύτερα αποτελέσματα από τους RSP3 αλγορίθμους. Οι CPU times, των επιμέρους ERSP3 αλγορίθμων είναι σαφώς χαμηλότεροι από αυτούς των RSP3 αλγορίθμων, ενώ το ποστοστό μείωσης δεδομένων τους είναι επίσης μεγαλύτερο απο αυτό των RSP3 αλγορίθμων. Το μεγαλύτερο ποστοστό μείωσης δεδομένων το πετυχαίνει ο CC-ERSP3, ο οποίος κάνει το ίδιο λάθος με τον CC-RSP3, όσον αφορά την μείωση των δεδομένων του.

Πίνακας 4.13: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων LS

| - | KNN |
|---|---|
| ACC | 89.944 (%) |
| RR | - |
| DIST | 6623416.800 |
| CPU | 15041.648(ms) |

Πίνακας 4.14: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων LS

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 89.758(%) | 90.022(%) | 86.618(%) | 89.462(%) |
| RR | 72.890(%) | 69.382(%) | 90.482(%) | 79.033(%) |
| DIST | 34016792.000 | 111570.797 | 77862.797 | 109798.797 |
| CPU | 113274.627 (ms) | 25918.413 (ms) | 3828.972 (ms) | 11050.781(ms) |

Πίνακας 4.15: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων LS

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 89.058(%) | 89.789(%) | 87.022(%) | 89.695(%) |
| RR | 89.060(%) | 88.147(%) | 94.169(%) | 90.936(%) |
| DIST | 34016792.000 | 111570.797 | 77862.797 | 109798.797 |
| CPU | 109307.939 (ms) | 26147.037 (ms) | 3687.873 (ms) | 11760.500 (ms) |

### 4.2.6 Πειραματικά αποτελέσματα Συνόλου Δεδομένων MGT

Στον πίνακα 4.16, φαίνεται η ακρίβεια η οποία πετυχαίνει ο KNN για το σύνολο δεδομένων MGT. Αυτή, αν και σαν σύνολο, ξεπερνάει αυτή των αλγορίθμων RSP3, είναι χαμηλότερη από αυτή των ERSP3 αλγορίθμων, εξαιρόντας τον CC-ERSP3. Ο χρόνος εκτέλεσης CPU όμως(CPU time), είναι σαφώς μικρότερη από το σύνολο των αλγορίθμων RSP3 και ERSP3.

Παρατηρόντας τον πίνακα 4.17, βλέπουμε πως οι RSP3 αλγόριθμοι σαν σύνολο δεν έχουν επιτύχει σημαντικές τιμές σε ACC και ποστοστό μείωσης δεδομένων σε σχέση με τους αλγορίθμους του πίνακα 4.18. Αυτό, ίσως και να οφείλεται σε πιθανό θόρυβο που μπορεί να έχει αυτό το σύνολο δεδομένων, ο οποίος θόρυβος θα αφαιρείται στη συνέχεια από τους ERSP3 αλγορίθμους και θα μας δίνει καλύτερα αποτελέσματα στις μετρικές μας.

Οι αλγόριθμοι του πίνακα 4.18, σαν σύνολο παρουσιάζουν πολύ καλύτερα αποτελέσματα από αυτούς του πίνακα 4.17. Επιμέρους όμως, όσον αφορά την ακρίβεια ο καλύτερος είναι ο ERSP3-RND, ο οποίος όμως υστερεί σημαντικά σε ποστοστό μείωσης δεδομένων, λιγότερες υπολογισμένες αποστάσεις και CPU time, τουλάχιστον όσον αφορά τους αλγορίθμους CC-ERSP3 και CC2-ERSP3.

Πίνακας 4.16: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων MGT

| - | KNN |
|---|---|
| ACC | 80.498 (%) |
| RR | - |
| DIST | 57875577.600 |
| CPU | 30990.911(ms) |

Πίνακας 4.17: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων MGT

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 77.412(%) | 77.323(%) | 71.975(%) | 76.986(%) |
| RR | 58.749(%) | 56.485(%) | 80.683(%) | 64.502(%) |
| DIST | 364760256.000 | 411587.188 | 320482.406 | 432967.594 |
| CPU | 1062318.134(ms) | 858787.884(ms) | 319918.106(ms) | 813923.719(ms) |

Πίνακας 4.18: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων MGT

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 81.976(%) | 82.039(%) | 77.112(%) | 81.455(%) |
| RR | 84.323(%) | 84.366(%) | 89.109(%) | 85.339(%) |
| DIST | 364760256.000 | 411587.188 | 320482.406 | 432967.594 |
| CPU | 1156025.815(ms) | 871534.343 (ms) | 315638.394 (ms) | 818109.317(ms) |

### 4.2.7 Πειραματικά αποτελέσματα Συνόλου Δεδομένων MNK

Το σύνολο δεδομένων MNK, πρόκειται για ένα ελαφρύ σύνολο όπως μας δείχνει ο CPU time στους πίνακες 4.19, 4.20 και 4.21 για τους επιμέρους αλγορίθμους. Όσον αφορά τον KNN, η ακρίβειά του είναι καλύτερη απ'όλους τους άλλους αλγορίθμους, πλην του RSP3 και ERSP3. Ο KNN όμως, αν και χάνει σε ακρίβεια συγκριτικά με αυτούς τους δύο, τους κερδίζει κατά πολύ σε λιγότερες υπολογισμένες αποστάσεις και CPU time.

Τα αποτελέσματα του RSP3, όπως προκύπτουν για από τον πίνακα 4.20, για μια ακόμη φορά είναι χειρότερα σαν σύνολο από αυτά των ERSP3 αλγορίθμων. Τη καλύτερη ακρίβεια τη πετυχαίνει ο RSP3, ο οποίος όμως υστερεί σε ποστοστό μείωσης δεδομένων, λιγότερες υπολογισμένες αποστάσεις και CPU time σε σχέση με τους CC-RSP3 και CC2-RSP3.

Όσον αφορά τον πίνακα 4.21, τη καλύτερη ακρίβεια τη πετυχαίνει ο ERSP3, και σαν σύνολο οι αλγόριθμοι του πίνακα 4.21 ξεπερνάνε τους επιμέρους αλγορίθμους του 4.20, πάντα όσον αφορά την ακρίβεια. Από αυτούς, οι CC-ERSP3 και CC2-ERSP3 πετυχαίνουν πολύ μεγάλη ποσοστιαία μείωση δεδομένων, ενώ οι CPU times τους και οι αποστάσεις οι οποίες υπολογίζουν είναι πολύ μικρές τιμές σε αριθμό.

