Ατομική Διπλωματική Εργασία

ΜΕΛΕΤΗ ΜΕΘΟΔΩΝ ΜΗΧΑΝΙΚΗΣ ΚΑΙ ΒΑΘΙΑΣ ΜΑΘΗΣΗΣ ΚΑΙ ΕΦΑΡΜΟΓΗ ΣΕ ΨΥΧΟΜΕΤΡΙΚΑ ΔΕΔΟΜΕΝΑ

Άντρια Τριγιώργη

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μελέτη Μεθόδων Μηχανικής και Βαθιάς Μάθησης

και Εφαρμογή Σε Ψυχομετρικά Δεδομένα

Άντρια Τριγιώργη

Επιβλέπων Καθηγητής

Χρύσης Γεωργίου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2016

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Χρύση Γεωργίου που μου εμπιστεύτηκε το θέμα και με βοήθησε κατά την διάρκεια εκπόνησης της διπλωματικής μου εργασίας. Επίσης θα ήθελα να ευχαριστήσω τον αδερφό μου Γιώργο για τις συμβουλές του και την βοήθεια του για την κατανόηση κάποιων εννοιών. Επιπλέον θέλω να ευχαριστήσω το Τμήμα Ψυχολογίας που μου παρείχαν τα δεδομένα και ιδιαίτερα την κ. Μαρία Καρεκλά.

Τέλος θέλω να ευχαριστήσω τους γονείς μου Ρίκκο και Νικολέττα, και τον φίλο μου Κυριάκο, για την συνεχή ενθάρρυνση και υποστήριξη που μου παρείχαν κατά την διάρκεια των σπουδών μου.

Περίληψη

Ο στόχος αυτής της εργασίας είναι η μελέτη και η εφαρμογή διάφορων αλγορίθμων κατηγοριοποίησης/συσταδοποίησης σε δύο διαφορετικά σύνολα από δεδομένα. Το πρώτο σύνολο δεδομένων περιείχε καπνιστές και θέλαμε να τους κατατάξουμε σε δύο κλάσεις, σε αυτούς που αποδέχονται την εμπειρική αποφυγή και σε αυτούς που δεν την αποδέχονται. Για το δεύτερο σύνολο δεδομένων, θέλαμε να κατατάξουμε τους ασθενείς σε δύο κλάσεις, σε αυτούς με ψηλό ή χαμηλό ρίσκο ως προς διατροφικές διαταραχές. Τα δεδομένα αυτά δημιουργήθηκαν από το Τμήμα Ψυχολογίας του Πανεπιστημίου Κύπρου. Περιέχουν τρεις μετρήσεις για τον κάθε ασθενή ECG (electrocardiography), SCR (skin conductance response) και COR (corrugator muscle).

Χρησιμοποιήσαμε τόσο αλγόριθμους επιβλεπόμενης μάθησης, όσο και αλγόριθμους μη-επιβλεπόμενης μάθησης. Για τους αλγορίθμους επιβλεπόμενης μάθησης χρησιμοποιήσαμε τα πιο πάνω δεδομένα με βάση το εμπειρικό συμπέρασμα που έβγαλε το Τμήμα Ψυχολογίας, δηλ. σε ποια κλάση ανήκει ο κάθε ασθενείς στην πραγματικότητα. Οι επιβλεπόμενοι αλγόριθμοι που εφαρμόστηκαν είναι, οι Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machine-SVM) και ο αλγόριθμος κατηγοριοποίησης k κοντινότερων γειτόνων (k-Nearest Neighbor- KNN). Ο μη-επιβλεπόμενος αλγόριθμος που εφαρμόστηκε είναι ο k-means.

Τέλος γίνεται η συγκριτική αξιολόγηση των αλγορίθμων με βάση κάποιες μετρικές. Καταλήξαμε στο συμπέρασμα ότι ο πιο αποδοτικός αλγόριθμος επιβλεπόμενης μάθησης στην ανάλυση των συγκεκριμένων δεδομένων, είναι ο SVM. Ενώ στην μη επιβλεπόμενη μάθηση, η πιο αποδοτική προσέγγιση εφαρμογής του k-means, ήταν να εξάγουμε την διάμεσο του κάθε δείγματος δεδομένων και μετά να εφαρμόσουμε τον συσταδοποιητή.

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

1.1 Κίνητρο

Σκεπτόμενες μηχανές και τεχνητά όντα υπήρχαν στους ελληνικούς μύθους, όπως το χάλκινο ρομπότ του Ήφαιστου και η Γαλάτεια του Πυγμαλίωνα. Επίσης τεχνητές απομιμήσεις ανθρώπινων μορφών εμφανίζονται σε όλους τους μεγάλους αρχαίους πολιτισμούς. Η δε λογική μελετήθηκε από φιλόσοφους και μαθηματικούς από την αρχαιότητα και η μελέτη της οδήγησε στην εφεύρεση του πρώτου προγραμματιζόμενου ψηφιακού ηλεκτρικού υπολογιστή που βασίζεται στις θεωρίες του μαθηματικού Alan Turing, ο οποίος εισηγήθηκε ότι μια μηχανή απλά χρησιμοποιώντας τα σύμβολα 0 και 1 θα μπορούσε να προσομοιάσει οποιαδήποτε πράξη μαθηματικής επαγωγής. Αυτή η εφεύρεση μαζί με άλλες σύγχρονες ανακαλύψεις στους τομείς της Νευρολογίας και της θεωρίας της Πληροφορικής ενέπνευσε ένα μικρό αριθμό ερευνητών να αρχίσουν να σκέπτονται σοβαρά την πιθανότητα κατασκευής ενός ηλεκτρονικού μυαλού. Σαν αποτέλεσμα τα τελευταία χρόνια το πεδίο της Τεχνητής Νοημοσύνης αναπτύσσεται ραγδαία, φτάνοντας στο σημείο να μπορεί να επεξεργάζεται πολύπλοκα προβλήματα με πολυδιάστατα δεδομένα [75]. Επειδή όμως είναι πρακτικά αδύνατον να μοντελοποιηθεί ολόκληρος ο κόσμος, είναι πολύ πιο εύκολο (εφόσον ο κόσμος είναι πάρα πολύ μεγάλος και αλλάζει συνεχώς) να μοντελοποιηθεί το μυαλό, ούτως ώστε να δημιουργηθούν συστήματα ικανά να μαθαίνουν τα πάντα που συμβαίνουν στον κόσμο [77]. Έτσι, η *Μηχανική* [4] *και Βαθιά Μάθηση* [2] αναδείχθηκαν οι πιο κεντρικοί τομείς της Τεχνητής Νοημοσύνης. Στην μελέτη μου έδειξα πως εφαρμόζεται η Μηχανική Μάθηση σε ένα σημαντικό και χαρακτηριστικό τομέα, όπως είναι αυτός της Ψυχολογίας [75]. Η έρευνα στον τομέα της Ψυχολογίας, όπως και σε άλλα επιστημονικά πεδία, είχε μεγάλη ανάπτυξη τα τελευταία χρόνια, όμως λόγω της επακόλουθης ανάγκης χειρισμού τεράστιου όγκου δεδομένων (εφόσον οι έρευνες επεκτείνονταν συνεχώς για

να συμπεριλάβουν όλο και μεγαλύτερο αριθμό ασθενών) προέκυψε η ανάγκη μιας νέας τεχνολογίας που να μπορέσει να βοηθήσει στην επεξεργασία και ανάλυση πολύπλοκων συστημάτων. Έτσι η ταυτόχρονη σχεδόν ανάπτυξη της Τεχνητής Νοημοσύνης ήρθε στην κατάλληλη στιγμή, από την μια να ενισχύσει ουσιαστικά την έρευνα σε πολλά νέα επιστημονικά πεδία, και από την άλλη αυτά τα αντικείμενα έρευνας άνοιξαν νέους ορίζοντες ανάπτυξης, αλλά και πρακτικής εφαρμογής της Μηχανικής Μάθησης.

1.2 Σκοπός

Ο σκοπός της παρούσας εργασίας είναι καταρχήν η μελέτη κάποιων αλγορίθμων Μηχανικής και Βαθιάς Μάθησης. Ακολούθως η εφαρμογή κάποιων από αυτούς τους αλγορίθμους σε δύο διαφορετικά σύνολα δεδομένων, ώστε να κατανοήσουμε την χρησιμότητα της Μηχανικής Μάθησης στην διάγνωση παθήσεων. Και στις δύο μελέτες, ο στόχος ήταν η ανάλυση τριών φυσιολογικών μετρήσεων, του ECG, SCR και COR και η κατάταξη των ασθενών σε δύο κλάσεις. Στην πρώτη μελέτη, έπρεπε οι καπνιστές να διαχωριστούν σε αυτούς με χαμηλή εμπειρική αποφυγή και σε αυτούς με ψηλή εμπειρική αποφυγή. Ενώ στην δεύτερη μελέτη, οι ασθενείς έπρεπε να διαχωριστούν σε αυτούς με χαμηλό ρίσκο ως προς διατροφικές διαταραχές και σε αυτούς με ψηλό ρίσκο.

1.3 Μεθοδολογία

Η μεθοδολογία που χρησιμοποίησα ήταν καταρχήν να συλλέξω κάποια άρθρα για την Μηχανική και Βαθιά Μάθηση, και για κάποιους από τους αλγορίθμους Επιβλεπόμενης και μη Επιβλεπόμενης Μάθησης που συνάντησα στην πορεία. Αφού μελέτησα κάποιους από τους αλγορίθμους που θεώρησα χρήσιμους για τα προβλήματα που είχα να λύσω, βρήκα έτοιμες υλοποιήσεις τους από την βιβλιοθήκη sklearn [78]. Για την καλύτερη κατανόηση των αλγορίθμων αυτών, τους έτρεξα σε δικά μου μικρά παραδείγματα, τα οποία αναφέρονται στην εξήγηση κάθε αλγορίθμου. Αργότερα παρέλαβα δύο σύνολα δεδομένων από το Τμήμα Ψυχολογίας με τις μετρήσεις των φυσιολογικών σημάτων για κάθε ασθενή. Καταρχήν επεξεργάστηκα αυτά τα δεδομένα ώστε να αφαιρέσω όλα τα ανεπιθύμητα σήματα και εξήγαγα κάποια σημαντικά χαρακτηριστικά από αυτά για να μπορέσω να εφαρμόσω με αποδοτικό τρόπο τους αλγορίθμους. Για την αξιολόγηση των αλγορίθμων μελέτησα διάφορες μετρικές αξιολόγησης τους. Στο τέλος αξιολόγησα τους αλγορίθμους και τους σύγκρινα μεταξύ τους, ώστε να καταλήξω στον πιο αποδοτικό από αυτούς για το συγκεκριμένο πρόβλημα.

1.4 Οργάνωση Διπλωματικής Εργασίας

Το υπόλοιπο της παρούσας εργασίας χωρίζεται σε επτά ενότητες που καταλαμβάνουν το Κεφάλαιο 2-8, αντίστοιχα. Συγκεκριμένα :

Στο Κεφάλαιο 2 παρουσιάζουμε κάποιους γενικούς ορισμούς, όπως ο ορισμός της Μηχανικής και Βαθιάς Μάθησης και περιγράφουμε τρία φυσιολογικά σήματα, το Ηλεκτροκαρδιογράφημα, η Στατική Αντίδραση Δέρματος και το Ηλεκτροκαρδιογράφημα.

Στο Κεφάλαιο 3 περιγράφουμε τα στάδια από τα οποία αποτελείται η Διαδικασία Μάθησης ενός Κατηγοριοποιητή.

Στο Κεφάλαιο 4 παρουσιάζουμε τις δύο Κατηγορίες Μάθησης, την Επιβλεπόμενη και μη Επιβλεπόμενη Μάθηση. Περιγράφουμε κάποιους από τους αλγορίθμους των δύο Κατηγοριών Μάθησης. Επιπλέον περιγράφουμε την Μέθοδο του "Αγκώνα" για την επιλογή του βέλτιστου k για τον συσταδοποιητή k-Means, και τον αλγόριθμο Ανάλησης Κύριων Συνιστώσων για την μείωση των διαστάσεων στα δεδομένα.

Στο Κεφάλαιο 5 παρουσιάζουμε μετρικές αξιολόγησης για την αξιολόγηση των κατηγοριοποιητών και των συσταδοποιητών, οι οποίες είναι η Ακρίβεια, η Ευαισθησία, η Προσδιοριστικότητα, ο Πίνακας Σύγχυσης και η Τιμή Σιλουέτας.

Στο Κεφάλαιο 6 γίνεται η περιγραφή των δεδομένων που χρησιμοποιήθηκαν για την διάγνωση Εμπειρικής Αποφυγής σε καπνιστές, αλλά και η ανάλυση των αποτελεσμάτων που παράχθηκαν από την εφαρμογή κάποιων αλγορίθμων Μάθησης σε αυτά τα δεδομένα.

Στο Κεφάλαιο 7 γίνεται η περιγραφή των δεδομένων που χρησιμοποιήθηκαν για την διάγνωση Διατροφικών Διαταραχών σε ασθενείς, αλλά και η ανάλυση των αποτελεσμάτων που παράχθηκαν από την εφαρμογή κάποιων αλγορίθμων Μάθησης σε αυτά τα δεδομένα.

Τέλος, στο Κεφάλαιο 8 βρίσκονται τα τελικά συμπεράσματα της εργασίας, τα κύρια προβλήματα που αντιμετωπίστηκαν και μελλοντικές επεκτάσεις που μπορούν να γίνουν.

Μετά από αυτά τα επτά κεφάλαια παρατίθεται η βιβλιογραφία της παρούσας εργασίας και τρία Παραρτήματα στα οποία παραθέτουμε τον κώδικα που έχουμε παράξει.

Κεφάλαιο 2

Γενικό Υπόβαθρο

2.1 Μηχανική και Βαθιά Μάθηση

Ένα σημαντικό μέρος του μεγάλου πεδίου της Τεχνητής Νοημοσύνης (artificial Intelligence) είναι η Μηχανική Μάθηση (machine learning) [4], η οποία ασχολείται με αλγορίθμους που προσπαθούν να μάθουν από την ανάλυση ενός συνόλου δεδομένων. Μάθηση για ένα πρόγραμμα, σημαίνει η εμπειρία που λαμβάνει μέσω των αλληλεπιδράσεων του με τον πραγματικό κόσμο και οι λήψεις αποφάσεων που παίρνει. Όπως είπε και ο Tom M. Mitchell, "Ένα πρόγραμμα υπολογιστή λέγεται ότι μαθαίνει από μια εμπειρία E σε σχέση με μια σειρά από έργα T και μια μέτρηση της απόδοσης P η οποία βελτιώνεται με την εμπειρία E" [4]. Ακόμα μάθηση για ένα πρόγραμμα μπορεί να σημαίνει όχι μόνο απλή χρήση της εμπειρίας, αλλά να βρει τις συσχετίσεις μεταξύ των διάφορων διαθέσιμων δεδομένων και να παράγει κάποια (λογικά) συμπεράσματα.

Η βαθιά μάθηση (deep learning) [2] είναι νέος κλάδος της Μηχανικής Μάθησης που αφορά αλγορίθμους με πολλαπλά επίπεδα αφαιρετικότητας δεδομένων [2]. Όσο λιγότερο περίπλοκα είναι τα δεδομένα τόσο πιο εύκολη θα είναι και η μάθηση. Για αυτό τον λόγο γίνεται επεξεργασία των δεδομένων για μείωση της πολυπλοκότητας [1]. Η βαθιά μάθηση έχει τις ρίζες της στις έρευνες των τεχνητών νευρωνικών δικτύων. Την συναντάμε δε και με τις εξής ονομασίες, βαθιά δομημένη μάθηση (deep structured learning) ή ιεραρχική μάθηση (hierarchical learning) [2].

Ας διαφοροποιήσουμε την μηχανική μάθηση από την βαθιά μάθηση. Η μηχανική μάθηση χρησιμοποιεί ρηχές αρχιτεκτονικές για επίλυση απλών προβλημάτων. Με τον όρο ρηχές εννοούμε την χρήση το πολύ δύο στρωμάτων από μη γραμμικούς μετασχηματισμούς χαρακτηριστικών. Ενώ η βαθιά αρχιτεκτονική μπορεί να λύσει πιο περίπλοκα προβλήματα με πιο περίπλοκα και περισσότερα δεδομένα. Γιατί τα ονομάζουμε περίπλοκα προβλήματα; Επειδή προσπαθούν να επεξεργαστούν διάφορα φυσικά σήματα, τα οποία είναι πιο περίπλοκα δεδομένα, όπως η ομιλία, η γλώσσα, η εικόνα, ο ήχος [2]. Δηλαδή μπορούμε να πούμε ότι η βαθιά μάθηση προσπαθεί να μιμηθεί τον ανθρώπινο εγκέφαλο [1]. Ωστόσο είναι δύσκολη η αυτοματοποίηση αυτών τον προβλημάτων, για τον λόγο ότι τα προβλήματα αυτά είναι διαισθητικά. Με τον όρο διαισθητικά εννοούμε ότι είναι εύκολο για τους ανθρώπους να πράξουν μια εργασία, αλλά είναι δύσκολο να περιγράψουν την διαδικασία που τους οδήγησε σε αυτήν την πράξη. Τα αρχικά προβλήματα που επιλύθηκαν στο τομέα της ΤΝ υποστήριζαν ακριβώς το αντίθετο, δηλ. ενώ ήταν δύσκολο για τον άνθρωπο να τα λύσει, για τον υπολογιστή ήταν απλή εκτέλεση κανόνων. Παραδείγματα τέτοιων προβλημάτων, είναι τα παιχνίδια που έχουν απλούς κανόνες εφαρμογής (π.χ. το σκάκι). Αντίθετα στα προβλήματα της βαθιάς μάθησης δεν καθορίζεται όλη η γνώση που θα χρειαστεί ο υπολογιστής σε κανόνες, αλλά όπως αναφέραμε και πριν, ο υπολογιστής μέσα από την εμπειρία προσπαθεί να κατανοήσει περίπλοκες έννοιες με την μετατροπή τους σε απλούστερες [4]. Λόγω αυτής της ιεραρχίας εννοιών/χαρακτηριστικών προκύπτει και ο όρος βαθιά μάθηση [2]. Μπορούμε να συσχετίσουμε αυτήν την εμπειρική μάθηση με την λειτουργία του νεοφλυού. Ο νεοφλυός, που είναι το εξωτερικό στρώμα του εγκεφάλου, σχετίζεται με τις επίκτητες ικανότητες. Αφήνει τα σήματα από τους αισθητήρες να διέλθουν στα πιο κάτω στρώματα του εγκεφάλου, χωρίς να τα επεξεργαστεί και μετά από μια περίοδο εμπειριών καταφέρνει να αναπαριστά τις διάφορες παρατηρήσεις με βάση τις ομοιότητες τους. Ένα παράδειγμα επίκτητης ικανότητας είναι το εξής [3]. Ο άνθρωπος την πρώτη φορά που βλέπει φωτιά δεν μπορεί να ξέρει ότι αν την ακουμπήσει θα καεί. Θα πρέπει να το πάθει, να νιώσει τον πόνο του εγκαύματος και έτσι την επόμενη φορά που θα ξανασυναντήσει φωτιά, ο εγκέφαλος θα του δώσει την εντολή να την αποφύγει.