Πίνακας 4.19: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων MNK

| - | KNN |
|---|---|
| ACC | 90.505 (%) |
| RR | - |
| DIST | 29859.600 |
| CPU | 17.878 (ms) |

Πίνακας 4.20: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων MNK

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 91.224(%) | 80.307(%) | 88.877(%) | 87.501(%) |
| RR | 61.329(%) | 71.850(%) | 95.202(%) | 95.665(%) |
| DIST | 125293.398 | 3866.800 | 2180.800 | 2192.800 |
| CPU | 98.156(ms) | 17.568(ms) | 7.171 (ms) | 7.815(ms) |

Πίνακας 4.21: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων MNK

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 91.214(%) | 84.707(%) | 89.808(%) | 87.728(%) |
| RR | 81.676(%) | 78.382(%) | 95.549(%) | 95.954(%) |
| DIST | 125293.398 | 3866.800 | 2180.800 | 2192.800 |
| CPU | 99.755 (ms) | 17.827(ms) | 7.029 (ms) | 7.856 (ms) |

### 4.2.8 Πειραματικά αποτελέσματα Συνόλου Δεδομένων PD

Για το σύνολο δεδομένων PD, ο KNN είναι αυτός που πετυχαίνει τη μεγαλύτερη ακρίβεια από τους υπόλοιπους αλγορίθμους, ενώ μετά από αυτών ακολουθούν οι RSP3 και ERSP3, οι οποίοι βρίσκονται αρκετά κοντά του. Όσον αφορά το CPU time, ο KNN πετυχαίνει πολύ καλύτερη τιμή από τον RSP3, ERSP3, RSP3-RND και ERSP3-RND, αλλά έχει πολύ χειρότερη τιμή από τους CC-RSP3, CC2-RSP3, CC-ERSP3 και CC2-ERSP3.

Στον πίνακα 4.23, η ακρίβεια των RSP3 αλγορίθμων σαν σύνολο, είναι ελάχιστα καλύτερη από αυτή των ERSP3. Οι RSP3 αλγόριθμοι όμως, έχουν στο σύνολό τους λιγότερο ποστοστό μείωσης δεδομένων από τους ERSP3 αλγορίθμους. Μεταξύ των RSP3 αλγορίθμων, ο RSP3 έχει τη μεγαλύτερη ακρίβεια ενώ ο CC-RSP3 έχει τη μεγαλύτερη τιμή στο ποστοστό μείωσης δεδομένων και ο CC2-RSP3 έχει το χαμηλότερο CPU time.

Από τους αλγορίθμους του πίνακα 4.24, ο ERSP3 πετυχαίνει τη μεγαλύτερη ακρίβεια σε σχέση με τους άλλους 3. Το μεγαλύτερο ποστοστό μείωσης δεδομένων το πετυχαίνει ο CC-ERSP3 ενώ ο CC2-ERSP3

έχει το χαμηλότερο CPU time. Αξίζει να σημειωθεί πως οι ERSP3 αλγόριθμοι σαν σύνολο έχουν χαμηλότερα CPU times από αυτούς του πίνακα 4.23.

Πίνακας 4.22: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων PD

| - | KNN |
|---|---|
| ACC | 99.327 (%) |
| RR | - |
| DIST | 19328332.800 |
| CPU | 16145.144(ms) |

Πίνακας 4.23: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων PD

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 99.163(%) | 98.999(%) | 96.779(%) | 98.226(%) |
| RR | 89.642(%) | 82.802(%) | 96.563(%) | 93.586(%) |
| DIST | 86144816.000 | 191536.000 | 151309.203 | 172439.203 |
| CPU | 133351.715 (ms) | 29453.706(ms) | 2851.872(ms) | 6092.594(ms) |

Πίνακας 4.24: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων PD

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 99.045(%) | 98.726(%) | 96.952(%) | 98.390(%) |
| RR | 93.313(%) | 89.831(%) | 97.748(%) | 96.072(%) |
| DIST | 86144816.000 | 191536.000 | 151309.203 | 172439.203 |
| CPU | 122846.055(ms) | 25858.502(ms) | 2538.588(ms) | 5971.035 (ms) |

### 4.2.9 Πειραματικά αποτελέσματα Συνόλου Δεδομένων PH

Παρατηρόντας τα αποτελέσματα του συνόλου δεδομένων PH, ο KNN για ακόμη μια φορά πετυχαίνει καλύτερη ακρίβεια σε σχέση με τους RSP3 και ERSP3 αλγορίθμους. Επίσης, ο KNN πετυχαίνει κατά πολύ το καλύτερο CPU time σε σχέση με όλους τους υπόλοιπους αλγορίθμους, αν και οι αποστάσεις που υπολογίζει είναι πιο πολλές απ' όλους τους αλγορίθμους εκτός του RSP3 και ERSP3.

Παρατηρόντας τον πίνακα 4.26, ο RSP3 πετυχαίνει τη καλύτερη ακρίβεια ενώ ο CC-RSP3 έχει το καλύτερο ποστοστό μείωσης δεδομένων και CPU time, γεγονός στο οποίο πιθανότατα προκαλεί τη χαμηλή του ακρίβεια. Σε σχέση με τους ERSP3 αλγορίθμους, οι RSP3 αλγόριθμοι υπερτερούν ως προς την ακρίβεια, αλλά υστερούν ως προς το ποστοστό μείωσης δεδομένων. Σε γενικές γραμμές, οι RSP3 αλγόριθμοι επιτυγχάνουν πολύ χαμηλότερο ποστοστό μείωσης δεδομένων σε σχέση με τους υπόλοιπους, για το συγκεκριμένο σύνολο δεδομένων.

Από τους αλγορίθμους του πίνακα 4.27, ο ERSP3 έχει πετύχει τη μεγαλύτερη ακρίβεια η οποία πάραυτα είναι πολύ μικρότερη από αυτή του KNN, δηλαδή τη μεγαλύτερη που έχουμε για το συγκεκριμένο σύνολο δεδομένων. Κατά τ'άλλα, οι CC-ERSP3 και CC2-ERSP3 έχουν πτεύχει πολύ καλύτερα ποστοστό μείωσης δεδομένων και CPU times σε σχέση με τους αντίστοιχους RSP3 αλγορίθμους.

Πίνακας 4.25: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων PH

| - | KNN |
|---|---|
| ACC | 89.636 (%) |
| RR | - |
| DIST | 4670785.200 |
| CPU | 1868.584 (ms) |

Πίνακας 4.26: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων PH

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 86.619(%) | 86.211(%) | 82.731(%) | 86.211(%) |
| RR | 69.313(%) | 67.398(%) | 80.944(%) | 74.078(%) |
| DIST | 21372454.000 | 89097.602 | 71237.203 | 87994.398 |
| CPU | 25040.943 (ms) | 17057.031(ms) | 8546.346(ms) | 11587.467(ms) |

Πίνακας 4.27: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων PH

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 86.156(%) | 85.897(%) | 83.787(%) | 85.804(%) |
| RR | 85.672(%) | 86.421(%) | 89.290(%) | 86.949(%) |
| DIST | 21372454.000 | 89097.602 | 71237.203 | 87994.398 |
| CPU | 22756.113 (ms) | 18748.315 (ms) | 8208.642 (ms) | 11732.231 (ms) |

### 4.2.10 Πειραματικά αποτελέσματα Συνόλου Δεδομένων SH

Σχετικά με τα αποτελέσματα του συνόλου δεδομένων SH, ο KNN πετυχαίνει τη καλύτερη ακρίβεια σε σχέση με τους αλγορίθμους RSP3 και ERSP3. Ο KNN όμως, υστερεί σε CPU time ειδικά όταν τον συγκρίνουμε με τα CPU times των CC-RSP3, CC2-RSP3, CC-ERSP3 και CC2-ERSP3. Οι αποστάσεις

που υπολογίζει, επίσης, είναι λιγότερες από τους RSP3 και ERSP3, αλλά είναι σημαντικά περισσότερες από αυτές όλων των υπόλοιπων αλγορίθμων.

Από τους αλγορίθμους του πίνακα 4.29, ο RSP3 πετυχαίνει τη μεγαλύτερη ακρίβεια, αν και ο CC-RSP3 όπως και ο RSP3-RND είναι αρκετά κοντά του σε τιμή. Οι CC-RSP3 και CC2-RSP3 έχουν σαφώς πολύ μικρότερα CPU times από τον RSP3, ενώ το ποστοστό μείωσης δεδομένων τους είναι επίσης μεγαλύτερο. Συγκριτικά με τους ERSP3 αλγορίθμους, οι RSP3 αλγόριθμοι έχουν χειρότερη ακρίβεια, ποστοστό μείωσης δεδομένων και CPU time από τους ERSP3 αλγορίθμους, χωρίς να υπολογίζω τους RND αλγορίθμους, μιας και είναι random και τα αποτελέσματά τους δεν είναι τόσο αντιπροσωπευτικά.

Από τους ERSP3 αλγορίθμους, ο απλός ERSP3 έχει πετύχει τη μεγαλύτερη ακρίβεια ενώ ο CC-ERSP3 ακολουθεί με μια σχετικά μικρή διαφορά. Σαν σύνολο, οι ERSP3 αλγόριθμοι τα έχουν πάει πολύ καλύτερα σε αυτό το σύνολο δεδομένων απ' ότι οι RSP3 αλγόριθμοι, αλλά δεν έχουν καταφέρει να περάσουν σε ακρίβεια τον κλασσικό KNN.

Πίνακας 4.28: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων SH

| - | KNN |
|---|---|
| ACC | 99.934 (%) |
| RR | - |
| DIST | 538202880.400 |
| CPU | 245160.714 (ms) |

Πίνακας 4.29: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων SH

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 99.483(%) | 99.250(%) | 99.005(%) | 95.827(%) |
| RR | 99.412(%) | 98.759(%) | 99.651(%) | 99.666(%) |
| DIST | 5990951936.000 | 768784.000 | 598077.188 | 643930.812 |
| CPU | 3517949.740 (ms) | 16762.461 (ms) | 5199.180(ms) | 5439.769(ms) |

Πίνακας 4.30: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων SH

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 99.583(%) | 99.197(%) | 98.710(%) | 95.950(%) |
| RR | 99.541(%) | 98.982(%) | 99.724(%) | 99.738(%) |
| DIST | 5990951936.000 | 768784.000 | 598077.188 | 643930.812 |
| CPU | 4183164.900 (ms) | 14852.368 (ms) | 4440.807(ms) | 4988.035(ms) |

### 4.2.11 Πειραματικά αποτελέσματα Συνόλου Δεδομένων TXR

Συγκρίνοντας τα αποτελέσματα των πινάκων 4.31, 4.32 και 4.33, για ακόμη μια φορά ο KNN ξεπερνάει όλους τους υπόλοιπους αλγορίθμους σε ακρίβεια. Ο KNN όμως, υστερεί σημαντικά σε CPU time σε σχέση με τους CC και CC2 RSP3 αλγορίθμους, όπως επίσης και τους CC και CC2 ERSP3 αλγορίθμους. Επιπλέον, οι αποστάσεις που υπολογίζει είναι λιγότερες από τους RSP3 και ERSP3, αλλά πολύ περισσότερες από όλους του υπόλοιπους αλγορίθμους.

Από τους RSP3 αλγορίθμους, τη μεγαλύτερη ακρίβεια πετυχαίνει ο RSP3, ενώ ο CC2-RSP3 βρίσκεται αρκετά κοντά του. Όσον αφορά το ποστοστό μείωσης δεδομένων, οι CC και CC2 επιφέρουν πολύ καλά αποτελέσματα σε σχέση με τους υπόλοιπους, όπως συμβαίνει και με το CPU time. Συγκριτικά με το σύνολο των ERSP3 αλγορίθμων, οι RSP3 αλγόριθμοι πετυχαίνουν καλύτερα αποτελέσματα στην ακρίβεια, αλλά χειρότερα σε ποστοστό μείωσης δεδομένων και CPU time.

Από τους αλγορίθμους του πίνακα 4.33, τη μεγαλύτερη ακρίβεια τη πετυχαίνει ο ERSP3. Στη προκειμένη περίπτωση, φαίνεται πως η παραλλαγή που έχουν οι ERSP3 αλγόριθμοι δεν απέδωσε, σε σχέση με τους RSP3 αλγορίθμους, καθώς πετυχαίνουν μικρότερη ακρίβεια από αυτούς, το οποίο πιθανότατα οφείλεται στο μεγάλο ποστοστό μείωσης δεδομένων που έχουν.

Πίνακας 4.31: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων TXR

| - | KNN |
|---|---|
| ACC | 98.909 (%) |
| RR | - |
| DIST | 4838240.000 |
| CPU | 12239.000(ms) |

Πίνακας 4.32: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων TXR

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 98.618(%) | 98.345(%) | 95.872(%) | 97.418(%) |
| RR | 82.323(%) | 77.918(%) | 94.618(%) | 88.941(%) |
| DIST | 25741710.000 | 87804.398 | 64631.602 | 76447.203 |
| CPU | 84248.107 (ms) | 8212.454 (ms) | 1696.360(ms) | 3262.223(ms) |

Πίνακας 4.33: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων TXR

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 97.654(%) | 97.581(%) | 94.854(%) | 96.690(%) |
| RR | 89.836(%) | 87.809(%) | 96.409(%) | 93.668(%) |
| DIST | 25741710.000 | 87804.398 | 64631.602 | 76447.203 |
| CPU | 81331.869 (ms) | 8028.862 (ms) | 1577.186(ms) | 3147.805(ms) |

### 4.2.12 Πειραματικά αποτελέσματα Συνόλου Δεδομένων YS

Στο σύνολο δεδομένων YS, γενικότερα, δε φαίνεται να πετυχαίνονται μεγάλα ποσοστά ακρίβειας. Ο KNN, δε πετυχαίνει τη μεγαλύτερη ακρίβεια όπως γίνεται συνήθως, αλλά αυτό το επιτυγχάνει ο ERSP3,

ο οποίος μάλιστα πετυχαίνει και το μεγαλύτερο ποστοστό μείωσης δεδομένων. Το CPU time το οποίο επιτυγχάνει ο KNN σε σχέση με όλους τους υπόλοιπους αλγορίθμους είναι πολύ χαμηλότερο.