Όπως αναφέραμε όταν ένα πρόγραμμα απαιτεί αυστηρά οριοθετημένες γνώσεις, τότε οι γνώσεις αυτές μπορούν να περιγραφούν στον υπολογιστή με απλούς κανόνες. Αντίθετα στην περίπτωση των προβλημάτων βαθιάς-μηχανικής μάθησης, όπου υπάρχει αρκετή περισσότερη και διαισθητική γνώση, είναι δύσκολο να περιγραφεί στον υπολογιστή υπό την μορφή απλών κανόνων. Αρχικά για περιγραφή απλών κανόνων στον υπολογιστή χρησιμοποιήθηκαν μηχανές που χρησιμοποιούν λογικούς κανόνες συμπερασμού, όπως η Cyc (Lenatand Guha, 1989). Η διαδικασία για την πιο πάνω μηχανή είναι η εξής: ένας άνθρωπος καταγράφει διάφορες δηλώσεις, υπό την μορφή της γλώσσας της μηχανής, για να περιγράψει τον πραγματικό κόσμο και η μηχανή προσπαθεί να καταλάβει και να βγάλει διάφορα συμπεράσματα με βάση αυτές τις δηλώσεις. Αυτή η μηχανή παρουσίαζε αδυναμίες, λόγω λανθάνων συμπερασμάτων που

παρήγαγε. Έτσι ανακαλύφθηκε η μηχανική μάθηση όπου ο υπολογιστής μέσω ανεπεξέργαστων δεδομένων που λαμβάνει, παράγει την δική του γνώση, με λιγότερη ανθρώπινη παρέμβαση. Σε αυτούς τους αλγορίθμους μηχανικής μάθησης υπάρχει η δυσκολία να αποφασιστεί σε ποια αναπαράσταση θα δοθούν τα δεδομένα. Γιατί για τον ίδιο αλγόριθμο αν του αλλάξουμε την αναπαράσταση δεδομένων που δέχεται το πιο πιθανόν να αλλάξει και η αποδοτικότητα του. Όταν λέμε αναπαράσταση δεδομένων, εννοούμε της μορφές των δεδομένων, π.χ. υπό την μορφή εικόνας, κειμένου κτλ. Η επιλογή της σωστής αναπαράστασης παίζει μεγάλο ρόλο στην αποδοτικότητα των αλγορίθμων. Για παράδειγμα οι άνθρωποι βρίσκουν πιο εύκολα το αποτέλεσμα μιας πράξης με αριθμούς στο δεκαδικό σύστημα, παρά στο δυαδικό. Εκτός από την κατάλληλη αναπαράσταση των δεδομένων, πρέπει να σκεφτούμε και την χρησιμότητα τους για συγκεκριμένο αλγόριθμο. Σε πολλές περιπτώσεις είναι δύσκολο να αποφασίζουμε ποια δεδομένα θα φανούν χρήσιμα για κάποιον αλγόριθμο. Για παράδειγμα η λογιστή παλινδρόμηση είναι ένας αλγόριθμος μηχανικής μάθησης ο οποίος μπορεί να προτείνει αν θα συστηθεί ή όχι η καισαρική τομή. Γι αυτόν τον αλγόριθμο ένα χρήσιμο δεδομένο θα ήταν αν στην μήτρα υπάρχει ουλή. Ακόμα ένα παράδειγμα αλγορίθμου, είναι ο αλγόριθμος που αναγνωρίζει το φύλο ενός ατόμου από την φωνή του. Ένα χρήσιμο δεδομένα σε αυτόν τον αλγόριθμο είναι η ένταση της φωνής. Αλγόριθμοι αναπαράστασης μάθησης μπορούν να βρουν χαρακτηριστικά και τις αναπαραστάσεις τους ακόμα και για πιο σύνθετες εφαρμογές. Όπως η ανίχνευση ενός αυτοκινήτου σε μια φωτογραφία. Ένα χρήσιμο δεδομένο θα ήταν η παρουσία ενός τροχού, αλλά ο τροχός σε μορφή pixels είναι δύσκολο να εντοπιστεί, λόγω των σκιών, του ήλιου και άλλων αντικειμένων που εμποδίζουν στον εντοπισμό του τροχού στην φωτογραφία [4].

Οι αρχιτεκτονικές μηχανικής μάθησης και ως εκ τούτου και βαθιάς μάθησης, αποτελούνται από πολλαπλά επίπεδα μη γραμμικής επεξεργασίας δεδομένων.Οι αρχιτεκτονικές αυτές κατατάσσονται σε τρεις κατηγορίες ανάλογα με την τελική τους χρήση. Βαθιά δίκτυα για μη επιβλεπόμενη μάθηση, Βαθιά δίκτυα για επιβλεπόμενη μάθηση και Υβριδικά βαθιά δίκτυα. Βαθιά δίκτυα χωρίς-επίβλεψη λαμβάνουν ορατά ή παρατηρήσιμα δεδομένα. Προσπαθούν να συσχετίσουν αυτά τα δεδομένα είτε για να πάρουν χρήσιμες πληροφορίες, είτε για την ανάλυση κάποιου μοτύβου-προτύπου. Συνήθως δεν υπάρχουν δεδομένα εξαρχής χωρισμένα σε κλάσεις. Βαθιά δίκτυα με εποπτευόμενη μάθησης μπορούν να διαχωρίσουν άμεσα τα ορατά δεδομένα σε κλάσεις, αφού η μάθηση είναι εποπτευόμενη.

Τα Υβριδικά δίκτυα είναι ο συνδυασμός των δύο πιο πάνω. Υπάρχουν πλεονεκτήματα και στα εποπτευόμενα μοντέλα μάθησης αλλά και στα μη εποπτευόμενα. Ας δώσουμε ένα παράδειγμα για να διαχωρίσουμε τις πιο πάνω κατηγορίες. Ένα πρόγραμμα προσπαθεί να ανακαλύψει σε μια φωτογραφία ποιος είναι ο σκύλος και ποια η γάτα. Όταν το πρόγραμμα έχει φωτογραφίες με τον σκύλο και την γάτα και άρα γνωρίζει πως μοιάζουν, τότε είναι με επίβλεψη. Ενώ όταν δεν έχει αυτές τις πληροφορίες, είναι χωρίς επίβλεψη [2].

Οι πιο πάνω αρχιτεκτονικές έχουν εφαρμοστεί σε πολλές εφαρμογές όπως την εφαρμογή ανίχνευσης προσώπου, αυτόματη αναγνώριση και ανίχνευση ομιλίας, ρομποτική, επεξεργασία γλώσσας, αναγνώριση αντικειμένων, υπολογιστική όραση, επεξεργασία κειμένου κτλ [4].

2.2 Τεχνητά Νευρωνικά Δίκτυα

Τα τεχνητά νευρωνικά δίκτυα προσπαθούν να μιμηθούν τον τρόπο με τον οποίο ο ανθρώπινος εγκέφαλος λειτουργεί και λαμβάνει αποφάσεις [52]. Δεν έχει βρεθεί ακόμα η ακριβής προσομοίωση για το πώς λειτουργεί ο εγκέφαλος, γι 'αυτό τα δίκτυα αυτά είναι λιγότερο περίπλοκα απ΄όσο είναι στην πραγματικότητα ο εγκέφαλος.

Δύο σχετικά νέοι τύποι νευρωνικών δικτύων είναι τα Συνελικτικά Νευρωνικά δίκτυα (Convolutional Neural Networks-CNNs) και τα Δίκτυα Βαθιάς Πίστης (Deep Belief Networks - DBNs), στα οποία θα αναφερθούμε και πιο κάτω. Αυτά είναι και τα δύο σημαντικά σημεία τα οποία δείχνουν προς τα που θα κινηθεί η Βαθιά Μάθηση.

Τα τεχνητά νευρωνικά δίκτυα είναι δίκτυα που αποτελούνται από διασυνδεδεμένους νευρώνες. Το νευρωνικό δίκτυο αποτελείται από τρία επίπεδα, το επίπεδο εισόδου (input layer), το κρυφό επίπεδο (hidden layer) και το επίπεδο εξόδου (output layer). Οι νευρώνες των τριών επιπέδων ονομάζονται νευρώνες εισόδου, κρυμμένοι ή υπολογιστικοί νευρώνες και νευρώνες εξόδου αντίστοιχα. Οι νευρώνες εισόδου δέχονται τις περιβαλλοντικές εισόδους του δικτύου (π.χ. νευρώνες που κωδικοποιούν τις τιμές των εικονοστοιχείων (pixels) της εικόνας εισόδου). Οι κρυμμένοι νευρώνες πολλαπλασιάζουν κάθε είσοδο τους με το αντίστοιχο συναπτικό βάρος και υπολογίζουν το ολικό άθροισμα των γινομένων. Το άθροισμα αυτό δίνεται ως είσοδος στην συνάρτηση ενεργοποίησης του νευρώνα. Και τέλος οι νευρώνες εξόδου δίνουν στο περιβάλλον τις τελικές αριθμητικές εξόδους του δικτύου. Η έξοδος y_k ενός νευρώνα k δίνεται από την εξής εξίσωση:

$$y_k = \varphi\left(\sum_{i=0}^{N} x_{ki} w_{ki}\right)$$

όπου x_{ki} είναι η i-οστή είσοδος του k νευρώνα, το w_{ki} είναι το i-οστό συναπτικό βάρος του k νευρώνα και φ είναι η συνάρτηση ενεργοποίησης του νευρώνα [68,69].

2.3 Φυσιολογικά Σήματα

Μια νέα πρόκληση στην Επιστήμη της Τεχνητής Νοημοσύνης είναι η επεξεργασία ενός μεγάλου όγκου από πολύπλοκα δεδομένα, όπως είναι τα φυσιολογικά σήματα. Το χαρακτηριστικό στοιχείο τους είναι η υψηλή διάσταση των δεδομένων σε σύγκριση με το μικρό πλήθος των δειγμάτων. Για παράδειγμα, ένα σύνολο δεδομένων με μετρήσεις ECG που χρησιμοποίησα στην μελέτη μου, περιείχε 71 ασθενείς και το κάθε δείγμα περιείχε περίπου 4,000,000 μετρήσεις.

Υπάρχει πλήθος φυσιολογικών σημάτων που διεγείρονται ακούσια από το Αυτόνομο Νευρικό Σύστημα κατά την παρουσία διάφορων εσωτερικών ή εξωτερικών ερεθισμάτων. Παραδείγματα τέτοιων σημάτων είναι, ο καρδιακός ρυθμός, η στατική αντίδραση δέρματος και το ηλεκτροεγκεφαλογράφημα. Στην παρούσα εργασία θα αναλυθούν τα εξής φυσιολογικά σήματα με την βοήθεια της Μηχανικής Μάθησης: ECG (electrocardiography), SCR (skin conductance response) και COR (corrugator muscle).

2.3.1 Ηλεκτροεγκεφαλογράφημα

Αναπόφευκτα ο άνθρωπος αλληλεπιδρά με μηχανές κατά την διάρκεια της ζωής του για την διεκπεραίωση διάφορων εργασιών. Συνήθως η αλληλεπίδραση αυτή περιορίζεται στο να περνά ο άνθρωπος δεδομένα στη μηχανή με άμεση φυσική κίνηση (π.χ. μέσω κουμπιών, πλήκτρων κτλ) [10]. Σήμερα υπάρχει μια μέθοδος επικοινωνίας με χρήση της νευρωνικής δραστηριότητας που παράγεται από τον εγκέφαλο, η οποία λέγεται Brain Computer Interface - BCI. Έτσι επιτρέπεται η επικοινωνία με μηχανές, τεχνητά μέλη, ρομπότ χωρίς την άμεση παρέμβαση του ανθρώπου όπως αναφέραμε πιο πάνω [11]. Το BCI συνδυάζει πολλές επιστήμες, την ιατρική, τη νευρολογία, την ψυχολογία, τη μηχανική αποκατάστασης, Αλληλεπίδραση

Ανθρώπου-Υπολογιστή (HCI-Human-Computer Interaction), επεξεργασία σήματος και μηχανική μάθηση [9].

Η επικοινωνία μεταξύ ανθρώπου με άνθρωπο και ανθρώπου με μηχανή έχει μια σημαντική διαφορά, στην πρώτη μπορούν να μοιράζονται τα συναισθήματά τους, ενώ στην δεύτερη μια μηχανή δεν μπορεί να αισθανθεί. Το συναίσθημα επηρεάζει διάφορες νοητικές λειτουργίες, όπως τη σωστή λήψη αποφάσεων, τη κατανόηση, τη γνώση, τη μάθηση, την επικοινωνία κτλ. Άρα είναι σημαντικό να μπορεί να αισθάνεται και μια μηχανή έτσι ώστε να βελτιωθεί η επικοινωνία μεταξύ ενός ανθρώπου και μιας μηχανής. Εν καιρό υπήρξαν πολλές μέθοδοι για την αναγνώριση του συναισθήματος, οι οποίοι αποδείχθηκαν αναποτελεσματικοί. Ένας από αυτούς είναι η έκφραση του προσώπου. Αν και οι εκφράσεις είναι ένα βασικό σημείο επικοινωνίας ώστε οι άνθρωποι να δείχνουν τα συναισθήματά τους, μια έκφραση μπορεί να ψεύδεται, δηλ. κάποιος μπορεί άλλο να αισθάνεται και άλλο να δείχνει προς τα έξω. Ένας άλλος τρόπος είναι η τάση στη φωνή ή οι κινήσεις. Αργότερα ανακαλύφθηκαν πιο ακριβές μέθοδοι, τα φυσιολογικά σήματα, όπως είναι ο καρδιακός ρυθμός (ECG-electrocardiography), η Galvanic skin response (GSR), διαστολή της κόρης και το ηλεκτροεγκεφαλογράφημα (EEG-electroencephalography)[10]. Η πιο αποδοτική τεχνική και αυτή που χρησιμοποιήθηκε περισσότερο σε μελέτες από της πιο πάνω είναι η χρήση του EEG. Ένα από τα παραδείγματα που δείχνει την καταλληλότητα ενός συστήματος με χρήση EEG, ήταν αυτό που πρότεινε ο Kim et al, το οποίο ανίχνευε τέσσερα διαφορετικά συναισθήματα θυμό, θλίψη, άγχος και έκπληξη. Η επιτυχία αυτού του συστήματος με δείγμα 50 ατόμων ήταν 78,4% για ανίχνευση τριών διαφορετικών συναισθημάτων και 61,8% για ανίχνευση τεσσάρων [15]. Το EEG ανακαλύφθηκε περίπου 80 χρόνια πριν από τον Hans Berger, ένα Γερμανό επιστήμονα [13]. Το ηλεκτροεγκεφαλογράφημα (EEG) καταγράφει την ηλεκτρική δραστηριότητα του εγκεφάλου με τη χρήση ηλεκτροδίων που είναι ομοιόμορφα διατεταγμένα στο τριχωτό της κεφαλής. Αυτή η ηλεκτρική δραστηριότητα απεικονίζεται ως διακύμανση τάσης που προκύπτει από την ροή ρεύματος μέσα στους νευρώνες του εγκεφάλου [10].

Θεωρητικά υπάρχουν δύο τρόποι για την ταξινόμηση των συναισθημάτων. Ο πρώτος είναι ότι ορίζονται κάποια βασικά συναισθήματα και με την σύνθεση τους να δημιουργηθούν πιο περίπλοκα συναισθήματα, όπως το λευκό χρώμα είναι η ένωση όλων των βασικών χρωμάτων. Ο δεύτερος τρόπος είναι το μοντέλο διαστάσεων, το οποίο αποτελείται από συνήθως δύο διαστάσεις, το σθένος και την διέγερση. Το σθένος αντιπροσωπεύει το θετικό και αρνητικό συναίσθημα και η διέγερση αντιπροσωπεύει τον ενθουσιασμό ή τον μη-ενθουσιασμό για κάτι. Κάθε συναίσθημα μπορεί να εκπροσωπηθεί από αρνητικό ή θετικό σθένος και υψηλή ή χαμηλή διέγερση [15].

Υπάρχουν πολλές εφαρμογές με βάση το συναίσθημα και πιο γενικά BCI εφαρμογές με χρήση EEG:

- Ειδικές ανάγκες:

Ένας άνθρωπος με ειδικές ανάγκες δυσκολεύεται στη χρήση μηχανημάτων που είναι κατασκευασμένα για υγιείς ανθρώπους. Έτσι ένα μηχάνημα με αξιολόγηση συναισθημάτων, θα έκανε την ζωή αυτών τον ανθρώπων πιο εύκολη [15].

- Αυτισμός:

Άνθρωποι που υποφέρουν από αυτισμό δυσκολεύονται να δείξουν τα συναισθήματα τους. Έτσι το μηχάνημα θα ανιχνεύει αυτά τα συναισθήματα και θα τα αποτυπώνει έτσι ώστε να μπορούν να είναι πιο εύκολη η επικοινωνία αυτών των ανθρώπων με τους γύρω τους [10].