Από το σύνολο των RSP3 αλγορίθμων, τη μεγαλύτερη ακρίβεια τη πετυχαίνει ο RSP3-RND, γεγονός το οποίο παίζει πέρα από τη τύχη, να οφείλεται και στη μεγάλη ιδιαιτερότητα που έχει το συγκεκριμένο σύνολο δεδομένων. Γενικά , οι ERSP3 αλγόριθμοι στο συγκεκριμένο σύνολο δεδομένων πετυχαίνουν πολύ καλύτερα αποτελέσματα από τους RSP3 αλγορίθμους.

Οι ERSP3 αλγόριθμοι, απ'την άλλη, πετυχαίνουν μεγάλες τιμές ποστοστό μείωσης δεδομένων, ειδικά όταν αυτές συγκρίνονται με τις αντίστοιχες τιμές των RSP3 αλγορίθμων. Από τους ERSP3 αλγορίθμους, ο απλός ERSP3 αλγόριθμος πετυχαίνει τη μεγαλύτερη ακρίβεια, ενώ ο RND επιτυγχάνει το μεγαλύτερο ποστοστό μείωσης δεδομένων και ο CC-ERSP3 το χαμηλότερο CPU time.

Πίνακας 4.34: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων YS

| - | KNN |
|---|---|
| ACC | 51.583 (%) |
| RR | - |
| DIST | 351886.000 |
| CPU | 259.973 (ms) |

Πίνακας 4.35: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων YS

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 49.763(%) | 50.503(%) | 45.378(%) | 47.336(%) |
| RR | 28.155(%) | 25.779(%) | 50.126(%) | 33.985(%) |
| DIST | 2140981.000 | 23615.600 | 22480.801 | 26221.600 |
| CPU | 3867.148 (ms) | 2379.304(ms) | 1466.617 (ms) | 2136.636 (ms) |

Πίνακας 4.36: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων YS

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 56.304(%) | 54.416(%) | 50.506(%) | 53.679(%) |
| RR | 83.825(%) | 84.465(%) | 81.331(%) | 82.544(%) |
| DIST | 2140981.000 | 23615.600 | 22480.801 | 26221.600 |
| CPU | 3327.065(ms) | 2139.384(ms) | 1363.432 (ms) | 2139.790 (ms) |

### 4.2.13 Πειραματικά αποτελέσματα Συνόλου Δεδομένων PM

Ο KNN, στο σύνολο δεδομένων PM πετυχαίνει ακρίβεια 70.535%, χαμηλότερη σε σχέση με τους ERS3P, ERSP3-RND και CC2-ERSP3. Σχετικά με το CPU time του όμως, αυτό είναι κατά πολύ χαμηλότερο σε σχέση με το σύνολο όλων των υπόλοιπων RSP3 και ERSP3 αλγορίθμων.

Από τους αλγορίθμους του πίνακα 4.38, ο CC2-RSP3 πετυχαίνει την καλύτερη ακρίβεια, αν και οι υπόλοιποι πέρα του CC-RSP3 βρίσκονται πολύ κοντά του. Το μεγαλύτερο ποστοστό μείωσης δεδομένων επιτυγχάνεται από τον CC-RSP3 και μαλίστα είναι πολύ μεγαλύτερο από τους υπόλοιπους τρεις αλγορίθμους, γεγονός που μπορεί να δικαιολογεί τη κακή ακρίβεια που πετυχαίνει.

Οι αλγόριθμοι του πίνακα 4.39, πετυχαίνουν καλύτερες μετρικές σαν σύνολο από τους αλγορίθμους του πίνακα 4.38. Εδώ, ξεχωρίζει το γεγονός ότι ο CC2-ERSP3, πετυχαίνει μεγαλύτερο ποσοστό ακρίβειας σε μικρότερο χρόνο από τον αντίστοιχο CC2-RSP3, ενώ το ποστοστό μείωσης δεδομένων και των τεσσάρων ERSP3 αλγορίθμων είναι σχεδόν διπλάσιο από αυτό των RSP3 αλγορίθμων.

Πίνακας 4.37: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων PM

| - | KNN |
|---|---|
| ACC | 70.535 (%) |
| RR | - |
| DIST | 94126.000 |
| CPU | 73.375(ms) |

Πίνακας 4.38: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων PM

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 67.673(%) | 67.671(%) | 63.493(%) | 67.798(%) |
| RR | 44.560(%) | 40.912(%) | 69.674(%) | 50.782(%) |
| DIST | 560834.625 | 12191.200 | 8499.200 | 11605.600 |
| CPU | 812.938 (ms) | 272.432(ms) | 99.770 (ms) | 284.076 (ms) |

Πίνακας 4.39: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων PM

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 70.928(%) | 73.141(%) | 68.969(%) | 71.183(%) |
| RR | 81.629(%) | 81.954(%) | 84.528(%) | 82.248(%) |
| DIST | 560834.625 | 12191.200 | 8499.200 | 11605.600 |
| CPU | 599.519 (ms) | 426.899(ms) | 139.477 (ms) | 234.967(ms) |

### 4.2.14 Πειραματικά αποτελέσματα Συνόλου Δεδομένων TN

Στο σύνολο δεδομένων TN, ο KNN δε πετυχαίνει τη μεγαλύτερη ακρίβεια, καθώς οι ERSP3 αλγόριθμοι φαίνεται να έχουν καλύτερη απόδοση στο σύνολό τους από αυτόν. Επίσης το CPU time το οποίο πετυχαίνει, είναι πολύ μεγαλύτερο από αυτό των CC-RSP3, CC2-RSP3, CC-ERSP3 και CC2-ERSP3.

Οι RSP3 αλγόριθμοι, σαν σύνολο έχουν χειρότερη ακρίβεια, ποστοστό μείωσης δεδομένων και CPU time από τους αλγορίθμους του πίνακα 4.42. Από τους RSP3 αλγορίθμους, μεγαλύτερη ακρίβεια έχει ο RSP3-RND, ενώ μεγαλύτερο ποστοστό μείωσης δεδομένων και χαμηλότερο CPU time έχει ο CC-RSP3, ο οποίος όμως υστερεί κατά πολύ σε ακρίβεια.

Από τους αλγόριθμους του πίνακα 4.42, ο ERSP3 επιτυγχάνει τη μεγαλύτερη ακρίβεια, έχοντας ένα αρκετά μεγάλο ποσοστό μείωσης δεδομένων. Γενικά, οι αλγόριθμοι ERSP3 κατάφεραν στο συγκεκριμένο σύνολο δεδομένων να έχουν τα καλύτερα ποσοστά σε ακρίβεια και αυτό σε συνδιασμό με τις καλύτερες τιμές σε ποστοστό μείωσης δεδομένων και CPU time.

Πίνακας 4.40: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων TN

| - | KNN |
|---|---|
| ACC | 94.702 (%) |
| RR | - |
| DIST | 8759232.000 |
| CPU | 11573.647(ms) |

Πίνακας 4.41: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων TN

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 93.107(%) | 93.513(%) | 82.403(%) | 92.675(%) |
| RR | 84.307(%) | 74.169(%) | 97.912(%) | 84.351(%) |
| DIST | 37490380.000 | 149042.797 | 47072.801 | 122332.797 |
| CPU | 74532.146 (ms) | 22834.317(ms) | 1092.025 (ms) | 8391.510(ms) |

Πίνακας 4.42: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων TN

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 95.608(%) | 95.553(%) | 84.646(%) | 94.837(%) |
| RR | 92.213(%) | 88.476(%) | 98.524(%) | 92.591(%) |
| DIST | 37490380.000 | 149042.797 | 47072.801 | 122332.797 |
| CPU | 67322.301(ms) | 21749.674(ms) | 790.943(ms) | 8150.904(ms) |

### 4.2.15 Πειραματικά αποτελέσματα Συνόλου Δεδομένων WF

Ο KNN, στο σύνολο δεδομένων WF, πετυχαίνει ακρίβεια μικρότερη και από τους RSP3, αλλά και από τους ERSP3 αλγορίθμους σαν σύνολο, με εξαίρεση τους CC-RSP3 και CC2-ERSP3. Όσον αφορά το CPU time, ο KNN πετυχαίνει μικρότερο CPU time από τους RSP3, RSP3-RND και τους αντίστοιχους ERSP3 αλγορίθμους, αλλά έχει αρκετά μεγαλύτερο CPU time από τους υπόλοιπους.

Από τους RSP3 αλγορίθμους, τη μεγαλύτερη ακρίβεια τη πετυχαίνει ο RSP3-RND, ο οποίος όμως έχει και το χαμηλότερο ποστοστό μείωσης δεδομένων. Το μεγαλύτερο ποστοστό μείωσης δεδομένων και το χαμηλότερο CPU time το πετυχαίνει ο CC-RSP3, ο οποίος όμως έχει και τη μικρότερη ακρίβεια. Σε γενικές γραμμές, οι RSP3 αλγόριθμοι έχουν χειρότερη απόδοση από τους ERSP3 αλγορίθμους στο συγκεκριμένο σύνολο δεδομένων.

Οι ERSP3 αλγόριθμοι, έχουν τη καλύτερη απόδοση όσον αφορά ακρίβεια, ποστοστό μείωσης δεδομένων και CPU time σαν σύνολο από όλους τους υπόλοιπους αλγορίθμους. Από τους ERSP3 αλγορίθμους τη μεγαλύτερη ακρίβεια τη πετυχαίνει ο απλός ERSP3.

Πίνακας 4.43: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων WF

| - | KNN |
|---|---|
| ACC | 77.260 (%) |
| RR | - |
| DIST | 4000000.000 |
| CPU | 5704.840(ms) |

Πίνακας 4.44: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων WF

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 77.540(%) | 77.940(%) | 70.840(%) | 77.860(%) |
| RR | 57.025(%) | 50.310(%) | 91.145(%) | 61.355(%) |
| DIST | 16991372.000 | 105062.797 | 55533.199 | 102477.602 |
| CPU | 52977.915 (ms) | 32091.700(ms) | 1813.371 (ms) | 17790.710(ms) |

Πίνακας 4.45: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων WF

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 81.040(%) | 80.880(%) | 71.940(%) | 80.360(%) |
| RR | 85.105(%) | 84.420(%) | 93.900(%) | 86.225(%) |
| DIST | 16991372.000 | 105062.797 | 55533.199 | 102477.602 |
| CPU | 46951.294 (ms) | 28847.012(ms) | 1491.677(ms) | 17662.833(ms) |

### 4.2.16 Πειραματικά αποτελέσματα Συνόλου Δεδομένων EEG

Ο KNN, στο συγκεκριμένο σύνολο δεδομένων επίσης δε πετυχαίνει τη μεγαλύτερη ακρίβεια, με τους RSP3 αλγορίθμους να υπερτερούν σε αυτό το κομμάτι. Όσον αφορά το CPU time του KNN, αυτό είναι χειρότερο από όλους τους αλγορίθμους, εκτός των CC-RSP3 και CC-ERSP3.

Οι αλγόριθμοι του πίνακα 4.47 είναι αυτοί που έχουν τη καλύτερη επίδοση όσον αφορά την ακρίβεια, υστερούν όμως σε ποστοστό μείωσης δεδομένων σε σχέση με τους ERSP3 αλγορίθμους. Τα CPU times τους είναι σχεδόν ίσα, με ελάχιστες αυξομοιώσεις. Από αυτούς τους αλγορίθμους, ο CC-RSP3 πετυχαίνει τη μεγαλύτερη ακρίβεια, το μεγαλύτερο ποστοστό μείωσης δεδομένων και το καλύτερο CPU time.

Από τους αλγορίθμους του πίνακα 4.48, ο CC-ERSP3 πετυχαίνει τη μεγαλύτερη ακρίβεια, το μεγαλύτερο ποστοστό μείωσης δεδομένων και το καλύτερο CPU time. Σαν σύνολο όμως, οι ERSP3 αλγόριθμοι αν και μειώνουν κατά πολύ τα δεδομένα, έχουν χειρότερη ακρίβεια από τους RSP3 αλγορίθμους, ενώ οι CPU times τους είναι σχεδόν ίδιοι.