- Επιληπτική κρίση:

Σε ανθρώπους με επιληπτική κρίση θα ήταν χρήσιμη μια εφαρμογή για να ανιχνεύει την έναρξη επιληπτικής κρίσης πριν να γίνει, αποφεύγοντας έτσι δυσάρεστους τραυματισμούς ή ακόμα και τον θάνατο λόγω των σπασμών που δημιουργούνται. Η εφαρμογή αυτή χρησιμοποιεί EEG, λόγω του ότι η επιληπτική κρίση προκαλεί διαταραχές στην ηλεκτρική δραστηριότητα του εγκεφάλου και έτσι αυτό την ανιχνεύει. Φυσικά δεν σχετίζονται όλες οι ανωμαλίες στη ρυθμική δραστηριότητα με την επιληπτική κρίση, αλλά για παράδειγμα μπορεί να σχετίζεται με τον ύπνο. Με την εφαρμογή αυτή θα μπορεί να να ενημερωθεί άμεσα ο φροντιστής πριν την έναρξη της επιληπτικής κρίσης ή ακόμα η εφαρμογή να παρέχει κάποια θεραπεία.Υπάρχει όμως ένα μειονέκτημα στη χρήση EEG, ότι τα τα χαρακτηριστικά ενός EEG με επιληπτική κρίση διαφέρουν από άτομο σε άτομο. Ακόμα και χαρακτηριστικά που για ένα άτομο δείχνουν την έναρξη επιληπτικής κρίσης, τα ίδια χαρακτηριστικά σε άλλο άτομο μπορούν να δείχνουν ακριβώς το αντίθετο. Φυσικά οι επιληπτικές κρίσης ενός συγκεκριμένου ατόμου έχουν συνοχή αν-ν προέρχονται από την ίδια περιοχή του εγκεφάλου [20].

- Εφαρμογές με νευρωνικά σήματα:

Υπάρχουν εφαρμογές για την ανίχνευση των διαταραχών του ύπνου, για νευρολογικές ασθένειες και γενικά για διάφορα θέματα παρακολούθησης του ανθρώπου. Ακόμα υπάρχουν εφαρμογές για την παρακολούθηση της συμπεριφοράς του ανθρώπου με βάση τα νευρωνικά σήματα [13].

- Εφαρμογές για διασκέδαση:

Εκτός από εφαρμογές που βοηθούν τους ανθρώπους, υπάρχουν και εφαρμογές για την διασκέδαση του, όπως είναι τα βιντεοπαιχνίδια [9].

2.3.2 Στατική αντίδραση δέρματος

Το σήμα της στατικής αντίδραση δέρματος (GSR-Galvanic Skin Response) είναι οι αλλαγές στις ηλεκτρικές ιδιότητες του δέρματος [31], δηλαδή το δέρμα γίνεται στιγμιαία καλύτερος αγωγός του ηλεκτρισμού [25]. Οι αλλαγές αυτές προκαλούνται-διεγείρονται λόγω διάφορων εσωτερικών ή εξωτερικών ερεθισμάτων, όπως η ψυχολογική κατάστασης (π.χ. το άγχος) ενός ατόμου και τα διάφορα γεγονότα(π.χ. θερμοκρασία, άσκηση) που διαδραματίζονται γύρω του [34]. Το GSR προκαλείται από το συμπαθητικό νευρικό σύστημα (SNS) ως αντίδραση των πιο πάνω ερεθισμάτων. Το (SNS) επηρεάζεται από δομές του εγκεφάλου που ασχολούνται με το συναίσθημα, όπως ο υποθάλαμος και το μεταιχμιακό σύστημα. Έτσι το σήμα GSR χρησιμοποιείται πολύ σε έρευνες σχετικά με συναισθήματα, συγκεκριμένα με τις αλλαγές διέγερσης. Η διέγερση θεωρείται μια από τις δύο διαστάσεις του συναισθηματικού μοντέλου διαστάσεων και αντιπροσωπεύει πόσο υψηλό ή χαμηλό ενθουσιασμό έχει κάποιος [33]. Πρέπει να επισημανθεί ότι η μέτρηση διέγερσης παίρνει ένα σημαντικό μέρος της μέτρησης συναισθήματος, αλλά όχι εξολοκλήρου. Η διέγερση τείνει χαμηλή όταν το άτομο κοιμάται και ψηλή όταν το άτομο βρίσκεται σε καταστάσεις ενέργειας, όπως όταν αντιμετωπίζει πολλή φόρτο εργασίας, όταν θυμώσει, όταν συγκινείται ή ακόμα και όταν έχει να λύσει κάποιο πρόβλημα [25]. Λόγω των διάφορων παραγόντων που ψηλώνουν την διέγερση, είναι δύσκολο για ένα σύστημα να βρει τον ακριβή λόγο. Υπάρχουν δύο κατηγορίες ανθρώπων ως προς τα επίπεδα σημάτων της αγωγιμότητάς τους. Αυτοί που τα σήματα αγωγιμότητας τους έχουν μικρή διαφορά όταν βρίσκονται σε ήρεμη κατάσταση σε σύγκριση με το όταν βρίσκονται σε κάποια ένταση. Από την άλλη υπάρχουν και αυτοί που έχουν συχνές στατικές αντιδράσεις του δέρματος ακόμα και όταν βρίσκονται σε ήρεμη κατάσταση [34].

Οι ηλεκτρικές αλλαγές στο δέρμα προκαλούνται από αυξημένη συναισθηματική διέγερση ή νοητικό φόρτο εργασίας που οδηγεί σε κόπωση ή έκκριση ιδρώτα [27]. Υπάρχουν φυσικοί τρόποι, έτσι ώστε μπορέσει κάποιος να ελέγξει την αγωγιμότητα του δέρματος, ώστε να βρίσκεται σε λογικά πλαίσια. Για παράδειγμα η σκέψη ευχάριστων και ηρεμιστικών στιγμών. Είναι όμως ακατόρθωτο για τους περισσότερους ανθρώπους μια τέτοια φυσική αντιμετώπιση, προκαλώντας τους ψυχικές διαταραχές [25]. Οι ψυχικές διαταραχές (όπως το υπερβολικό άγχος) ενός ατόμου μπορεί να προκαλέσουν ακόμα και σωματικές διαταραχές. Αυτό οφείλεται στον εξής λόγο, όταν προκαλούνται ψυχικές διαταραχές μπορεί να προκληθεί δυσλειτουργία του αυτόματου νευρικού μας συστήματος (ANS-autonomic nervous system), ενδοκρινικού μας συστήματος και του ανοσοποιητικού συστήματος, μέσω της επίδρασης των νευρολογικών μονοπατιών [34].

Κάθε συναίσθημα μπορεί να συνδεθεί με ένα συγκεκριμένο ερέθισμα προκαλώντας την επίτευξη ή την αποτυχία ενός στόχου. Οι στόχοι αυτοί μπορεί να είναι από τους πιο απλούς,

βασικούς στόχους επιβίωσης ως τους πιο περίπλοκους. Για παράδειγμα το συναίσθημα της λύπης μπορεί να συνδεθεί με ένα δυσάρεστο ερέθισμα που προκαλεί αρνητικές επιπτώσεις στο στόχο της επιβίωσης [34].

Έχουν γίνει πολλές έρευνες γύρω από την εύρεση φυσιολογικών προτύπων για τα βασικά συναισθήματα. Οι έρευνες αυτές βασίζονται στις δραστηριότητες του αυτόνομου νευρικού συστήματος. Ωστόσο για μερικά συναισθήματα έχουν βρεθεί διαφορετικά πρότυπα από κάθε έρευνα. Αυτό μπορεί να συμβαίνει διότι οι μεταβλητές που μετρήθηκαν σε μια έρευνα δεν οδηγούν σε σαφή διαχωρισμό των διαφορετικών συναισθημάτων. Μερικά πρότυπα που ήταν κοινά σε έρευνες είναι τα εξής. Το πρότυπο του φόβο αποτελείται από τις εξής μεταβλητές, την αύξηση του καρδιακού ρυθμού, το επίπεδο αγωγιμότητας του δέρματος, τη συστολή της αρτηριακής πίεσης. Οι μεταβλητές στο πρότυπο της οργής είναι, η αύξηση του καρδιακού ρυθμού και η συστολή και διαστολή της αρτηριακής πίεσης. Ένα από τα πρότυπα που δίχασαν τις έρευνες είναι αυτό της θλίψης. Σε μερικές έρευνες βρήκαν μείωση του καρδιακού ρυθμού, σε άλλες αύξηση του καρδιακού ρυθμού και σε άλλες δεν μπόρεσαν να βρουν καν την διαφορά με έναν φυσιολογικό καρδιακό ρυθμό. Ακόμα ένα αμφιλεγόμενο πρότυπο είναι αυτό των θετικών συναισθημάτων, όπου σε διαφορετικές έρευνες βρέθηκε είτε μείωση είτε αύξηση των εξής μεταβλητών, του καρδιακού ρυθμού, της συστολής και διαστολή της αρτηριακής πίεσης. Τέλος, το πιο δύσκολο πρότυπο για να αποφασιστεί είναι αυτό της αηδίας. [34]

Το GSR (γαλβανική αντίδραση του δέρματος) είναι η πιο παλιά ορολογία για το αναφερόμενο σήμα. Υπάρχουν πολλοί άλλοι πιο πρόσφατοι όροι -ονομασίες για το σήμα GSR, όπως EDA (ηλεκτροδερμική δραστηριότητα), EDR (ηλεκτροδερμική απόκριση), EDL (ηλεκτροδερμικό επίπεδο), SCA (δραστηριότητα αγωγιμότητας του δέρματος), SCR (απάντηση αγωγιμότητας του δέρματος) κτλ. Έτσι βλέπουμε ότι το GSR είναι ένα πολυδιάστατο σήμα το οποίο μπορεί να μετρηθεί χρησιμοποιώντας ηλεκτρο-φυσιολογικές μετρήσεις, όπως ECG ή EMG (ηλεκτρομυογράφημα) ή και συνδυασμό των δύο μεθόδων μέτρησης [31]. Σε μια έρευνα μέτρησης του GSR, τα δεδομένα συλλέγονται κατά την διάρκεια που ένας άνθρωπο αλλάζει τα συναισθήματα του επηρεασμένος από τις εξωτερικές συνθήκες (π.χ. δείχνοντάς του εικόνες). Έτσι με την παρακολούθηση του GSR μπορεί να ανιχνευτούν διάφορα συναισθήματα, ενθουσιασμός, άγχος, ενδιαφέρον κτλ [34].

Ένα παράδειγμα συσκευής με χρήση GSR είναι ο αισθητήρας άγχους. Είναι ένα μηχάνημα που ανιχνεύει αυτόματα το άγχος ώστε να βοηθά τους ανθρώπους να βρουν μια ισορροπία άγχους στη ζωή τους [30]. Το άγχος έρχεται σε μια δύσκολη κατάσταση που πρέπει να αντιμετωπίσει ένα άτομο. Όταν το νευρικό μας σύστημα αντιλαμβάνεται το άγχος απελευθερώνει ορμόνες άγχους, όπως η αδρεναλίνη και κορτιζόλη. Εκτός από το ότι το άγχος μπορεί να δυσκολέψει το άτομο να φτάσει στον στόχο του, μπορεί να του αποφέρει και προβλήματα υγείας όπως καρδιαγγειακές παθήσεις, αγγειακή κεφαλική νόσο, διαβήτη και ανοσοανεπάρκειες [35].

Ο αισθητήρας άγχους δουλεύει τοποθετώντας δύο ηλεκτρόνια στα δάκτυλα ενός ατόμου και ανιχνεύει το άγχος λόγω της διαφορετικής αγωγιμότητας του δέρματος όταν υπάρχει ή όταν απουσιάζει το άγχος [35]. Αφού συλλέξαμε χαρακτηριστικά GSR, με τον αισθητήρα ακολούθως τα δίνουμε ως είσοδο σε ένα κατηγοριοποιητή για εκπαίδευση. Συγκεκριμένα ο SVM κατηγοριοποιητής ανιχνεύει το άγχος με 92% γενικευμένη ακρίβεια. Ένα πλεονέκτημα των GSR χαρακτηριστικών είναι ότι επιτρέπουν σε έναν κατηγοριοποιητή να γενικεύσει και να μην εξαρτάται από συγκεκριμένα άτομα. Ενώ αν ο κατηγοριοποιητής παίρνει ως είσοδο χαρακτηριστικά από ομιλίες μπορεί να έχουν μεγαλύτερη προβλεπόμενη ισχύ από τα χαρακτηριστικά GSR, αλλά δεν έχουν το πιο πάνω πλεονέκτημα. Ένας τρόπος για να δεις αν ένα άτομο έχει άγχος μέσω τις ομιλίας, είναι η συχνότητα των αναπνοών του. Όταν κάποιος έχει άγχος οι αναπνοές γίνονται με πιο έντονα ρυθμό και αυτό επηρεάζει και την ομιλία, αφού παραμένει λιγότερος χρόνος μεταξύ των αναπνοών για κάποιο να μιλήσει. Ένας παρόμοιος τρόπος ανίχνευσης άγχους, παρόμοιος με αυτό την ομιλίας είναι η ανίχνευση άγχους από την φωνή. Αυτό γίνεται μετρώντας το πόσο έντονα συσπώνται οι μύες στο σώμα ενός ατόμου, για παράδειγμα αν υπάρχει τρέμουλο στις φωνητικές χορδές. Επίσης εκτός από το GSR ας αναφέρουμε και άλλα φυσιολογικά σήματα που βοηθούν στην ανίχνευση του άγχους, όπως η ηλεκτρική δραστηριότητα στο τριχωτό της κεφαλής, η πίεση του αίματος (BP), ο όγκος του αίματος Pulse-παλμός (BVP), Blood Pressure (BP), Blood Volume
Pulse (BVP) [30].

2.3.3 Ηλεκτροκαρδιογράφημα

Ένα ηλεκτροκαρδιογράφημα (ECG) μετρά την ηλεκτρική δραστηριότητα της καρδιάς [39]. Είναι ένα μη επεμβατικό μέσο για να διαγνωστούν διάφορες καρδιαγγειακές παθήσεις (CVD), όπως Αρρυθμία, κολπική μαρμαρυγή, κολποκοιλιακός (AV) δυσλειτουργίες, και της στεφανιαίας αρτηριακής νόσου, κλπ [47]. Καταγράφεται με την χρήση ηλεκτροδίων που τοποθετούνται επιφανειακά σε διάφορα μέρη του ανθρώπου, κυρίως στα χέρια, στο στήθος ή στα πόδια [37].

Λόγω του μεγάλου όγκου δεδομένων που συλλέγονται από σήματα ECG είναι αδύνατον οι άνθρωποι να εξάγουν γνώση από αυτά σε σύντομο χρονικό διάστημα [49]. Σε πολλές έρευνες χρησιμοποιούνται τα δεδομένα από την βάση δεδομένων αρρυθμίας της MIT (MIT BIH Arrhythmia Database) τα οποία περιέχουν 48 αποσπάσματα από καταγραφές ECG ασθενών [38]. Είναι δύσκολο μια μεγάλη ποσότητα δεδομένων να αναλυθεί με το χέρι, γι' αυτό υπάρχουν διαφορετικά είδη υπολογιστικών αλγορίθμων που χρησιμοποιούνται για να αναλύσουν το σήμα ECG για την διάγνωση CVD. Ένας τέτοιου είδους αλγόριθμος είναι αυτός που βασίζεται στον ταξινομητή σημάτων ECG. Κάθε παλμός του ECG σήματος απεικονίζεται με τη μορφή ηλεκτρικού κύματος με κορυφές και κοιλάδες [37]. Από αυτή την κυματομορφή, ένας

ταξινομητής ECG εξάγει σημαντικά χαρακτηριστικά σχετικά με τον ρυθμό και την λειτουργία της καρδιάς. Υπάρχουν δύο τρόποι εξαγωγής χαρακτηριστικών από τα δεδομένα ενός σήματος ECG, είτε τα χαρακτηριστικά εξάγονται άμεσα από τα δεδομένα αυτά, είτε εξάγονται με την βοήθεια επεξεργασίας σήματος. Ο ταξινομητής εκπαιδεύεται με την βοήθεια των χαρακτηριστικών αυτών και αξιολογείται κατά πόσο είναι επιτυχής στις διαγνώσεις του. Φυσικά τα χαρακτηριστικά τα οποία δεν βοηθούν στην ακρίβεια ταξινόμησης διαγράφονται [47].

Όπως αναφέραμε και πιο πάνω η αρρυθμία είναι ένα είδος CVD. Η αρρυθμία μπορεί να οριστεί ως οι ανωμαλίες στον ECG παλμό του ανθρώπου. Υπάρχουν πολλών ειδών αρρυθμίες, όπως η καρδιακή αρρυθμία, κοιλιακή μαρμαρυγή, πτερυγισμό κοιλίας κτλ. Αν η αρρυθμία δεν θεραπευτεί άμεσα υπάρχει κίνδυνος ο ασθενής να οδηγηθεί σε καρδιακή ανακοπή, αιμοδυναμική κατάρρευση ή ακόμα και σε ξαφνικό θάνατο [51]. Οι πιο θανατηφόρες αρρυθμίες είναι οι κοιλιακές, όπως κοιλιακή ταχυκαρδία (VT) και η κοιλιακή μαρμαρυγή (VF). Υπάρχουν φυσικά και εξαιρέσεις που δεν είναι τόσο σοβαρές, αλλά παρουσιάζουν διαταραχές στην καρδιά, όπως η κοιλιακή πρόωρη συστολή (PVC) [49].

Μια προσέγγιση για την διάγνωση αρρυθμίας είναι αυτή της ταξινόμησης παλμών. Όπως αναφέραμε και πιο πάνω η αρρυθμία αποτελείται από ανώμαλους παλμούς. Γι' αυτό σε αυτή την προσέγγιση γίνεται διαχωρισμός των καταγραφών ECG στα συστατικά παλμών και μετά ταξινόμηση παλμών σε φυσιολογικούς και ακανόνιστους. Ένα παράδειγμα μοντέλου που ακολουθεί αυτήν την μέθοδο, είναι το μοντέλο RST. Αυτό το μοντέλο δημιουργήθηκε για ένα μεμονωμένο ασθενή που παρακολουθείται συχνά για αρρυθμίες και ως δεδομένα στο σύστημα μπαίνουν δεδομένα για τον ασθενή που έχουν εντοπιστεί από τον καρδιολόγο. Ένα άλλο πιο γενικό μοντέλο παίρνει ως είσοδο, εγγραφές ECG ως μια σειρά παλμών [38].