Πίνακας 4.46: Αποτελέσματα του KNN αλγορίθμου για το σύνολο δεδομένων EEG

| - | KNN |
|---|---|
| ACC | 45.621 (%) |
| RR | - |
| DIST | 35904064.000 |
| CPU | 27268.073(ms) |

Πίνακας 4.47: Αποτελέσματα των RSP3 αλγορίθμων για το σύνολο δεδομένων EEG

| - | RSP3 | RSP3-RND | CC-RSP3 | CC2-RSP3 |
|---|---|---|---|---|
| ACC | 47.310(%) | 46.809(%) | 48.104(%) | 46.929(%) |
| RR | 53.758(%) | 45.754(%) | 76.884(%) | 61.013(%) |
| DIST | 499108608.000 | 360756.406 | 258374.797 | 391864.000 |
| CPU | 972427.494(ms) | 639186.626(ms) | 249054.110(ms) | 493542.400(ms) |

Πίνακας 4.48: Αποτελέσματα των ERSP3 αλγορίθμων για το σύνολο δεδομένων EEG

| - | ERSP3 | ERSP3-RND | CC-ERSP3 | CC2-ERSP3 |
|---|---|---|---|---|
| ACC | 44.479(%) | 44.713(%) | 46.208(%) | 45.127(%) |
| RR | 81.195(%) | 82.825(%) | 86.824(%) | 82.008(%) |
| DIST | 499108608.000 | 360756.406 | 258374.797 | 391864.000 |
| CPU | 1054544.415(ms) | 597605.365(ms) | 238978.029 (ms) | 492219.924(ms) |

## 4.3 Σύγκριση των αποτελεσμάτων από τα διαφορετικά σύνολα δεδομένων

Τα αποτελέσματα τα οποία πήραμε, ήταν αρκετά ενθαρρυντικά όσον αφορά την απόδοση και τη συμπεριφορά των προτεινόμενων αλγορίθμων μας στα δεκαέξι διαφορετικά σύνολα δεδομένων πάνω στα οποία δοκιμάστικαν. Εμείς, περιμέναμε να μείνει η ακρίβεια της κατηγοριοποίησης στα ίδια επίπεδα με τους κλασσικούς αλγορίθμους, ενώ η ποσοστιαία μείωση δεδομένων και το CPU time θα βελτιωνόταν, πέρα από τους αλγορίθμους ERSP3, ERSP3-RND, ERSP3-CC και ERSP3-CC2, όπου εκεί περιμέναμε και βελτίωση στην ακρίβεια. Στον Πίνακα 4.49, θα παρουσιαστεί μια σύνοψη η οποία θα αφορά την αποτελεσματικότητα των αλγορίθμων μας και θα μας βοηθήσει να καταλάβουμε ποιοι αλγόριθμοι υπερέχουν από τους υπόλοιπους και σε ποιους τομείς.

Πέρα από αυτό όμως, υπήρχαν σύνολα δεδομένων στα οποία οι προτεινόμενοι αλγόριθμοι παρουσίαζαν πολύ καλύτερη απόδοση συνολικά, σε σχέση με τον KNN και τον απλό RSP3. Χαρακτηριστικό παράδειγμα, είναι το σύνολο δεδομένων BL, στο οποίο ο RSP3 πετυχαίνει ποσοστό ακρίβειας 72.7%, ενώ ο ERSP3 επιτυγχάνει ποσοστό ακρίβειας 86.8% και μάλιστα σε χαμηλότερο CPU time και με μεγαλύτερη ποσοστιαία μείωση δεδομένων (63.2% έναντι 86.4%). Στο συγκεκριμένο σύνολο δεδομένων, ισχύει για τον CC2-ERSP3 ότι και για τον απλό ERSP3. Ένα δεύτερο παράδειγμα, στο σύνολο δεδομένων WF, ο CC2-RSP3 και CC2-ERSP3 φαίνεται να έχουν μεγαλύτερα ποσοστά ακρίβειας όπως και ποσοστιαίας μείωσης δεδομένων από τον απλό RSP3. Εκεί, ο RSP3 πετυχαίνει 77.5% ακρίβεια με 57% RR ενώ και οι δύο CC2 αλγόριθμοι που προαναφέρθηκαν περυχαίνουν μεγαλύτερη ακρίβεια με μεγαλύτερο RR. Υπάρχουν όμως και περιπτώσεις όπως στο KDD σύνολο δεδομένων, στο οποίο ο απλός KNN πετυχαίνει το μεγαλύτερο ποσοστό ακρίβειας από όλους τους άλλους αλγορίθμους.

Συγκρίνοντας τον τρόπο λειτουργίας των CC-RSP3/CC-ERSP3 και CC2-RSP3/CC2-ERSP3 αλγορίθμων, όπου οι πρώτοι βρίσκουν ένα τεχνητό μέσο στιγμιότυπο και το χρησιμοποιούν για τον υπολογισμό αποστάσεων, ενώ οι δεύτεροι βρίσκουν το κοντινότερο υπαρκτό σήμειο στο τεχνητό μέσο, και έπειτα υπολογίζουν αποστάσεις, φαίνεται πως οι αλγόριθμοι αυτοί δεν μας δίνουν ένα σαφές αποτέλεσμα ώς προς το ποιος είναι πιο αποδοτικός, αλλά η αποτελεσματικότητά τους δείχνει να εξαρτάται από το σύνολο δεδομένων. Χαρακτηριστικό παράδειγμα αυτού του συμπεράσματος, είναι το σύνολο δεδομένων LS, στο οποίο ο CC2-RSP3 έχει μεγαλύτερη ακρίβεια από τον CC-RSP3 αλλά πετυχαίνει μικρότερη ποσοστιαία μείωση δεδομένων. Το ίδιο ισχύει και για τους CC-ERSP3 και CC2-ERSP3 αλγορίθμους στο συγκεκριμένο σύνολο δεδομένων. Απ΄την άλλη όμως, έχουμε και το παράδειγμα του συνόλου δεδομένων EEG, στο οποίοι οι CC-RSP3 και CC-ERSP3, φαίνεται να έχουν και μεγαλύτερη ακρίβεια, αλλά και μεγαλύτερη ποσοστιαία μείωση δεδομένων από τους αντίστοιχους CC2-RSP3 αλγορίθμους. Επομένως, καταλαβαίνουμε πως στα αποτελέσματα μεγάλο ρόλο παίζει και το σύνολο των δεδομένων, πέρα από τη λειτουργία των αλγορίθμων.

Στον Πίνακα 4.49 που ακολουθεί, θα δείτε το σύνολο των μετρικών και των αλγορίθμων που χρησιμοποιήσαμε κατά τη πειραματική διαδικασία, με εξαίρεση τον KNN, και στη συνέχεια θα σχολιάσουμε το περιεχόμενο του συγκεκριμένου πίνακα. Θα χρησιμοποιηθούν τα σύμβολα ✓ − και ? στα κελιά του Πίνακα 4.49, ως ένας τρόπος αξιολόγησης του κάθε αλγορίθμου σχετικά με μία από τις τέσσερις μετρικές που χρησιμοποιήσαμε κατά τη πειραματική διαδικασία. Ας σημειωθεί, ότι με ✓ θεωρούμε πως ο εκάστοτε αλγόριθμος υπερέχει σε σχέση με τους υπόλοιπους, με - ο αλγόριθμος υστερεί σε σχέση με τους υπόλοιπους, ενώ ? βάζουμε σε ένα κελί όταν η απόδοση του αλγορίθμου για τη συγκεκριμένη μετρική

έχει ποικίλες διακυμάνσεις ανάλογα με το σύνολο δεδομένων.

Πίνακας 4.49: Συμπεράσματα για την αποτελεσματικότητα των αλγορίθμων μας

| - | ACC | RR | DIST | CPU |
|---|---|---|---|---|
| RSP3 | - | - | - | - |
| RSP3-RND | ✓ | - | ? | ? |
| RSP3-CC | - | ✓ | ✓ | ✓ |
| RSP3-CC2 | ? | ✓ | ✓ | ✓ |
| ERSP3 | ✓ | ? | - | - |
| ERSP3-RND | ✓ | ? | ? | ? |
| ERSP3-CC | ? | ✓ | ✓ | ✓ |
| ERSP3-CC2 | ✓ | ✓ | ✓ | ✓ |

Συνολικά, τα αποτελέσματα τα οποία πήραμε δεν μας οδηγήσανε σε σαφή συμπεράσματα για το αν το σύνολο των αλγορίθμων μας είναι αποδοτικό ή όχι ως προς την ακρίβεια. Σε κάθε περίπτωση όμως, η ακρίβεια παραμένει σε αποδεκτά επίπεδα. Οι ERSP3 αλγόριθμοι, φαίνεται σε γενικές γραμμές να βελτίωσαν την ακρίβεια της κατηγοριοποίησης, καθώς στη πλειοψηφία των συνόλων πετυχαίνουν μεγαλύτερα ποσοστά ακρίβειας από τους RSP3 αλγορίθμους. Επίσης, τα αποτελέσματα μας οδήγησαν στο συμπέρασμα πως οι αλγόριθμοι οι οποίο προτείναμε δεσμεύουν κατά λιγότερο χρόνο τη CPU και πως η ποσοστιαία μείωση των δεδομένων η οποία πετυχαίνουν είναι μεγαλύτερη σε σχέση με αυτή των παραδοσιακών αλγορίθμων. Τώρα όσον αφορά τους CC-RSP3/CC-ERSP3 και CC2-RSP3/CC2-ERSP3 αλγορίθμους, φαίνεται πως το τεχνητό μέσο σε κάποια σύνολα δεδομένων έχει καλύτερη απόδοση, ένω το υπαρκτό μέσο έχει καλύτερη απόδοση σε κάποια άλλα. Επειδή τα αποτελέσματα των αλγορίθμων έχουν άμεση σχέση με τα σύνολα των δεδομένων πάνω στα οποία εκτελούνται, μπορούν και έχουν διακυμάνσεις από σύνολο σε σύνολο. Τελειώνοντας, αξίζει να σημειωθεί πως δοκιμάζοντας τους αλγόριθμούς μας σε ποικίλα σύνολα δεδομένων, θα μπορούμε να έχουμε μια πιο αντικειμενική εικόνα για τα αποτελέσματά των αλγορίθμων. Αυτός είναι και ο λόγος για τον οποίο χρησιμοποιούμε τόσα σύνολα δεδομένων και τα συγκρίνουμε μεταξύ τους.

## Κεφάλαιο 5ο: Συμπεράσματα και Μελλοντική έρευνα

Στη συγκεκριμένη εργασία, εστιάσαμε στους αλγόριθμους μείωσης δεδομένων και τους αλγορίθμους κατηγοριοποίησης στιγμιοτύπων. Στο παρελθόν, είχαν προταθεί διάφοροι αλγόριθμοι μείωσης των δεδομένων, οι οποίο αν και αποτελεσματικοί, μας άφησαν κάποια περιθώρια για να τους βελτιώσουμε. Σκοπός αυτών των αλγορίθμων, είναι η δημιουργία ενός συμπυκνωμένου συνόλου δεδομένων, το οποίο θα έχει πολύ μικρότερο μέγεθος και υπολογιστικό κόστος από το αρχικό, ενώ ταυτόχρονα η ακρίβειά του δε θα επηρεάζεται αρνητικά σε μεγάλο βαθμό. Για να επιτευχθεί αυτό, χρησιμοποιούνται ήδη γνωστοί αλγόριθμοι για τη μείωση των δεδομένων με διαχωρισμό του χώρου όπως ο RSP3, ο οποίος αποτέλεσε ένα από τα κύρια μέρη της παρούσας εργασίας. Ο συγκεκριμένος αλγόριθμος όμως, έχει το μειονέκτημα του μεγάλου υπολογιστικού κόστους για τη δημιουργία του συμπικνωμένου συνόλου. Για τον λόγο αυτό, προτάθηκαν παραλλαγές του RSP3, οι οποίες είχαν σαν σκοπό τη μείωση του υπολογιστικού κόστους και τη βελτίωση των μετρικών που χρησιμοποιήσαμε για την αξιολόγηση των αλγορίθμων κατά τη πειραματική διαδικασία. Έτσι, αφού παρουσιάσαμε τις βασικές έννοιες της κατηγοριοποίησης, του k-NN αλγορίθμου, τις τεχνικές μείωσης των δεδομένων και των υπάρχοντων αλγορίθμων μείωσης δεδομένων βάση διαχωρισμού του χώρου, προτείναμε κάποιες νέες παραλλαγές αυτών των αλγορίθμων μείωσης των δεδομένων με διαχωρισμό του χώρου, με σκοπό την αντιμετώπιση των προβλημάτων που παρουσιάζουν οι ήδη υπάρχοντες αλγόριθμοι. Στο τέλος, παρουσιάστηκε μια εκτεταμένη πειραματική μελέτη στην οποία οι προτεινόμενοι αλγόριθμοι συγκρίθηκαν μεταξύ τους αλλά και με τους ήδη υπάρχοντες αλγορίθμους, σε πειράματα τα οποία εκτελέστηκαν για διάφορα σύνολα δεδομένων. Τα αποτελέσματα αυτά, ανέδειξαν την αποτελεσματικότητα που είχαν οι προτεινόμενοι αλγόριθμοι σε ορισμένες μετρικές, ενώ ταυτόχρονα μας έδειξαν και τα μειονεκτήματά τους.