Το μοντέλο για να μπορεί να ανιχνεύσει τους ανώμαλους παλμούς, πρέπει να μάθει τους περίπλοκους κανόνες που χρησιμοποιεί ο καρδιολόγος για αυτή την ανίχνευση. Για διευκόλυνση αυτής της διαδικασίας εξάγουμε τα πιο σημαντικά χαρακτηριστικά από κάθε παλμό για να τα εισάγουμε μετά στο μοντέλο για την εκπαίδευσή του. Παραδείγματα τέτοιων χαρακτηριστικών είναι η μέγιστη και ελάχιστη τάση κατά τη διάρκεια ενός παλμού, η καθυστέρηση μεταξύ δύο κορυφών παλμού, ο μέσος όρος και μέση τετραγωνική τάση του παλμού κτλ [38].

Κεφάλαιο 3

Διαδικασία Μάθησης Κατηγοριοποιητή

Ένας άνθρωπος νιώθει συναίσθημα και αντιδρά διαισθητικά σε διάφορα ερεθίσματα που δέχεται, όπως ο σωματικός πόνος. Γι 'αυτό τον λόγο είναι δύσκολο για τον άνθρωπο να δώσει οριοθετημένες οδηγίες στον υπολογιστή για το πώς να αντιδρά στις διάφορες καταστάσεις. Μια μηχανή, μπορεί να κατανοήσει κάτι, μόνο αν της δοθούν εντολές ελέγχου που πρέπει να ακολουθήσει για να το πετύχει [24]. Για αυτό τον σκοπό θα αναλύσουμε φυσιολογικά σήματα του εγκεφάλου (EEG, GSR, ECG), ώστε να εξάγουμε κάποια γνώση [49]. Για έναν άνθρωπο είναι δύσκολο αναλύσει αυτά τα δεδομένα με το χέρι, έτσι για αυτό τον σκοπό υπάρχουν διάφορα είδη αλγορίθμων.

Για την ανίχνευση του συναισθήματος ή κάποιας συγκεκριμένης πάθησης, ακολουθείται μια προκαθορισμένη διαδικασία και ανάλογα με το τι θέλουμε να ανιχνεύσουμε, επιλέγουμε το

κατάλληλο σήμα για ανάλυση. Τα βήματα είναι τα εξής, αφού καταγραφούν τα σήματα από τον εγκέφαλο, επεξεργάζονται και αφαιρούνται από αυτά διάφορες ανεπιθύμητες παρεμβολές, μετά εξάγονται τα σημαντικότερα χαρακτηριστικά από αυτά τα σήματα, τα οποία και χρησιμοποιούνται για την κατηγοριοποίηση των δεδομένων, έτσι ώστε να είναι πιο εύκολη η μάθησή τους από τον κατηγοριοποιητή [10]. Το τελευταίο βήμα, δηλ. η μάθηση του μηχανήματος, είναι και ο σκοπός της μηχανικής μάθησης. Ακολουθεί πιο λεπτομερώς και αναλυτικά η διαδικασία αυτή.

Στην παρούσα εργασία, ακολουθήθηκε αυτή διαδικασία για την εκπαίδευση των αλγορίθμων μάθησης που εφαρμόστηκαν. Το πρώτο βήμα που είναι η καταγραφή των σημάτων, πραγματοποιήθηκε από το Τμήμα Ψυχολογίας. Στο δεύτερο βήμα για την επεξεργασία των σημάτων, χρειάστηκε να αφαιρέσουμε κάποια ανεπιθύμητα σήματα. Στο τρίτο βήμα εξάγαμε κάποια χαρακτηριστικά, τα οποία ήταν είτε στατιστικά στοιχεία, είτε ένα ιστόγραμμα με τις τιμές των μετρήσεων, είτε απλά κόψαμε μετρήσεις από τα δείγματα για να έχουν όλα το ίδιο μέγεθος. Τέλος χρησιμοποιήσαμε κάποιους αλγόριθμους μάθησης για να διαχωρίσουμε τα δεδομένα σε δύο κλάσεις, ανάλογα με το πρόβλημα και αξιολογήσαμε την απόδοση του κάθε αλγορίθμου με τη χρήση κάποιων μετρικών.

3.1 Καταγραφή Σημάτων

Η καταγραφή κάποιου σήματος μπορεί να γίνει από επεμβατικές ή από μη-επεμβατικές μεθόδους ή και με τους δύο τρόπους, ανάλογα με το είδος του φυσιολογικού σήματος. Πιο κάτω αναφέρεται ο τρόπος καταγραφής των τριών σημάτων, EEG, GSR, ECG, ξεχωριστά.

3.1.1 Καταγραφή EEG σημάτων

Η καταγραφή των EEG σημάτων μπορεί να γίνει τόσο από επεμβατική όσο και από μη-επεμβατική μέθοδο. Η επεμβατική μέθοδος απαιτεί άμεση εμφύτευση ηλεκτροδίων στον εγκέφαλο με νευροχειρουργική επέμβαση. Ενώ στις μη επεμβατικές μεθόδους γίνεται χρήση ηλεκτροεγκεφαλογραφήματος, όπου τα ηλεκτρόνια τοποθετούνται στην επιφάνεια του τριχωτού της κεφαλής. Η μη επεμβατική μέθοδος έχει κάποια μειονεκτήματα. Τα σήματα έχουν κακή ποιότητα λόγω του ότι το κρανίο υγραίνει τα σήματα και οι νευρώνες θολώνουν τα ηλεκτρομαγνητικά σήματα [15]. Επιπλέον πολλά χαρακτηριστικά του EEG μπορεί να καλυφθούν στο τριχωτό της κεφαλής, λόγω του ότι το EEG εκεί φθείρεται από πιθανά τεχνουργήματα (παρεμβολές στις οποίες θα αναφερθούμε στο δεύτερο στάδιο της διαδικασίας) [20]. Παρόλα αυτά η μη επεμβατική μέθοδος χρησιμοποιείται περισσότερο στις μέρες μας [15].

3.1.2 Καταγραφή GSR σημάτων

Η καταγραφή των GSR σημάτων μπορεί να γίνει από μη επεμβατικές μεθόδους. Για να μετρήσουμε τις αλλαγές στην αγωγιμότητα του δέρματος πρέπει να χρησιμοποιήσουμε αισθητήρα(wearable συσκευή), ο οποίος μετατρέπει την αναλογική ανάγνωση σε ψηφιακές ακατέργαστες τιμές, οι οποίες θα σταλούν στον υπολογιστή. Αυτός ο τρόπος μέτρησης καθιστά το σήμα ευάλωτο μπροστά σε διάφορα τεχνουργήματα [33], λόγω της φύσης του τρόπου λειτουργίας του αισθητήρα Ένας τρόπος πρόληψης για αποφυγή των τεχνουργημάτων είναι η καλή στήριξη των ηλεκτροδίων του GSR αισθητήρα. Για περισσότερη σταθερότητα τα ηλεκτρόδια μπορούν να τοποθετηθούν στον δείκτη και το μεσαίο δάκτυλο. Ένα πλεονέκτημα αυτού του τρόπου είναι ότι τα ηλεκτρόδια δεν εμποδίζουν την κίνηση του χεριού του ασθενή [34]. Και άλλοι τρόποι για να πετύχουμε την αποφυγή τεχνουργημάτων, είναι η τοποθέτηση κλιματισμού στο δωμάτιο πειράματος, για σταθερή θερμοκρασία δωματίου και λήψη μέτρων για αποφυγή οποιουδήποτε θορύβου.

Συνήθως οι αισθητήρες GSR έχουν καλώδια, άβολα ηλεκτρόδια και αυτοκόλλητα gel, τα οποία δυσκολεύουν την κίνηση των ατόμων που τα φορούν. Έτσι ανακαλύφθηκαν και ασύρματοι αισθητήρες, όπως είναι το βραχιόλι iCalm, το οποίο είναι ασύρματο, αδιάβροχο και ελαστικό για να μπορεί να τοποθετηθεί σε κάθε σημείο του σώματος. Υπάρχουν διάφοροι λόγοι που κάποιος θέλει να γνωρίζει για τις μεταβολές του GSR του κατά την διάρκεια της μέρας, είτε σωματικοί λόγοι υγείας είτε ψυχολογικοί. Το βραχιόλι αυτό έχει χρησιμοποιηθεί κυρίως σε αυτιστικά παιδιά, τα οποία δυσκολεύονται να κατανοήσουν και να ρυθμίσουν τα συναισθήματά τους [29].

3.1.3 Καταγραφή ECG σημάτων

Η καταγραφή της ηλεκτρικής δραστηριότητα της καρδιάς μπορεί να γίνει από μη επεμβατικές μεθόδους. Η καταγραφή αυτή γίνεται με την χρήση ηλεκτροδίων που τοποθετούνται επιφανειακά σε διάφορα μέρη του ανθρώπου, κυρίως στα χέρια, στο στήθος ή στα πόδια [37]. Τα δεδομένα που εξάγονται από αυτή την καταγραφή είναι σε μορφή ηλεκτρικού κύματος με κορυφές και κοιλάδες [37].

Σε πολλές έρευνες αυτό το στάδιο της καταγραφής παραλείπεται και χρησιμοποιούνται τα διαθέσιμα δεδομένα από την βάση δεδομένων της MIT (MIT BIH Arrhythmia Database) τα οποία περιέχουν αποσπάσματα από διάφορες καταγραφές σημάτων από ασθενείς [38].

3.2 Επεξεργασία Σημάτων

Όπως αναφέραμε και πιο πάνω αργότερα γίνεται η επεξεργασία των σημάτων. Υπάρχουν πολλά εργαλεία για αυτό τον σκοπό, για παράδειγμα EDF Browser, Matlab, Waikato κτλ. Σε αυτό το στάδιο γίνεται επίσης, η αφαίρεση των ανεπιθύμητων σημάτων, δηλαδή των τεχνουργημάτων (artifacts). Υπάρχουν δύο είδη τεχνουργημάτων, τα φυσιολογικά τεχνουργήματα και τα μη-φυσιολογικά.

Τα φυσιολογικά τεχνουργήματα είναι αυτά που δημιουργούνται λόγω διάφορων σωματικών δραστηριοτήτων, τα οποία διαχωρίζονται ως εξής : τα EOG (electro-oculogram) τεχνουργήματα είναι αυτά που προέρχονται από την κίνηση των ματιών και το ανοιγόκλεισμα. Το ECG (electro-cardiogarphy- ηλεκτροκαρδιογράφημα) τεχνούργημα προέρχεται από τον καρδιακό παλμό. Το EMG (electro-myography) προέρχεται από την κίνηση του κεφαλιού και των μυών. Ενώ τα μη φυσιολογικά τεχνουργήματα είναι αυτά που προέρχονται από εξωτερικούς παράγοντες, δηλ. όχι από το σώμα του ανθρώπου, όπως η βρωμιά, η ποιότητα του ηλεκτροδίου, η μεταβολή στην αντίσταση του ηλεκτροδίου, η προσαρμογή της συσκευής, ο ηλεκτρονικός θόρυβος, ακόμα και ένας εξωτερικός θόρυβος όπως η μετακίνηση μιας καρέκλας κτλ [10].

Η αφαίρεση των τεχνουργημάτων είναι σημαντική για την αποτελεσματικότητα του επόμενου σταδίου, που είναι η εξαγωγή χαρακτηριστικών. Αν αυτά τα τεχνουργήματα δεν ανιχνευτούν και δεν αφαιρεθούν, θα αποφέρουν λανθασμένα συμπεράσματα κατά την ανάλυση του σήματος. Για παράδειγμα στην περίπτωση επεξεργασίας του GSR, μπορεί να παραποιήσουν την αγωγιμότητα του δέρματος και να ανιχνευτεί λανθασμένα αυξημένο άγχος.
Η ανίχνευση των τμημάτων του σήματος που περιέχουν αυτά τα τεχνουργήματα μπορεί να γίνει και με το χέρι. Σε περίπτωση όμως που ο αριθμός των δεδομένων που θα επεξεργαστούμε είναι μεγάλος, τότε είναι απαραίτητη η χρήση αυτοματοποιημένων αλγορίθμων για την αφαίρεση των τεχνουργημάτων [33]. Φυσικά οι τεχνικές αφαίρεσης τεχνουργημάτων δεν επηρεάζουν αρνητικά την ποιότητα του σήματος, ως εκ τούτου οι νευρωνικές δραστηριότητες μένουν ανεπηρέαστες [21].

3.3 Εξαγωγή Σημαντικών Χαρακτηριστικών από τα Σήματα

Το επόμενο στάδιο της διαδικασίας είναι η εξαγωγή σημαντικών χαρακτηριστικών από τα σήματα. Τα χαρακτηριστικά αυτά επιλέγονται είτε απ΄ευθείας είτε αλγοριθμικά από ένα σύνολο υποψηφίων χαρακτηριστικών που έχουν καθοριστεί από τον άνθρωπο. Τα χαρακτηριστικά αυτά θα περάσουν ως είσοδο στον κατηγοριοποιητή. Γι' αυτό το λόγο είναι σημαντικό τα χαρακτηριστικά που επιλέγονται να διαχωρίζουν καλά τα δεδομένα και επίσης να σχετίζονται με το στόχο του κατηγοριοποιητή. Τα χαρακτηριστικά εκτός από το ότι προσφέρουν

γνώση στον κατηγοριοποιητή, μπορούν επίσης να μειώσουν το υπολογιστικό του φορτίο, κάνοντας τον κατηγοριοποιητή πιο γρήγορο. Συνεπώς η απόφαση επιλογής είναι σημαντική, γι 'αυτό είναι απαραίτητο να γίνεται κάποια αξιολόγηση απόδοσης κατηγοριοποίησης για κάθε υποψήφιο χαρακτηριστικό. Έτσι τελικά χρησιμοποιούνται τα χαρακτηριστικά που προέβλεψαν πιο αποτελεσματικά την κατηγορία συγκεκριμένων δειγμάτων. Ας σημειώσουμε ότι πριν από αυτό το στάδιο μπορεί να γίνει και κάποια οπτική ανάλυση, για να προσδιοριστεί κατά πόσο είναι διακριτές οι διαφορές των δεδομένων των σημάτων μεταξύ των υπό εξέταση κλάσεων [22].

Υπάρχουν διάφοροι μέθοδοι μετασχηματισμού, οι οποίοι παίρνουν ως είσοδο τα χρήσιμα χαρακτηριστικά που εξάχθηκαν από το σήμα και τα μετασχηματίζουν ως διανύσματα χαρακτηριστικών. Αυτά τα διανύσματα περνούν ως είσοδο στον κατηγοριοποιητή.

3.4 Κατηγοριοποίηση των Δεδομένων

Ο κατηγοριοποιητής διαχωρίζει τα δεδομένα σε κλάσεις με βάση το διάνυσμα χαρακτηριστικών, το οποίο πήρε από το προηγούμενο στάδιο. Φυσικά τα χαρακτηριστικά τα οποία δεν βοηθούν στην ακρίβεια κατηγοριοποίησης διαγράφονται [47]. Ένας κατηγοριοποιητής επηρεάζεται από πολλαπλούς παράγοντες, όπως το περιβάλλον πειράματος, τεχνικές προεπεξεργασίας των δεδομένων κτλ. Έτσι δεν μπορεί να υπάρξει ένας κατηγοριοποιητής που είναι καλός για όλες τις εφαρμογές, αλλά κάθε φορά επιλέγεται ο πιο κατάλληλος κατηγοριοποιητής για συγκεκριμένη εφαρμογή. Για παράδειγμα για την αναγνώριση των συναισθημάτων έχει αποδειχθεί ότι ο καλύτερος κατηγοριοποιητής είναι ο Support Vector Machine [10]. Επίσης σε BCI (Brain Computer Interface) έρευνα τα νευρωνικά δίκτυα είναι ο πιο χρησιμοποιημένος κατηγοριοποιητής [13].

α) Μερικοί ταξινομητές που χρησιμοποιούνται στην κατηγοροποίηση EEG είναι οι εξής, Support Vector Machine, τεχνητό νευρωνικό δίκτυο (Artificial Neural Network), KNNK-nearest neighbor. [10], γραμμικοί ταξινομητές, μη γραμμικοί ταξινομητές, μη γραμμικοί Bayesian ταξινομητές, ακόμα και συνδυασμός ταξινομητών (Lotte et al., 2007) [13], Δίχτυα βαθιάς πίστης (DBNs-Deep belief nets), δέντρα απόφασης (DTs) ή αλλιώς δέντρα κατηγοριοποίησης [22].

β) Μερικοί ταξινομητές που χρησιμοποιούνται στην κατηγοροποίηση GSR είναι οι εξής, KMeans κατάταξη σύμφωνα με διανυσματική ποσοστοποίησης (quantization), δέντρο απόφασης κατάταξης, Gaussian Mixture Model (GMM) κατηγοριοποίηση [15], (έχει χρησιμοποιηθεί πολύ για αναγνώριση ομιλητή), και η Support Vector Machine (LibSVM εργαλείο [6], RBF kernel με ενδοπιστοποίηση (cross-validation-επικύρωση).

γ) Μερικοί ταξινομητές που χρησιμοποιούνται στην κατηγοροποίηση ECG είναι οι εξής, Gaussian mixture μοντέλο (GMM), error back propagation neural network (EBPNN), και support vector machine (SVM). Ο πιο επιτυχής κατηγοριοποιητής για το σήμα EEG είναι ο SVM, μετά ακολουθεί ο EBPNN, KNNK-nearest neighbor και τέλος ο GMM [51].

Ακόμα υπάρχει δυνατότητα να συνδυάσουμε δύο διαφορετικά μοντέλα μαζί (για παράδειγμα ομιλίας and GSR). Αυτή η μέθοδος λέγεται λογιστική παλινδρόμηση. Χρειάζεται τα δεδομένα των δύο μοντέλων την ίδια στιγμή και επίσης τα δεδομένα αυτά να είναι ευθυγραμμισμένα μεταξύ τους [30].

Παρατηρούμε ότι στους παραπάνω ταξινομητές των τριών σημάτων, υπάρχουν κάποιοι που μπορούν να χρησιμοποιηθούν αποτελεσματικά και στους τρεις τύπους σημάτων. Οι πιο διαδεδομένοι από αυτούς τους ταξινομητές αναφέρονται πιο κάτω.

3.5 Εκπαίδευση Κατηγοριοποιητή

Το τελευταίο βήμα της διαδικασίας είναι η εκπαίδευση του κατηγοριοποιητή. Ο κατηγοριοποιητής εκπαιδεύεται με την βοήθεια των χαρακτηριστικών (το οποίο αναφέραμε στο προηγούμενο στάδιο) και αξιολογείται κατά πόσο είναι επιτυχής στις διαγνώσεις του. Η αξιολόγηση γίνεται με βάση κάποιων μετρικών, όπως η ακρίβεια, η ευαισθησία και η προσδιοριστικότητα, στις οποίες θα αναφερθούμε πιο κάτω.