Για τον λόγο αυτό, στα πλαίσια των μελλοντικών ερευνών που θα γίνουν πάνω στο συγκεκριμένο αντικείμενο, θα μπορούσαν να δημιουργηθούν νέες διαφορετικές παραλλαγές του αλγορίθμου RSP3, σαν αυτές που παρουσιάστηκαν στη παρούσα εργασία, από διαφορετικούς ερευνητές. Επίσης, θα μπορούσαν άλλοι ερευνητές να βελτιώσουν τους αλγορίθμους που αναφέρονται σε αυτή την εργασία, είτε ως προς τη λειτουργία τους κατά τη πειραματική διαδικασία, είτε ως προς τη λογική την οποία ακολουθείται σε αυτούς. Τέλειώνοντας, στην εποχή των Big Data την οποία βρισκόμαστε και θα συνεχίσουμε να βρισκόμαστε για πολλά χρόνια ακόμα, αλγόριθμοι όπως ο RSP3, οι οποίοι έχουν μεγάλο υπολογιστικό κόστος ακόμα και σε μικρότερα σύνολα δεδομένων, είναι λογικό πως δεν μπορούν να έχουν πρακτική χρήση. Για εξής λόγο, χρειαζόμαστε νέες παραλλαγές αυτού του αλγορίθμου, οι οποίες θα βελτιώνουν τον αλγόριθμο τόσο σε υπολογιστικό κόστος, αλλά και όσο γίνεται και στις υπόλοιπες τρεις μετρικές που προαναφέρθηκαν στη συγκεκριμένη εργασία.

## ΒΙΒΛΙΟΓΡΑΦΙΑ


[1] J. Han, M. Kamber, and J. Pei, "Data mining concepts and techniques, third edition," 2012.

[2] M. James, *Classification Algorithms*. USA: Wiley-Interscience, 1985.

[3] J. Fürnkranz, *Decision Tree*, pp. 263–267. Boston, MA: Springer US, 2010.

[4] L. Rokach and O. Maimon, *Data Mining With Decision Trees: Theory and Applications*. USA: World Scientific Publishing Co., Inc., 2nd ed., 2014.

[5] S. Haykin, *Neural Networks: A Comprehensive Foundation*. USA: Prentice Hall PTR, 2nd ed., 1998.

[6] P. Domingos and M. Pazzani, "On the optimality of the simple bayesian classifier under zero-one loss," *Mach. Learn.*, vol. 29, p. 103–130, Nov. 1997.

[7] O. Harrison, "Machine learning basics with the knearest neighbors algorithm.."

[8] S. Ougiaroglou, A. Nanopoulos, A. Papadopoulos, Y. Manolopoulos, and T. Welzer, "Adaptive k -nearest-neighbor classification using a dynamic number of nearest neighbors," pp. 66–82, 09 2007.

[9] M. M. Deza and E. Deza, *Encyclopedia of distances*. Berlin: Springer, 2009.

[10] S. A. Dudani, "The distance-weighted k-nearest-neighbor rule.," *IEEE Trans. Syst. Man Cybern.*, vol. 6, no. 4, pp. 325–327, 1976.

[11] I. Triguero and F. Herrera, "A taxonomy and experimental study on prototype generation for nearest neighbor classification," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev*, pp. 86–100, 2012.

[12] B. V. Dasarathy, J. Sánchez, and S. Townsend, "Nearest neighbour editing and condensing tools - synergy exploitation," 2000.

[13] J. Sánchez and F. Bañón, "Data reduction techniques in classification processes," *http://www.tesisenxarxa.net*, 01 2007.

[14] J. Sánchez, "High training set size reduction by space partitioning and prototype abstraction.," *Pattern Recognition*, vol. 37, pp. 1561–1564, 01 2004.

[15] C. Chen and A. Józwik, "A sample set condensation algorithm for the class sensitive artificial neural network," *Pattern Recognit. Lett.*, vol. 17, pp. 819–823, 1996.

[16] J. Sánchez, "High training set size reduction by space partitioning and prototype, volume = 37, journal = Pattern Recognition," pp. 1561–1564, 01 2004.

[17] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Series in Data Management Systems, Amsterdam: Morgan Kaufmann, 3 ed., 2011.

[18] S. Galliano, E. Geoffrois, G. Gravier, J.-F. Bonastre, D. Mostefa, and K. Choukri, "Corpus description of the ESTER evaluation campaign for the rich transcription of French broadcast news," in *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, (Genoa, Italy), European Language Resources Association (ELRA), May 2006.

[19] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp. 1–6, 2009.

[20] H. Azwar, M. Murtaz, M. Siddique, and S. Rehman, "Intrusion detection in secure network for cybersecurity systems using machine learning and data mining," in *2018 IEEE 5th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, pp. 1–9, 2018.

[21] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, and S. García, "Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *Multiple-Valued Logic and Soft Computing*, vol. 17, no. 2-3, pp. 255–287, 2011.

[22] H. Al-Sheakh, "Letter recognition data using neural network," *International Journal of Scientific and Engineering Research*, vol. 4, 05 2013.

[23] K. Bache and M. Lichman, "UCI machine learning repository," 2013.

[24] S. Thrun, J. Bala, E. Bloedorn, I. Bratko, J. Cheng, K. De Jong, S. Džeroski, S. Fahlman, D. Fisher, R. Hamann, K. Kaufman, S. Keller, I. Kononenko, J. Kreuziger, T. Mitchell, P. Pachowicz, Y. Reich, H. Vafaie, and J. Wnek, "The monk's problems a performance comparison of different learning algorithms," 01 1992.

[25] J. Wnek and R. Michalski, "Hypothesis-driven constructive induction in aq17: A method and experiments," vol. 14, 08 2002.

[26] G. Tecuci, M. Kaumann, J. Wnek, and R. Michalski, "Comparing symbolic and subsymbolic learning: Three studies," 08 2002.

[27] R. Cerri, Silva, R. Rodrigues Oliveira da Silva, A. de Carvalho, and A. F, "Comparing methods for multilabel classification of proteins using machine learning techniques," 07 2009.

[28] B. L., "Bias, variance and arcing classifiers.," April 1996.

[29] R. T. Olszewski, R. Maxion, and D. Siewiorek, *Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data*. PhD thesis, USA, 2001. AAI3040489.

[30] T. Wang, S.-U. Guan, K. Man, and T. Ting, "Time series classification for eeg eye state identification based on incremental attribute learning," 05 2014.

[31] T. Tsuji, O. Fukuda, H. Ichinobe, and M. Kaneko, "A log-linearized gaussian mixture network and its application to eeg pattern classification," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 29, pp. 60 – 72, 03 1999.

[32] M. Yeo, X. Li, K. Shen, and E. Wilder-Smith, "Can svm be used for automatic eeg detection of drowsiness during car driving?," 2009.

[33] G. Libralon, A. de Carvalho, and A. Lorena, “Pre-processing for noise detection in gene expression classification data,” *Journal of the Brazilian Computer Society*, vol. 15, pp. 3–11, 03 2009.

[34] D. Berrar, *Cross-Validation*. 01 2018.

[35] J. D. Foley and A. V. Dam, “Fundamentals of interactive computer graphics.,” 1982.