Σχήμα 3.1 : Το σχήμα απεικονίζει τη Διαδικασία Μάθησης του Κατηγοριοποιητή

Κεφάλαιο 4

Κατηγορίες Μάθησης

Ο άνθρωπος μπορεί να μάθει είτε μέσω κάποιου εκπαιδευτή, είτε χωρίς εκπαιδευτή. Αντίστοιχα και η Μηχανική Μάθηση χωρίζεται σε δύο κατηγορίες, την επιβλεπόμενη μάθηση και την μη επιβλεπόμενη. Η χρήση μιας από τις δύο αυτές μεθόδους εξαρτάται από το αν τα δεδομένα περιέχουν και την κλάση στην οποία ανήκουν ή αν η κλάση είναι άγνωστη. Στην πρώτη περίπτωση χρησιμοποιούμε επιβλεπόμενη μάθηση, ενώ στην δεύτερη περίπτωση μη επιβλεπόμενη μάθηση.

Στην παρούσα εργασία θα εφαρμόσουμε σε πραγματικά δεδομένα κάποιους από τους αλγορίθμους μάθησης που θα περιγράψουμε πιο κάτω, οι οποίοι είναι οι Μηχανές Διανυσμάτων Υποστήριξης, ο αλγόριθμος Κατηγοριοποίησης k Κοντινότερων Γειτόνων και ο ομαδοποιητής k-means.

4.1 Επιβλεπόμενη Μάθηση

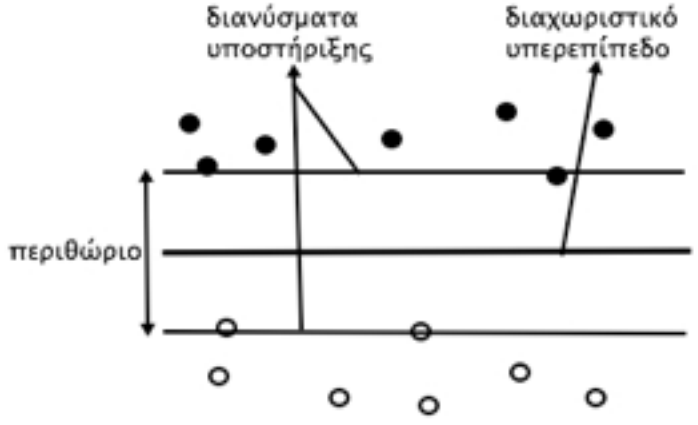
Στην επιβλεπόμενη μάθηση ή μάθηση με εκπαιδευτή, ο εκπαιδευτής τροφοδοτείται με δεδομένα εκπαίδευσης. Τα δεδομένα εκπαίδευσης είναι ζεύγη εισόδου-εξόδου, όπου είσοδος είναι ένα δεδομένο και έξοδος η κλάση στην οποία ανήκει (label data). Έτσι ο εκπαιδευτής αποκτά αρκετή γνώση για την γενίκευση και για μια είσοδο με άγνωστη έξοδο. Συνεπώς ο αλγόριθμος εκμάθησης προσπαθεί να ελαχιστοποιήσει το σφάλμα σε σχέση με τα δεδομένα εισόδου. Έτσι ενόσω τα δεδομένα εκπαίδευσης αυξάνονται, ο εκπαιδευτής βελτιώνεται.
Ένα από τα μειονεκτήματα της επιβλεπόμενης μάθησης είναι ότι είναι δαπανηρή όταν υπάρχουν πολλά δεδομένα εκπαίδευσης, τα οποία πρέπει να χαρακτηριστούν. Ακόμα το είδος της κλάσης των δεδομένων μπορεί να μην είναι γνωστό.

Τα πλεονεκτήματα τέτοιων αλγορίθμων εποπτευόμενης μάθησης είναι ότι, είναι πιο κατάλληλοι για εκπαίδευση και για μάθηση πολύπλοκων συστημάτων, και πιο εύκολοι για κατασκευή.

Πιο κάτω θα περιγραφούν οι εξής αλγόριθμοι επιβλεπόμενης μάθησης, οι Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machine-SVM), ο αλγόριθμος Κατηγοριοποίησης k Κοντινότερων Γειτόνων (k-Nearest Neighbor- KNN), Συνελικτικά Νευρωνικά δίκτυα (Convolutional Neural Networks-CNNs) [58].

4.1.1 Μηχανές Διανυσμάτων Υποστήριξης

Είναι ένας δημοφιλής αλγόριθμος επιβλεπόμενης μάθησης (δηλαδή έχει την ικανότητα να μαθαίνει), που περιορίζεται μόνο σε προβλήματα κατηγοριοποίησης σε δύο ομάδες. Το κριτήριο κατηγοριοποίησης καθορίζεται εξαρχής. Χρησιμοποιεί την αρχή Ελαχιστοποίησης του Δομικού μέσου. Δηλαδή ο SVM (Support Vector Machine-SVM) κατηγοριοποιητής διαχωρίζει τα δείγματα εκπαίδευσης (training data set) σε δύο σύνολα διανυσμάτων προσπαθώντας η απόσταση μεταξύ των δύο να είναι όσο το δυνατόν μεγαλύτερη, ώστε να κατασκευάσει ένα διαχωριστικό υπερεπίπεδο μεταξύ των δύο συνόλων. Με αυτό τον τρόπο ελαχιστοποιείται το σφάλμα κατηγοριοποίησης [40]. Το περιθώριο είναι το σημείο μεταξύ των δύο κατηγοριών, που δεν περιέχει σημεία δεδομένων. Και τα διανύσματα υποστήριξης (support vectors) είναι τα σημεία δεδομένων που βρίσκονται πιο κοντά στο υπερεπίπεδο.



Σχήμα 4.1 : Το σχήμα απεικονίζει το διαχωριστικό υπερεπίπεδο μεταξύ δύο συνόλων διανυσμάτων. Με βάση αυτό θα κατασκευάσουμε μια συνάρτηση απόφασης η οποία θα διαχωρίζει με τον βέλτιστο τρόπο τα δεδομένα [53].

Η συνάρτηση απόφασης είναι η πιο κάτω.

$$f\left(x\right)=sgn\left[\sum_{i=1}^{l} y_i a_i \left(x.x_i\right)+b\right]$$

όπου α_i είναι οι πολλαπλασιαστές Lagrange.

Ένας σημαντικός παράγοντας που επηρεάζει την ακρίβεια του κατηγοριοποιητή είναι η επιλογή της μεθόδου πυρήνα. Υπάρχουν διάφορες συναρτήσεις πυρήνα με διαφορετική αποδοτικότητα, όπως η γραμμική, πολυωνυμική και ακτινική συνάρτηση [40].

4.1.1.1 Γραμμικές Μηχανές Διανυσμάτων Υποστήριξης

Οι Γραμμικές μηχανές διανυσμάτων υποστήριξης (Linear Support Vector Machines) μαθαίνουν με βάση διαχωρίσιμα δεδομένα. Αυτά τα δεδομένα εκπαίδευσης έχουν την εξής μορφή

$$\{x_i, y_i\}, i = 1, \cdots, l, y_i \in \{-1, 1\}, x_i \in R^d,$$

όπου το 1, -1 υποδεικνύουν τις δύο κατηγορίες που διαχωρίζονται από το υπερεπίπεδο και το x_i διάνυσμα μπορεί να ανήκει σε μια από τις δύο κατηγορίες. Τα σημεία x που βρίσκονται στο υπερεπίπεδο πρέπει να ικανοποιούν τον περιορισμό $w \cdot x + b = 0$, όπου το w είναι κάθετο διάνυσμα στο υπερεπίπεδο, $| b | /|| w||$ είναι κάθετη απόσταση του υπερεπιπέδου με την origin και $||w||$, είναι το Ευκλείδεια νόρμα(Euclidean norm) του w. Ο στόχος του SVM, στην γραμμική διαχωρίσιμη περίπτωση, είναι η μεγιστοποίηση του περιθωρίου. Το περιθώριο βρίσκεται μεταξύ δύο υπερεπιπέδων, H1 και H2,που το διαχωρίζουν. Τα δύο αυτά υπερεπίπεδα ικανοποιούν αυτές τις δύο εξισώσεις αντίστοιχα, H1: $xi \cdot w + b = 1$ και H2: $xi \cdot w + b = -1$. Το περιθώριο μεταξύ των δύο υπερεπιπέδων ισούται με $2 /||w||$ και δεν περιέχει σημεία δεδομένων. Αφού το $||w||$ είναι αντιστρόφως ανάλογο με το περιθώριο, όσο πιο μικρό είναι, τόσο πιο μεγάλο θα είναι το περιθώριο. Από τις πιο πάνω παρατηρήσεις προκύπτουν οι εξής περιορισμοί για κάθε διάνυσμα x_i :

$x_i \cdot w + b \geq 1$ για $y_i = +1$ (1)

$x_i \cdot w + b \leq -1$ για $y_i = -1$ (2)

Οι δύο πιο πάνω ανισότητες μπορούν να γραφτούν ως μια :

$y_i (x_i \cdot w + b) - 1 \geq 0 \; \forall i$ (3)

Τα στοιχεία που ικανοποιούν τις πιο πάνω εξισώσεις,δηλαδή βρίσκονται πάνω σε ένα από τα δύο υπερεπίπεδα, ονομάζονται διανύσματα υποστήριξης (support vectors) και η αφαίρεσή τους θα οδηγήσει σε οδηγήσει σε άλλη λύση δύο διαστάσεων.

Για ευκολία των υπολογισμών θα χρησιμοποιήσουμε την διατύπωση του προβλήματος με πολλαπλασιαστές Lagrange. Έτσι τα δεδομένα εκπαίδευσης είναι σε μορφή γινομένων μεταξύ διανυσμάτων. Βάσει αυτών δημιουργείται η πιο κάτω συνάρτηση Lagrangian :

$$L_p \equiv \frac{1}{2}\|w\|^2 - \sum_{i=1}^{l} a_i y_i \left(x_i \cdot w + b\right) + \sum_{i=1}^{l} a_i$$

όπου για κάθε για κάθε περιορισμό ανισότητας (3), υπάρχει μπροστά ένας θετικός πολλαπλασιαστής Lagrange α_i, $i=1,...,l$. Για τον σχηματισμό της πιο πάνω Lagrange συνάρτησης χρησιμοποιήθηκε ο κανόνας ότι, οι εξισώσεις περιορισμών είναι σε μορφή ανισότητας, και έτσι πολλαπλασιάστηκαν με τους πολλαπλασιαστές Lagrange και αφαιρέθηκαν από μια αντικειμενική συνάρτηση.

Το πρωτεύον πρόβλημα (primal problem) είναι η ελαχιστοποίηση της συνάρτησης LP ως προς τα w και b, εξαφανίζοντας όλα τα διανύσματα α_i, όπου $\alpha_i \geq 0$. Αυτό είναι ένα κυρτό τετραγωνικό πρόβλημα προγραμματισμού (convex quadratic programming problem). Για την βελτιστοποίηση των περιορισμών σε αυτό το πρόβλημα χρησιμοποιούνται οι εξής συνθήκες Karush-Kuhn-Tucker (KKT) :

$$\frac{\partial}{\partial w_v} L_p = w_v - \sum_i a_i y_i x_{iv} = 0 \qquad v = 1,...,d$$

$$\frac{\partial}{\partial b} L_p = -\sum_i a_i y_i = 0$$

$$y_i \left(x_i \cdot w + b\right) - 1 \geq 0 \qquad i = 1,...l$$

$$a_i \geq 0 \qquad \forall i$$

$$a_i \left(y_i \left(w \cdot x_i + b\right) - 1\right) = 0 \qquad \forall i$$

Ακόμα ένα κυρτό τετραγωνικό πρόβλημα προγραμματισμού είναι, το δυϊκό πρόβλημα (Wolfe dual problem), όπου γίνεται μεγιστοποίηση της LP ως προς τους πολλαπλασιαστές α_i, όπου $\alpha_i \geq 0$. Σε αντίθεση με το το πρωτεύον πρόβλημα τα w και b εξαφανίζονται και έτσι προκύπτουν οι εξής προϋποθέσεις :

$$w = \sum_i a_i y_i x_i$$

$$\sum_i a_i y_i = 0$$

Οι δύο πιο πάνω εξισώσεις μπορούν να γραφτούν ως μια :

$$L_D = \sum_i a_i - \frac{1}{2} \sum_{i,j} a_i a_j y_i y_j x_i \cdot x_j$$

Συμπερασματικά οι δύο διαμορφώσεις, LP και LD, διαφέρουν ως προς τους περιορισμούς, όμως είναι ισοδύναμες [53].

4.1.1.2 Μη Γραμμικές Μηχανές Διανυσμάτων Υποστήριξης

Οι Μη Γραμμικές Μηχανές Διανυσμάτων (Nonlinear Support Vector Machines) χρησιμοποιούνται στην περίπτωση που τα δεδομένα εκπαίδευσης δεν μπορούν να διαχωριστούν αποτελεσματικά με μια γραμμική συνάρτηση υπερεπιπέδου. Δηλαδή μπορούμε να πούμε ότι οι Μη Γραμμικές Μηχανές Διανυσμάτων είναι μια γενίκευση των Γραμμικών Μηχανών Διανυσμάτων. Για την μεγιστοποίηση του περιθωρίου στους μη γραμμικούς ταξινομητές (Boser, Guyon και Vapnik, 1992) χρησιμοποιούνται συναρτήσεις πυρήνα (Aizerman, 1964).

Όπως αναφέραμε και πιο πάνω ο αλγόριθμος εκπαίδευσης χρησιμοποιεί πολλαπλασιαστές Lagrange, δηλαδή τα δεδομένα εκπαίδευσης είναι σε μορφή γινομένων μεταξύ διανυσμάτων, $x_i \cdot x_j$. Αυτά τα δεδομένα εκπαίδευσης πρέπει να χαρτογραφηθούν σε κάποιο άλλο Ευκλείδειο χώρο H: $\Phi : R^d \longmapsto H$.

Έτσι τα δεδομένα εκπαίδευσης μπορούν να διαχωριστούν χρησιμοποιώντας γινόμενα μη γραμμικών διανυσματικών συναρτήσεων

$\Phi(x_i) \cdot \Phi(x_j)$. Ο αλγόριθμος εκπαίδευσης αντικαθιστά αυτά τα γινόμενα με μια συνάρτηση πυρήνα K έτσι ώστε $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$. Αυτή η αντικατάσταση έχει ως πλεονέκτημα ότι δεν θα χρειάζεται η κατανόηση του τι είναι το Φ και η ρητή χρήση του, το οποίο ευκολύνει τους υπολογισμούς, ιδικά στην περίπτωση που το H είναι άπειρων διαστάσεων. Με αυτή την τεχνική πυρήνα, οδηγούμαστε στην περίπτωση των γραμμικά διαχωριζόμενων δεδομένων με την αντικατάσταση κάθε x_i με $\Phi(x_i)$. Επίσης και τα γινόμενα με w μπορούν να υπολογιστούν με την συνάρτηση πυρήνα και να αποφύγουμε και εδώ τον υπολογισμό του $\Phi(x)$. Άρα η συνάρτηση απόφασης του SVM υπολογίζεται από το *sign* της πιο κάτω εξίσωσης

$$f(x) = \sum_{i=1}^{N_s} a_i y_i \Phi(s_i) \cdot \Phi(x) + b = \sum_{i=1}^{N_s} a_i y_i K(s_i, x) + b$$

όπου s_i είναι το διάνυσμα υποστήριξης.

Συνοψίζοντας, όταν τα δεδομένα δεν είναι γραμμικά διαχωρίσιμα, τότε τα τοποθετούμε σε ένα άλλο χώρο μεγαλύτερης διάστασης έτσι ώστε να πετύχουμε ένα μη γραμμικό διαχωρισμό και να ελαχιστοποιηθούν τα λάθη στον διαχωρισμό των δεδομένων. Αυτή μετάβαση σε ένα άλλο χώρο γίνεται μέσω των συναρτήσεων πυρήνα $K(x_i, x_j)$. Αυτή η τεχνική είναι χρήσιμη ιδιαίτερα όταν το Φ έχει πολύ μεγαλύτερη διάσταση από το πεδίο ορισμού του (domain). Ένας χώρος χαμηλής διάστασης συμβολίζεται με το γράμμα L, ενώ ένας χώρος ψηλής διάστασης συμβολίζεται με H.

Υπάρχουν πολλές διαφορετικές συναρτήσεις πυρήνα. Σε κάθε πρόβλημα η αποτελεσματικότερη συνάρτηση είναι διαφορετική. Γι' αυτό γίνεται κάποια αξιολόγηση μεταξύ διάφορων πιθανών-εναλλακτικών συναρτήσεων και επιλέγεται αυτή που διαχωρίζει τα δεδομένα καλύτερα. Επίσης για κάθε πυρήνα διαλέγουμε την κατάλληλη Φ χαρτογράφηση και τον χώρο H. Κάθε πυρήνας μπορεί να έχει πολλές εναλλακτικές για Φ και H.

Παράδειγμα συνάρτησης πυρήνα

Το πρόβλημα έχει τα εξής χαρακτηριστικά :
- Τα δεδομένα είναι διανύσματα στο R^2.
- Η συνάρτηση πυρήνα είναι η $K(x_i, x_j) = (x_i \cdot x_j)^2$.

Υπάρχουν διάφοροι χώροι διάστασης H και χαρτογράφησης Φ για αυτόν τον πυρήνα, μερικά παραδείγματα φαίνονται πιο κάτω [53]:

Περίπτωση 1

-$H = R3$
-$\Phi : R^2 \longmapsto H$

$$\Phi(x) = \begin{pmatrix} x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \end{pmatrix}$$

Περίπτωση 2

-$H = R3$
-$\Phi : R^2 \longmapsto H$

$$\Phi(x) = \frac{1}{\sqrt{2}} \begin{pmatrix} (x_1^2 - x_2^2) \\ 2x_1x_2 \\ (x_1^2 + x_2^2) \end{pmatrix}$$

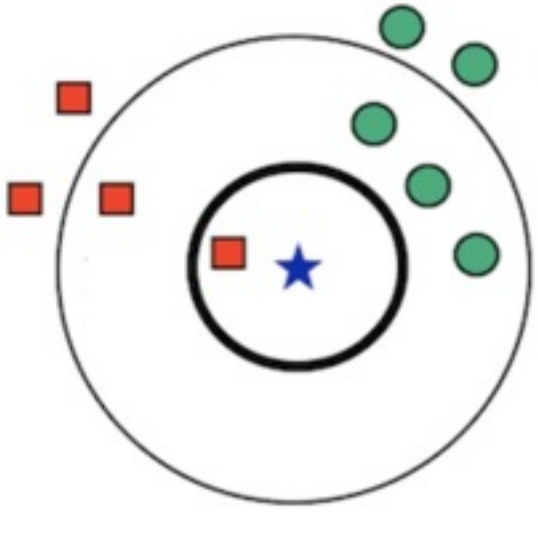
Περίπτωση 3

-$H = R3$
-$\Phi : R^2 \longmapsto H$

$$\Phi(x) = \begin{pmatrix} x_1^2 \\ x_1x_2 \\ x_1x_2 \\ x_2^2 \end{pmatrix}$$

4.1.2 Αλγόριθμος κατηγοριοποίησης k κοντινότερων γειτόνων

Είναι ένας μη παραμετρικός αλγόριθμος εκμάθησης, ο οποίος κατηγοριοποιεί ένα άγνωστο στοιχείο στην κλάση που ανήκουν τα k κοντινότερα του πρότυπα από τα δείγματα εκπαίδευσης (training data set). Στη περίπτωση που τα κοντινότερα πρότυπα δεν ανήκουν όλα στην ίδια κλάση, τότε υπερισχύει η πλειοψηφία μεταξύ των k πιο κοντινών προτύπων. Όσο πιο μεγάλο είναι το σύνολο εκπαίδευσης, τόσο πιο μεγάλος είναι και ο αριθμός k, έτσι ώστε να είμαστε σίγουροι ότι η κατηγοριοποίηση γίνεται σωστά. Υπάρχουν πολλές παραλλαγές αυτού του αλγορίθμου. Εκτός από τα μέτρα απόστασης k, μπορούν να χρησιμοποιηθούν και διαφορετικά βάρη σε κάθε γείτονα [40]. Το ποσοστό σφάλματος που έχει ο αλγόριθμος knn δεν είναι μεγαλύτερο από το διπλάσιο του ποσοστού σφάλματος Bayes, το οποίο είναι το ελάχιστο σφάλμα σε σχέση με την κατανομή των δεδομένων [54].



Σχήμα 4.2 : Πιο πάνω φαίνεται ένα **παράδειγμα knn κατηγοριοποίησης**. Το αστεράκι συμβολίζει το σημείο δοκιμής και τα υπόλοιπα σχήματα συμβολίζουν τα σημεία του συνόλου εκπαίδευσης, όπου τα ίδια σχήματα είναι σημεία της ίδιας κλάσης. Αν εφαρμοστεί κατηγοριοποίηση κοντινότερου γείτονα (k=1), τότε το σημείο δοκιμής θα τοποθετηθεί στην κλάση των τετραγώνων καθώς το κοντινότερο σημείο είναι τετράγωνο (μικρός σκούρος κύκλος). Ενώ αν εφαρμοστεί κατηγοριοποίηση πέντε κοντινότερων γειτόνων (k=5), τότε το

σημείο δοκιμής θα τοποθετηθεί στην κλάση κύκλων, αφού η πλειοψηφία των πέντε κοντινότερων σημείων είναι κύκλοι (δηλαδή οι τρεις γείτονες είναι κύκλοι και οι υπόλοιποι δύο είναι τετράγωνα, στο μεγάλο λεπτό κύκλο) [54].

Για την κατηγογοροποίηση knn υπάρχει αποθηκευμένο ένα διαθέσιμο σύνολο εκπαίδευσης που αποτελείται από N επισημασμένα παραδείγματα
$\{x_i, y_i\}^N$ όπου x_i είναι ένα πρότυπο και y_i η κλάση στην οποία ανήκει. Ο κατηγοριοποιητής knn παίρνει ως είσοδο ένα άγνωστο πρότυπο (ή αλλιώς query vector ή αλλιώς σημείο δοκιμής/test point) και βγάζει σαν έξοδο την ετικέτα κλάσης (class label) του. Όπως αναφέραμε και πιο πάνω, αρχικά βρίσκονται οι k κοντινότεροι γείτονες του σημείου δοκιμής και μετά βάση αυτών βρίσκεται η κλάση του. Για την εύρεση των k κοντινότερων γειτόνων, συνήθως εφαρμόζεται Ευκλείδεια απόσταση ως μετρικό απόστασης (distance metric). Δηλαδή υπολογίζεται η απόσταση του σημείου δοκιμής από κάθε x_i του συνόλου εκπαίδευσης.

Για την απόδοση του κατηγοριοποιητή knn είναι σημαντική η επιλογή του k, καθώς και η επιλογή του μετρικού απόστασης (distance metric). Η καλύτερη επιλογή του k εξαρτάται από τα δεδομένα, γι' αυτό για κάθε διαφορετική εφαρμογή είναι διαφορετικό το βέλτιστο k. Γενικά, οι μεγαλύτερες τιμές του k, κάνουν τον κατηγοριοποιητή πιο ανθεκτικό στην παρουσία θορύβου, αλλά κάνουν λιγότερο διακριτά τα όρια μεταξύ των κλάσεων [55]. Η επιλογή του k μπορεί να γίνει από διάφορες ευρετικές τεχνικές [54]. Στην περίπτωση που τα δεδομένα δεν είναι ομοιόμορφα κατανεμημένα, τότε ένα σταθερό k δυσκολεύει την σωστή κατηγοριοποίηση.
Ακόμα στην περίπτωση που το μοντέλο κατηγοριοποίησης έχει μόνο δύο κλάσεις κατηγοριοποίησης (δυαδικό πρόβλημα), μια καλή επιλογή για το k είναι ένας περιττός αριθμός για να μην υπάρξουν ισοψηφίες [54]. Στο συγκεκριμένο πρόβλημα το βέλτιστο k μπορεί να βρεθεί με την μέθοδο bootstrap.

Παραδείγματα μετρικών απόστασης για χαρακτηριστικά προτύπων με συνεχείς τιμές χρησιμοποιείται είναι, η Ευκλείδεια απόσταση και η απόσταση Manhattan. Ενώ παράδειγμα μετρικού απόστασης για διακριτές τιμές είναι το μετρικό επικάλυψης (overlap metric ή απόσταση Hamming), η οποία έχει χρησιμοποιηθεί για την κατηγοριοποίηση κειμένου. Για μεγαλύτερη απόδοση κατηγοριοποιητή, οι μετρικές απόστασης χρησιμοποιούν εξειδικευμένους αλγορίθμους, όπως Large Margin Nearest Neighbor or Neighbourhood components analysis [54].

Στην περίπτωση που ο αλγόριθμος κατηγοριοποίησης knn εφαρμόζει πλειοψηφία για να βρει την κλάση στην οποία ανήκει ένα σημείο δοκιμής, καλό θα ήταν οι τάξεις να είναι όσο πιο συμμετρικά κατανεμημένες γίνεται. Σε αντίθετη περίπτωση θα είναι πιο συχνή η τοποθέτηση σημείων δοκιμής σε μια συγκεκριμένη κλάση, επειδή δε έχει περισσότερους κοινούς γείτονες με αυτό το πρότυπο, αλλά μεν επειδή έχει πολύ μεγαλύτερο αριθμό προτύπων σε σχέση με άλλες κλάσεις. Μια λύση σε αυτό το πρόβλημα είναι η απονομή βαρών σε κάθε σημείο κάθε κλάσης,

σε σχέση με την απόσταση από το σημείο δοκιμής. Το βάρος ενός σημείου σε μια κλάση ισούται με το αντίστροφο της απόστασης του σημείου αυτού με το σημείο δοκιμής. Άρα η τάξη ενός σημείου πολλαπλασιάζεται με αυτό το βάρος. Μια άλλη λύση στο πιο πάνω πρόβλημα είναι η αφαίρεση στην αναπαράσταση των δεδομένων [54].

Υπάρχουν πολλές εκδόσεις του knn αλγορίθμου κατηγοριοποίησης. Μια από αυτές τις εκδόσεις είναι η εύρεση των k κοντινότερων γειτόνων με χρήση της Ευκλείδειας απόστασης και τα ίσα βάρη στον κάθε γείτονα [40]. Η πιο απλή έκδοση του αλγορίθμου ονομάζεται πλησιέστερος αλγόριθμος γείτονα, όπου η κλάση του υπό σημείου δοκιμής είναι αυτή του πιο κοντινού του σημείου από το σύνολο δεδομένων εκπαίδευσης (δηλαδή k=1). Αυτή η έκδοση είναι εύκολο να υλοποιηθεί, απλά μετρώντας όλες τις αποστάσεις από το σημείο δοκιμής σε όλα τα άλλα αποθηκευμένα σημεία. Όμως το μειονέκτημα αυτής της έκδοσης είναι ότι κάνει όλους τους δυνατούς υπολογισμούς για να βρει τελικά ποια είναι η πιο κοντινή απόσταση από το σημείο δοκιμής, το οποίο έχει πάρα πολύ μεγάλο υπολογιστικό κόστος ιδικά για μεγάλα σύνολα δεδομένων. Για αυτό ο κύριος στόχος των αλγορίθμων πλησιέστερου γείτονα είναι να μειώσουν όσο το δυνατόν τον αριθμό των υπολογισμών αποστάσεων που εκτελούνται. Ακόμα για καλύτερη απόδοση, οι αλγόριθμοι μπορούν να χρησιμοποιήσουν προηγούμενη γνώση (έννοια πληροφόρησης)και να μην επηρεάζονται αρνητικά από την αλλαγή των παραμέτρων εισόδου. Η ιδέα πίσω από ένα knn αλγόριθμο πληροφόρησης είναι κάθε στοιχείο στο σύνολο δεδομένων εκπαίδευσης να επεξεργάζεται για να έχει πληροφόρηση, έτσι ώστε να έχει περισσότερες διακριτές διαφορές με κάποιο άλλο στοιχείο σε κάποια άλλη κλάση και να περισσότερες ομοιότητες με κάποιο στοιχείο δοκιμής. Άρα το μετρικό απόστασης που χρησιμοποιείται σε αυτή την περίπτωση είναι η πληροφόρηση [55].

4.1.2.1 Αλγόριθμος LI-KNN

Ένας αλγόριθμος πληροφόρησης είναι ο LI-KNN (Locally Informative KNN).
Η ομοιότητα αυτού του κατηγοριοποιητή σε σχέση με τον knn κατηγοριοποιητή είναι ότι τα σημεία που έχουν μικρή διαφορά απόστασης τοποθετούνται στην ίδια κλάση και επίσης για κάθε σημείο δοκιμής μετρούν την απόσταση του από τους γείτονες. Ο επιπλέον περιορισμός που έχει ο LI-knn είναι ότι οι γείτονες ενός σημείου δοκιμής πρέπει να έχουν και αυτοί μικρή απόσταση μεταξύ τους και μεγάλη απόσταση από άλλα απομακρυσμένα σημεία. Έτσι δεν αντιμετωπίζονται όλοι οι γείτονες του σημείου δοκιμής με τον ίδιο τρόπο όπως γινόταν με προηγούμενα μετρικά του knn που αναφέραμε.

Πιο κάτω αναφέρονται τα βήματα του αλγορίθμου LI-knn για να ταξινομήσει ένα σημείο δοκιμής. Το πρώτο βήμα που ο κατηγοριοποιητής LI-knn κάνει είναι να βρει τους k πιο κοντινούς του γείτονες του σημείου δοκιμής με την χρήση της Ευκλείδειας απόστασης. Μετά

βρίσκει την πληροφόρηση μόνο για κάθε ένα από αυτά τα k τοπικά σημεία με τη χρήση του μετρικού πληροφόρησης. Τα k πληροφοριακά σημεία ανήκουν στο σύνολο I. Και τέλος το σημείο δοκιμής τοποθετείται στην κλάση στην οποία ανήκει η πλειοψηφία των σημείων I.
Το σημαντικό σε αυτή την κατηγοριοποίηση είναι ότι δεν υπολογίζει την πληροφόρηση όλων των σημείων στο σύνολο εκπαίδευσης, αλλά μόνο των k γειτόνων. Αυτό μειώνει υπερβολικά τους υπολογισμούς που πρέπει να γίνουν, εστιάζοντας την κατηγοριοποίηση σε τοπικά πληροφοριακά σημεία, εξ ου και το όνομα του αλγορίθμου (τοπικός πληροφοριακός αλγόριθμος). Σε αντίθετη περίπτωση αν ο αλγόριθμος υπολόγιζε την πληροφόρηση όλων των σημείων εκπαίδευσης, τότε το υπολογιστικό κόστος θα ήταν απίθανα πιο ψηλό, ιδικά όταν όταν η διάσταση του χώρου ήταν ψηλή.

Μετρικό της πληροφόρησης

Οι πιο κάτω συμβολισμοί εμφανίζονται στις εξισώσεις
- Q : query point (σημείο δοκιμής)
- k : k κοντινότεροι γείτονες
- I : τα σημεία που είναι πιο πληροφοριακά
- x_i : το i-οστό διάνυσμα χαρακτηριστικών
- x_{ij} : το j-οστό χαρακτηριστικό του x_i
- y_i : η κλάση στην οποία ανήκει το
- N : το σύνολο σημείων εκπαίδευσης (training points)

Για τον καθορισμό της πληροφόρησης κάθε ενός από τα σύνολα εκπαίδευσης $\{x_j, y_j\}^N$ για ένα συγκεκριμένο query point χρησιμοποιείται η πιο κάτω εξίσωση [55]:

$$I\left(x_j \mid Q = x_i\right) = -log\left(1 - P\left(x_j \mid Q = x_i\right)\right) * P\left(x_j \mid Q = x_i\right) \quad j = 1,...N, j \neq i$$

όπου το $P\ (x_j \mid Q = x_i)$ είναι η πιθανότητα που το σημείο x_j είναι πληροφοριακό. Άρα αν το x_j είναι κοντά στο Q η πιθανότητα ισούται με 1, σε αντίθετη περίπτωση η πιθανότητα ισούται με 0. Η τιμή της πιθανότητας αυτής βρίσκεται από την πιο κάτω εξίσωση [5]]:

$$P\left(x_j \mid Q = x_i\right) = \frac{1}{Z_i}\left\{ Pr\left(x_j \mid Q = x_i\right)^{\eta}\left(\prod_{n=1}^{N}\left(1 - Pr\left(x_j \mid Q = x_n \mid\right)_{\left[y_j \neq y_n\right]}\right)\right)^{1-\eta}\right\}$$

όπου για να βρούμε την πιθανότητα $Pr(x_j \mid Q = x_i)$, μπορούμε να βρούμε την απόσταση μεταξύ του σημείου Q και του σημείου εκπαίδευσης. Άρα σχηματίζεται η πιο κάτω εξίσωση [55]:

$$Pr\left(x_j \mid Q = x_i\right) = f\left(\left\|x_i - x_j\right\|_p\right)$$

Όμως θέλουμε όσο μικρότερη είναι η απόσταση $||x_i - x_j||_p$, τόσο πιο ψηλή να είναι η πιθανότητα. Για' αυτό η $f(.)$ πρέπει να είναι αντιστρόφως ανάλογη με την απόσταση αυτή. Έτσι η εξίσωση μετατρέπεται όπως φαίνεται πιο κάτω [55]:

$$Pr\left(x_j \mid Q = x_i\right) = exp\left(-\frac{\left\|x_i - x_j\right\|^2}{\gamma}\right) \qquad \gamma > 0$$

Όμως δεν έχουν όλα τα χαρακτηριστικά ενός σημείου την ίδια σημαντικότητα, και έτσι κάθε χαρακτηριστικό έχει το δικό του βάρος.
Οπότε η απόσταση ορίζεται ως

$$\left\|x_i - x_j\right\|^2 = \sum_p w_p \left(x_{ip} - x_{jp}\right)^2$$

όπου w_p είναι η κλιμάκωση (scaling) που αντιστοιχεί στην σημαντικότητα ενός χαρακτηριστικού ,η οποία μπορεί να οριστεί με την εξίσωση [55]:

$$w_p = \frac{1}{m}\sum_{k=1}^{m} w_{pk} = \frac{1}{m}\sum_{k=1}^{m} Var_{x_k}\left(x_{pk}\right)$$

4.1.3 Συνελικτικά Νευρωνικά Δίκτυα

Το CNN είναι παραλλαγή του πολυεπίπεδου νευρωνικού δικτύου (multi-layer neural network) το οποίο αποτελείται από πολλαπλά επίπεδα και είναι παρόμοιο με ένα νευρώνα. Είναι το πρώτο

επιτυχές δίκτυο βαθιάς μάθησης, όπου τα πολλαπλά επίπεδα της ιεραρχίας είναι επιτυχώς εκπαιδευμένα με ένα ισχυρό τρόπο. Μέσω των πολλαπλών επιπέδων διακινούνται πληροφορίες, όπου κάθε επίπεδο είναι υπεύθυνο για να εξάγει τα κυριότερα χαρακτηριστικά από αυτές της πληροφορίες. Το κυρίως κίνητρο του CNN είναι οι ελάχιστες απαιτήσεις προεπεξεργασίας δεδομένων που χρειάζεται. Ακόμα ένα κίνητρο του CNN είναι μαθαίνει πολύ λιγότερες παραμέτρους απ' ότι ένα αντίστοιχο (με τον ίδιο αριθμό κρυφών μονάδων) πλήρως συνδεδεμένο δίκτυο. Το CNN χρησιμοποιείται σε διασδιάστατα δεδομένα, όπως είναι η αναγνώριση εικόνας, ήχου, βίντεο, χρονικά δεδομένα (time-siries data), δεδομένα καταγραφής κίνησης (motion capture data) κτλ [1].

Το CNN αποτελείται πρώτα από συνελικτικά επίπεδα (convolutional layers) τα οποία ακολουθούνται από τα subsampling επίπεδα. Τα subsampling επίπεδα προαιρετικά ακολουθούνται από πλήρως συνδεδεμένα επίπεδα. Η είσοδος που λαμβάνει ένα συνελικτικό επίπεδο είναι μια εικόνα με διαστάσεις m x m x r, όπου m είναι το ύψος και το πλάτος της εικόνας και το r είναι ο αριθμός των καναλιών (channels) στον όγκο της εικόνας (π.χ. μια εικόνα με 3 κανάλια είναι μια έγχρωμη εικόνα με κόκκινο, πράσινο και μπλε). Η εικόνα συνελίσσεται με ένα σύνολο από k φίλτρα (πυρήνες), τα οποία έχουν μικρότερες διαστάσεις από την εικόνα και μικρότερο είτε ίδιο αριθμό καναλιών με την εικόνα. Έτσι το αποτέλεσμα των συνελικτικών διαδικασιών είναι οι χάρτες χαρακτηριστικών (feature maps). Το επόμενο επίπεδο του CNN είναι το επίπεδο subsampling, το οποίο έχει ως έξοδο ένα δείγμα κάθε χάρτη χαρακτηριστικών εφαρμόζοντας συγκέντρωση (pooling) στους γειτονικούς του νευρώνες. Τέλος, είτε πριν είτε μετά το επίπεδο subsampling σε κάθε χάρτη χαρακτηριστικών εφαρμόζεται κανονικοποίηση, δηλαδή κανονικοποιούνται οι διαφορές γειτονικών τιμών χαρακτηριστικών[67,1].

4.2 Μη-επιβλεπόμενη Μάθηση

Στην μη-επιβλεπόμενη μάθηση δεν υπάρχουν προκαθορισμένες κατηγοριοποιήσεις (unlabel data). Για τον λόγο αυτό, γίνεται χρήση κάποιου συστήματος ποιότητας (reward system) για να δείξει την επιτυχία. Έτσι ο αλγόριθμος μάθησης του ομαδοποιητή προσπαθεί να βρει ομοιότητες μεταξύ των διανυσμάτων χαρακτηριστικών των προτύπων που παίρνει στην είσοδο, και δημιουργεί αυτόματα κλάσεις με τα όμοια πρότυπα εισόδου.
Τα πλεονεκτήματα τέτοιων δικτύων μη-εποπτευόμενης μάθησης είναι ότι είναι πιο εύκολο στο να συνθέσουν χαρακτηριστικά, να χειριστούν την αβεβαιότητα, να ενσωματώσουν γνώσεις, να ερμηνεύσουν.

Πιο κάτω θα περιγραφούν οι εξής αλγόριθμοι μη επιβλεπόμενης μάθησης, ο k-Means, τα Βαθιά Δίκτυα Πίστης και ο Χάρτης Αυτο-οργάνωσης.

4.2.1 Συσταδοποίηση

Στόχος της ομαδοποίησης ή αλλιώς συσταδοποίησης είναι ο διαχωρισμός των δεδομένων (ή διανυσμάτων) σε ομάδες (συστάδες-clusters) σύμφωνα με ένα μέτρο απόστασης. Ο σκοπός είναι τα δεδομένα μιας ομάδας να είναι όσο το δυνατόν πιο όμοια μεταξύ τους και τα δεδομένα από διαφορετικές ομάδες να είναι όσο το δυνατόν πιο ανόμοια. [59, 60] Με τον όρο όμοια εννοούμε να έχουν μικρή απόσταση μεταξύ τους, ενώ με τον όρο ανόμοια εννοούμε να έχουν μεγάλη απόσταση μεταξύ τους.

Η ομαδοποίηση εντάσσεται στην κατηγορία των αλγορίθμων μη επιβλεπόμενης μάθησης, δηλαδή δεν γνωρίζουμε ποια δεδομένα ανήκουν σε ποια κλάση (unlabel data), αλλά οι ομάδες αποφασίζονται από τον αλγόριθμο (fit method). Έτσι ο αλγόριθμος επιστρέφει ένα πίνακα με ακέραιες ετικέτες (labels), μια για κάθε δείγμα, για να καθορίσει σε ποια ομάδα ανήκει το καθένα (π.χ. υπάρχουν δύο ομάδες με labels 1 και 0 αντίστοιχα). Αν οι ομάδες των δεδομένων ήταν προκαθορισμένες (label data), τότε θα στρεφόμασταν προς ένα αλγόριθμο κατηγοριοποίησης δεδομένων (data classification) [61, 62].

Υπάρχει μια πληθώρα αλγορίθμων ομαδοποίησης. Υπάρχουν δύο βασικές στρατηγικές ομαδοποίησης των αλγορίθμων αυτών:

- Διαμεριστική Συσταδοποίηση (Partitioning Relocation Clustering)
  Οι διαμεριστικοί αλγόριθμοι έχουν ένα προκαθορισμένο αριθμό ομάδων και τοποθετούν τα σημεία στις ομάδες αυτές. Ο σκοπός είναι τα σημεία να τοποθετηθούν στην ομάδα που ταιριάζουν καλύτερα με βάση κάποια μετρική. Οι αναθέσεις των σημείων αυτών επαναπροσδιορίζονται έως ότου κάποιο κριτήριο τερματισμού εκπληρωθεί.

- Ιεραρχική Συσταδοποίηση (Hierarchical Clustering)
  Οι ιεραρχικοί αλγόριθμοι αρχίζουν με μια ομάδα για κάθε σημείο. Οι ομάδες συγχωνεύονται με την χρήση κάποιου ορισμού "εγγύτητας". Σταματούν να γίνονται συγχωνεύσεις όταν μια επιπλέον συγχώνευση οδηγήσει σε κάποιο μη επιθυμητό αποτέλεσμα. Υπάρχουν διάφοροι λόγοι που καθορίζουν αυτόν τον τερματισμό. Ένας λόγος είναι να έχουμε προκαθορισμένο αριθμό ομάδων και να ξέρουμε πότε πρέπει να σταματήσουμε. Αυτό συμβαίνει όταν ξέρουμε σε ποιες ομάδες θα ανήκουν τα δεδομένα μας. Ακόμα ένας λόγος, είναι η νέα ομάδα που θα προκύψει να μην αποφέρει καλύτερη μοντελοποίηση από την προηγούμενη ομαδοποίηση με βάση κάποιο μέτρο (π.χ. διάμετρος, διακύμανση) [62].

4.2.1.1 Αλγόριθμος k-Means

Ο αλγόριθμος k-Means είναι από τους πιο δημοφιλής αλγορίθμους ομαδοποίησης, λόγω της απλότητας της υλοποίησης του και του ότι μπορεί να εφαρμοστεί σε ένα σύνολο από ένα μεγάλο αριθμό δειγμάτων. Ανήκει στην κατηγορία Διαμεριστικής Συσταδοποίησης. Ο αλγόριθμος αυτός ομαδοποιεί τα δεδομένα σε k συστάδες ίσης διακίμανσης (variance), αρκεί να καθοριστεί εκ των προτέρων ο αριθμός k. Η κάθε συστάδα περιγράφεται από την μέση τιμή (means) των δειγμάτων από τα οποία αποτελείται, η οποία ονομάζεται κέντρο βάρους της ομάδας (centroid). Ο αλγόριθμος επιλέγει centroids που ελαχιστοποιούν το άθροισμα των τετραγώνων (sum of squares) εντός της συστάδας, το οποίο περιγράφεται από την πιο κάτω αντικειμενική συνάρτηση :

$$\sum_{i=0}^{n} \min_{\mu_j \in C} \left(\left\| x_j - \mu_i \right\|^2 \right)$$

όπου x_j είναι είναι ένα διάνυσμα (δεδομένο) d-διαστάσεων, μ_i είναι η μέση τιμή της συστάδας i στην οποία ανήκει το διάνυσμα και n είναι το μέγεθος του συνόλου δεδομένων.

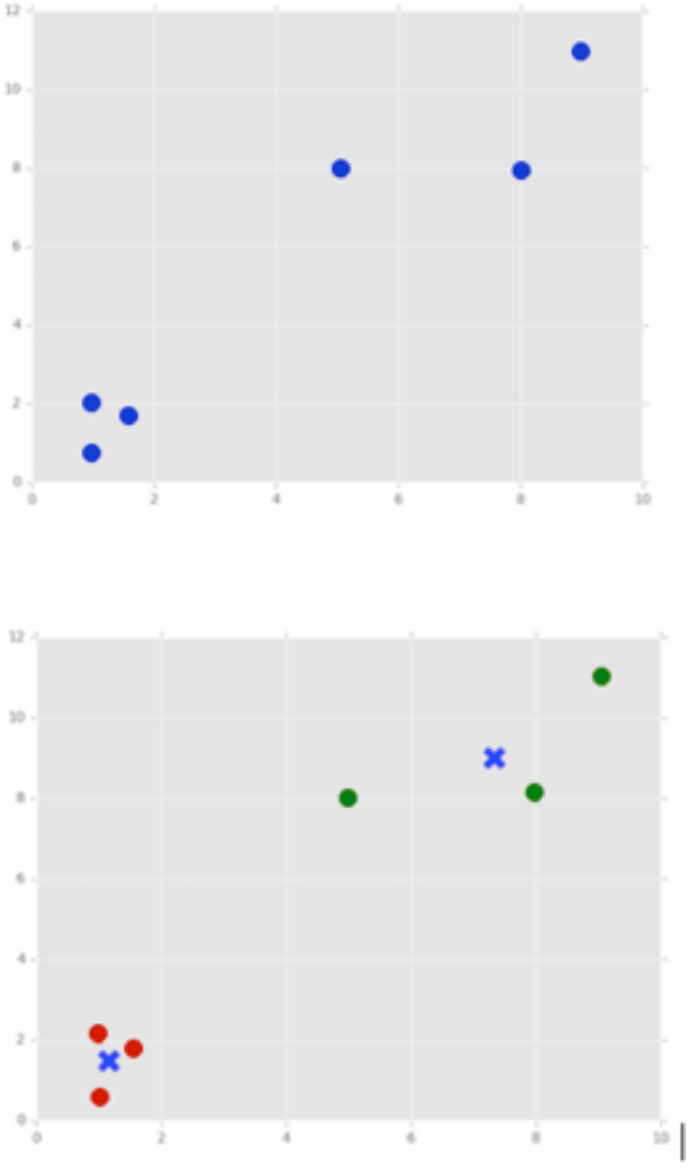
Το άθροισμα τετραγώνων είναι ένα μέτρο απόστασης που χρησιμεύει για να μετρά την απόσταση κάθε διανύσματος x_j από το centroid μ_i της σηστάδας στην οποία ανήκει.

Αυτό το μέτρο απόστασης που χρησιμοποιείται επηρεάζει το αποτέλεσμα του αλγορίθμου. Το μειονέκτημα του είναι ότι προϋποθέτει ότι τα δεδομένα των συστάδων βρίσκονται σε κυρτό(σφαιρικό) σχήμα και ίδιο μέγεθος. Δυσκολεύεται να αναγνωρίσει συστάδες με ακανόνιστο σχήμα και διαφορετικό μέγεθος. Το πρόβλημα αυτό εμφανίζεται σε χώρους πολύ μεγάλων διαστάσεων, όπου οι Ευκλείδειες αποστάσεις τείνουν να γίνουν πολύ μεγάλες, το οποίο αναφέρεται ως η ”κατάρα διαστασιμότητας”-curse of dimensionality. Ένα συνέπεια αυτού του προβλήματος είναι ότι τα ζεύγη σημείων απέχουν το ίδιο το ένα από το άλλο. Ακόμα μια συνέπεια είναι ότι δύο διανύσματα είναι κάθετα μεταξύ τους. Το πρόβλημα αυτό μπορεί να λυθεί με εφαρμογή κανονικοποίησης στα δεδομένα πριν από την εφαρμογή του αλγορίθμου. Ένας τρόπος είναι ένας αλγόριθμος μείωσης διάστασης, όπως είναι ο PCA.

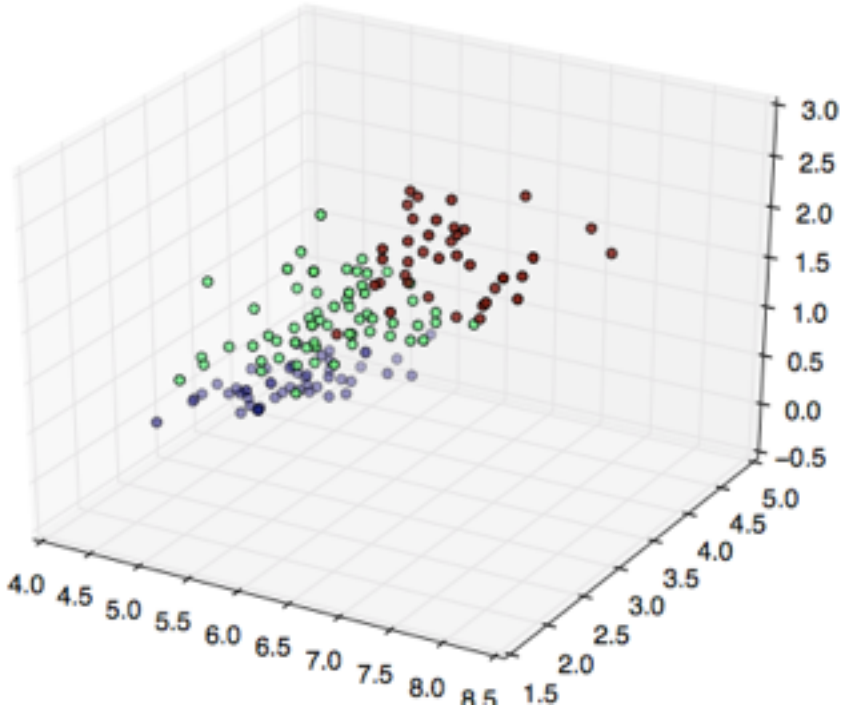
Ας περιγράψουμε τα τρία βήματα που ακολουθεί ο αλγόριθμος k-means. Πρώτα αρχικοποιεί τα centroids. Ας επισημάνουμε ότι τα centroids δεν είναι σημεία από τα δεδομένα, αν και ζουν στον ίδιο χώρο με αυτά. Η αρχικοποίηση των centroids επηρεάζει το τελικό αποτέλεσμα του αλγόριθμου, έτσι επιλέγονται να είναι όσο πιο μακρυά το ένα από το άλλο. Υπάρχουν διάφορες μεθόδοι για μια αποτελεσματική αρχικοποίηση των centroids. Μια από αυτές είναι η k-means ++, η οποία τοποθετεί τα centroids όσο το δυνατόν πιο μακρυά το ένα από

το άλλο. Η βασική μέθοδος, η οποία επιλέγει k σημεία από το σύνολο δεδομένων για την αρχικοποίησή τους. Στο δεύτερο βήμα ο αλγόριθμος τοποθετεί κάθε δεδομένο στην ομάδα που έχει το πιο κοντινό centroid απ' αυτό. Και το τρίτο βήμα είναι η δημιουργία νέων centroids τα οποία θα ισούνται με την μέση τιμή των δειγμάτων που περιέχει κάθε συστάδα που δημιουργήθηκε στο προηγούμενο βήμα. Τα δύο τελευταία βήματα επαναλαμβάνονται μέχρις ότου η διαφορά μεταξύ του προηγούμενου και του νέου centroid δεν είναι σημαντική [63].

Το αποτέλεσμα σίγουρα εξαρτάται από τον αριθμό των συστάδων k, που παίρνει σαν είσοδο ο αλγόριθμος. Μερικές φορές δεν είναι δυνατόν να γνωρίζουμε τον αριθμό των k συστάδων. Έτσι πριν την εφαρμογή του αλγορίθμου k-means πρέπει να γίνει κάποια εκτίμηση του βέλτιστου αριθμού k. Ο καθορισμός του βέλτιστου k είναι κάπως υποκειμενικός, αφού εξαρτάται από το χαρακτηριστικό που χρησιμοποιείται για την προσδιορισμό ομοιοτήτων και από το επίπεδο λεπτομέρειας που απαιτείται. Δηλαδή, η εκτίμηση του k που αποφέρει τον καλύτερο διαχωρισμό μεταξύ των δεδομένων, δίνοντας ως αποτέλεσμα συστάδες που επιθυμεί ο χρήστης. Η αύξηση του k μειώνει τα σφάλματα που αποφέρει η συσταδοποίηση, με ακραία περίπτωση κάθε δεδομένα να έχει τη δική του συστάδα. Υπάρχουν διάφοροι μέθοδοι για την επιλογή του καταλληλότερου k.



Σχήμα 4.3 : Πιο πάνω φαίνεται ένα απλό παράδειγμα συσταδοποίησης με χρήση του αλγορίθμου k-means (από την βιβλιοθήκη sklearn), όπου η πρώτη εικόνα παρουσιάζει το αρχικό σύνολο δεδομένων και η δεύτερη το σύνολο των δεδομένων αυτών σε ομάδες. Τα δεδομένα του προβλήματος όπως φαίνεται και πιο κάτω είναι σημεία με συνιστώσες (x,y), τα οποία θέλαμε να ομαδοποιηθούν σε δύο ομάδες. Στην δεύτερη εικόνα τα μπλε σημεία σε σχήμα "x" αντιπροσωπεύουν τα κέντρα βάρους (centroids) των δύο συστάδων και τα σημεία με ίδιο χρώμα ανήκουν στην ίδια συστάδα. (Βλέπε στο παράρτημα Α : kmeans.py)



Σχήμα 4.4 : Πιο πάνω φαίνεται ένα ακόμα παράδειγμα συσταδοποίησης με χρήση του αλγορίθμου k-means (από την βιβλιοθήκη sklearn). Τα δεδομένα του προβλήματος είναι από την βιβλιοθήκη datasets της sklearn και περιέχουν τρία χαρακτηριστικά για κάθε λουλούδι ίρις, το μήκος του φύλλου, το πλάτος του φύλλου και το μήκος του πετάλου. Όπως φαίνεται στο σχήμα τα λουλούδια διαχωρίζονται σε τρεις συστάδες, όπου τα δεδομένα της κάθε ομάδας έχουν το ίδιο χρώμα. (Βλέπε στο παράρτημα Β : iris.py)

4.2.1.2 Η μέθοδος του “Αγκώνα"

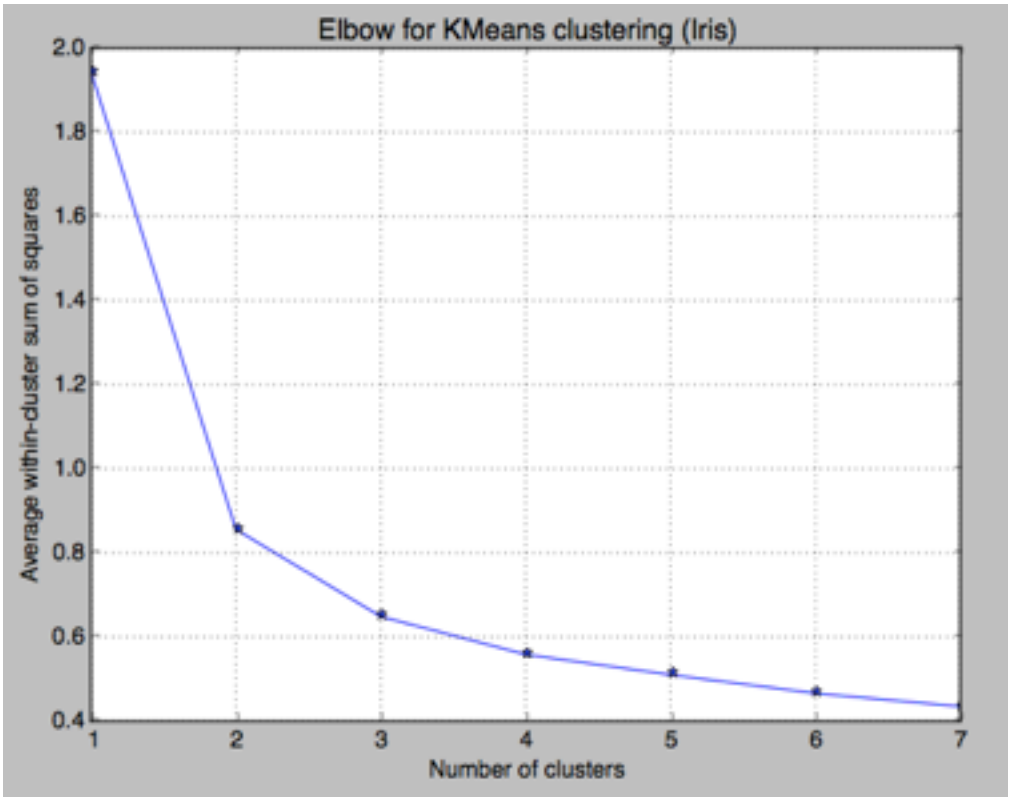
Όπως αναφέραμε μέχρι τώρα ο αλγόριθμος k-means μπορεί να εκτελεστεί για ένα μοναδικό k, το οποίο θα αποφασίσει ο χρήστης. Για να βρούμε το βέλτιστο k πρέπει να τρέξουμε τον αλγόριθμο k-means για ένα διάστημα (range) από πιθανές τιμές για το k, έτσι ώστε με ένα κριτήριο να αποφασίσουμε ποια από αυτές είναι η πιο κατάλληλη. Το κριτήριο που θα χρησιμοποιήσουμε είναι η μέθοδος "αγκώνα" (Elbow Method) που εξετάζει το ποσοστό της διακύμανσης ως προς τον αριθμό των συστάδων (k). Το ποσοστό της διακύμανσης είναι ο λόγος της διακύμανσης μεταξύ των ομάδων σε σχέση με την συνολική διακύμανση. Στόχος αυτής της μεθόδου είναι να βρει το βέλτιστο k, έτσι ώστε η πρόσθεση ακόμα μιας συστάδας να μην αποφέρει πολύ καλύτερη μοντελοποίηση των δεδομένων. Σχηματίζοντας μια γραφική παράσταση συναρτήσει της διακύμανσης ως προς τον αριθμό των συστάδων, θα προσέξουμε ότι οι πρώτοι αριθμοί k δίνουν πολλή πληροφόρηση (διακύμανση), αλλά σε κάποιο σημείο το οριακό κέρδος θα μειωθεί, δίνοντας μια γωνία στο γράφημα (εξ ου και το όνομα του αλγορίθμου "αγκώνας"). Αυτό το σημείο που σχηματίζει γωνία, δίνει τον βέλτιστο αριθμό συστάδων (k) [64].

Η πιο κάτω εξίσωση είναι το άθροισμα των αποστάσεων μεταξύ των σημείων εντός μιας συστάδας C_k που περιέχει n_k σημεία (intra-cluster sum of squares-Ευκλείδεια απόσταση):

$$D_k = \sum_{x_i \in C_k} \sum_{x_j \in C_k} \left\| x_i - x_j \right\|^2 = 2n_k \sum_{x_i \in C_k} \left\| x_i - \mu_k \right\|^2$$

Αν προσθέσουμε τα κανονικοποιημένα αθροίσματα των τετραγώνων μιας συστάδας, βρίσκουμε την πυκνότητα (διακύμανση) της συστάδας [60]:

$$W_k = \sum_{k=1}^{K} \frac{1}{2n_k} D_k$$



Σχήμα 4.5 : Πιο πάνω φαίνεται ένα παράδειγμα εφαρμογής της μεθόδου αγκώνα στα δεδομένα iris από την βιβλιοθήκη datasets της sklearn. Οι πιθανές τιμές για τον αριθμό συστάδων-k που δόθηκαν στον αλγόριθμο kmeans είναι το διάστημα 1 ως 8. Όπως φαίνεται από την γραφική παράσταση ο βέλτιστος αριθμός συστάδων-k είναι το 3, αφού σε εκείνο το σημείο σχηματίζεται μια γωνία. (Βλέπε στο παράρτημα Β :clustergram_iris.py)

4.2.1.3 Ανάλυση Κύριων συνιστωσών (Principal Component Analysis / PCA)

Σε πολλές περιπτώσεις τα σύνολα δεδομένων έχουν πολλές μεταβλητές, ακόμα και περισσότερες από τον αριθμό των δειγμάτων. Αυτή η ψηλή πολυδιαστατικότητα κάνει δύσκολη την οπτικοποίηση (visualization) των δειγμάτων και την εξερεύνηση των δεδομένων. Γι' αυτό τον σκοπό χρησιμοποιείται ο αλγόριθμος PCA, ο οποίος μειώνει την διάσταση των δεδομένων.

Τα διανύσματα πριν την επεξεργασία ζουν σε ένα χώρο n διαστάσεων, όπου κάθε μεταβλητή (χαρακτηριστικό) απεικονίζεται με ένα άξονα. Με την εφαρμογή του PCA αλγορίθμου, τα διανύσματα αυτά μεταφέρονται σε ένα χώρο χαμηλότερων διαστάσεων, όπου οι νέες μεταβλητές είναι γραμμικοί συνδυασμοί των αρχικών μεταβλητών. Αυτές οι νέες μεταβλητές ονομάζονται κύριες συνιστώσες. Οι αρχικές μεταβλητές είναι συσχετισμένες μεταξύ τους, ενώ οι κύριες συνιστώσες ασυσχέτιστες. Ο αριθμός των κύριων συνιστωσών δεν μπορεί να είναι μεγαλύτερος από τον αριθμό των αρχικών μεταβλητών.
Η πρώτη κύρια συνιστώσα έχει τη μεγαλύτερη δυνατή διακύμανση (δηλαδή εκφράζει όσο το δυνατόν πιο πολύ από την μεταβλητότητα των δεδομένων). Η δεύτερη κύρια συντεταγμένη έχει την δεύτερη μεγαλύτερη διακύμανση. Και αντίστοιχα, η τελευταία κύρια συνιστώσα έχει την πιο χαμηλή διακύμανση. Όλες μαζί οι κύριες συνιστώσες εκφράζουν το 100% της διακύμανσης. Έτσι επιλέγοντας ένας μέρος των συνιστωσών, κάθε δείγμα μπορεί να αναπαρασταθεί από λίγες μεταβλητές, αντί από τον μεγάλο αριθμό των αρχικών μεταβλητών. Δηλαδή αφαιρούμε τις συνιστώσες που καλύπτουν ελάχιστη διακύμανση και κρατούμε μόνο αυτές που καλύπτουν ένα μεγάλο μέρος της.

Έτσι μετά από αυτό τον μετασχηματισμό, τα δείγματα ευκολύνεται η οπτικοποίηση των δειγμάτων και η εκτίμηση των διαφορών και ομοιοτήτων τους έτσι ώστε να δούμε αν τα δείγματα θα μπορέσουν να ομαδοποιηθούν [65, 66].

4.3 Δίκτυα Βαθιάς Πίστης

Το DBN είναι ένα είδος βαθιού νευρωνικού δικτύου, το οποίο είναι πιθανοτικό γενετικό μοντέλο [1, 22]. Αποτελείται από πολλαπλά επίπεδα Περιορισμένων Boltzmann μηχανών (Restricted Boltzmann Machines-RBMs). Μια Boltzmann μηχανή είναι ένα είδος νευρωνικού δικτύου χωρίς επίβλεψη, που αποτελείται από ένα ορατό επίπεδο (visible layer) και ένα κρυφό επίπεδο (hidden layer). Το ορατό επίπεδο αποτελείται από ορατές μονάδες (visible units) και το κρυφό

επίπεδο από κρυφές μονάδες (hidden units). Υπάρχουν συνδέσεις μεταξύ αυτών των δύο επιπέδων, αλλά όχι μεταξύ των μονάδων μέσα σε κάθε επίπεδο. Στο ορατό επίπεδο παρατηρούνται συσχετισμοί δεδομένων, ενώ το κρυφό επίπεδο είναι αυτό που συλλέγει αυτούς τους συσχετισμούς. Αυτά τα δύο επίπεδα αποτελούν την συσχετιστική μνήμη (associative memory). Τα επίπεδα του DBN συνδέονται από top-down generative weights. Για την απόκτηση αυτών των βαρών, γίνεται μια άπληστη διαδικασία εκπαίδευσης από επίπεδο σε επίπεδο, όπου σε κάθε επίπεδο εφαρμόζεται αντιπαράθεση απόκλισης σύμφωνα με τον Hinton [1, 22].

4.4 Χάρτης Αυτο-οργάνωσης

Ο χάρτης αυτο-οργάνωσης (SMO) ή χάρτης αυτο-οργάνωση χαρακτηριστικού (SOFM) είναι ένας τύπος μη επιβλεπόμενου τεχνικού νευρωνικού δικτύου. Ενώ άλλοι τύπου τεχνητού νευρωνικού δικτύου εφαρμόζουν μάθηση διόρθωσης σφάλματος (error-correction learning), το SMO εφαρμόζει ανταγωνιστική μάθηση.

Δομή ενός χάρτη αυτο-οργάνωσης:

Ο χάρτης αυτο-οργάνωσης προσπαθεί να μιμηθεί την λειτουργία των νευρώνων του εγκεφάλου. Έτσι ο χάρτης αυτός αναπαριστάται από κόμβους και ακμές. Κάθε κόμβος του χάρτη αντιστοιχεί σε ένα νευρώνα του εγκεφάλου και κάθε ακμή πάνω σε κάποιο κόμβο αντιστοιχεί σε ένα διάνυσμα βάρους της ίδιας διάστασης. Οι νευρώνες είναι διατεταγμένοι σε δισδιάστατο πλέγμα (εξαγωνικό ή ορθογώνιο). Ο χάρτης χαρτογράφησης, χαρτογραφεί τις εισόδους ενός ψηλότερου χώρου διάστασης σε ένα χάρτη με χαμηλή διάσταση χώρου.

Μάθηση:

Το πρώτο βήμα της μάθησης SOM είναι η αρχικοποίηση των βαρών των νευρώνων. Τα βάρη μπορεί να αρχικοποιηθούν είτε με μικρές τυχαίες τιμές, είτε τα βάρη παράγονται εκ των προτέρων. Στην πρώτη περίπτωση όλοι οι νευρώνες στο χάρτη αποκτούν την ίδια προτεραιότητα. Όμως στη δεύτερη περίπτωση η μάθηση γίνεται σε πολύ λιγότερο χρόνο.

Για κάθε πρότυπο εισόδου υπολογίζεται η Ευκλείδεια απόστασή του προς το διάνυσμα βάρους κάθε νευρώνα και βρίσκεται ο πλησιέστερος νευρώνας από το πρότυπο εισόδου, ονομαζόμενος και ως νικητής. Έτσι η μάθηση ονομάζεται ανταγωνιστική, λόγω αυτής της ανταγωνιστικότητας των νευρώνων για το ποιος θα υπερισχύσει. Για την τοποθέτηση του προτύπου εισόδου στο χάρτη, πρέπει τα διανύσματα βάρους του νικητή και των νευρώνων της τοπολογικής γειτονιάς του νικητή(δηλαδή κοντά στον νικητή) στο πλέγμα, να προσαρμοστούν

προς αυτό το διάνυσμα εισόδου. Έτσι το ενημερωμένο διάνυσμα βαρών κάθε νευρώνα v με $W_v(s)$ υπολογίζεται από τον πιο κάτω τύπο, ο οποίος θα εφαρμοστεί σε κάθε νευρώνα της τυπολογικής γειτονιάς του νικητή στο πλέγμα :

$$W_v(s + 1) = W_v(s) + \Theta(u, v, s)\, \alpha(s)(D(t) - W_v(s)),$$

όπου s = δείκτης βήματος (step index),
t = ο δείκτης στο δείγμα εκπαίδευσης,
$D(t)$ = το διάνυσμα εισόδου,
υ = ο δείκτης του νικητή
$\alpha(s)$ = συντελεστής μονοτονικής φθίνουσας μάθησης
$\Theta(u, v, s)$ = η συνάρτηση γειτνίασης

Η συνάρτηση γειτονιάς $\Theta(u, v, s)$ δίνει την απόσταση μεταξύ του νευρώνα u και του νευρώνα v στο s βήμα. Η πιο απλή έκδοση αυτής της συνάρτησης δίνει 1 για όλους τους νευρώνες κοντά στον νικητήριο νευρώνα και 0 στους υπόλοιπους. Στην αρχή του αλγορίθμου η συνάρτηση εκτελείται σε όλους τους νευρώνες ως προς τον νικητή και σταδιακά η γειτονιά μειώνεται σε λίγους νευρώνες [56,57].

Κεφάλαιο 5

Αξιολόγηση Απόδοσης Κατηγοροποιητή - Ομαδοποιητή

Οι κατηγοριοποιητές/συσταδοποιητές που θα εφαρμόσουμε στην συνέχεια της εργασίας μπορούν να αξιολογηθούν ως προς την απόδοση τους από διάφορα στατιστικά μέτρα. Τα μέτρα που θα χρησιμοποιήσουμε είναι η ακρίβεια (accuracy), η ευαισθησία (sensitivity) και η προσδιοριστικότητα (specificity). Αυτά τα στατιστικά μέτρα εφαρμόζονται μόνο για δοκιμή δυαδικής κατηγοριοποίησης /συσταδοποίησης, δηλαδή κατηγοροποίηση δειγμάτων σε δύο κλάσεις. Επίσης θα παρουσιαστεί και ο πίνακας σύγχυσης (confusion matrix) για κάθε κατηγοροποιητή, ο οποίος βοηθά στην οπτικοποίηση της απόδοσης του ταξινομητή. Ακόμα αναφέραμε και ένα επιπλέον μέτρο που θα χρησιμοποιήσουμε για την συσταδοποίηση, την τιμή Silhouette.

Για τον υπολογισμό των πιο πάνω στατιστικών μέτρων που αναφέρθηκαν χρησιμοποιούνται οι ετικέτες (labels) που υπολόγισε ο υπό εξέταση κατηγοροποιητής για κάθε δείγμα (δηλαδή σε ποια κλάση ανήκει το κάθε δείγμα) και οι αντίστοιχες πραγματικές τους ετικέτες. Στις πιο κάτω επεξηγήσεις εννοιών αναφέρονται κάποιοι όροι:

Ορθό θετικό αποτέλεσμα : έχει αναγνωριστεί ορθά θετικό από τον κατηγοριοποιητή σύμφωνα με την πραγματική του ετικέτα

Λανθασμένα θετικό αποτέλεσμα : έχει αναγνωριστεί λανθασμένα θετικό από τον κατηγοριοποιητή σύμφωνα με την πραγματική του ετικέτα

Ορθά αρνητικό αποτέλεσμα : έχει απορριφθεί ορθά αρνητικά από τον κατηγοριοποιητή σύμφωνα με την πραγματική του ετικέτα

Λανθασμένα θετικό αποτέλεσμα : έχει απορριφθεί λανθασμένα αρνητικά από τον κατηγοριοποιητή σύμφωνα με την πραγματική του ετικέτα

(Ο Πίνακας Σύγχυσης και η τιμή Silhouette υπάρχουν υλοποιημένα στην βιβλιοθήκη **sklearn.**)

Ακρίβεια

Η ακρίβεια (Accuracy) είναι το στατιστικό μέτρο που εκφράζει το πόσο καλά ένας κατηγοριοποιητής εντοπίζει την ορθή κλάση των δειγμάτων. Δηλαδή ισούται με το ποσοστό των ορθών αποτελεσμάτων (δηλαδή και ορθών θετικών και ορθών αρνητικών) ως προς το συνολικό αριθμό των περιπτώσεων που κατηγοριοποιήθηκαν.

$$\text{ακρίβεια} = \frac{\text{αριθμός ορθών θετικών + ορθών αρνητικών}}{\text{αριθμός ορθών θετικών + ορθών αρνητικών + λανθασμένων θετικών + λανθασμένων αρνητικών}}$$

Από την εξίσωση παρατηρούμε ότι όσο πιο πιο μεγάλη είναι η ακρίβεια, τόσο πιο μεγάλη πιθανότητα υπάρχει τα δείγματα να κατηγοριοποιήθηκαν στην σωστή κλάση. Η ακρίβεια από μόνη της, χωρίς τον συνδυασμό και άλλων στατιστικών μέτρων μπορεί να είναι παραπλανητική όταν ο αριθμός των δειγμάτων σε διαφορετικές κατηγορίες είναι ασύμμετρος, δηλαδή στην μια κατηγορία να αντιστοιχούσαν 10 δείγματα και στην άλλη μόνο ένα [70].

Ευαισθησία

Η ευαισθησία (Sensitivity) είναι το στατιστικό μέτρο που εκφράζει το πόσο καλά ένας κατηγοριοποιητής εντοπίζει τα δείγματα που είναι θετικά στην πραγματικότητα. Δηλαδή ισούται με το ποσοστό των ορθών θετικών αποτελεσμάτων ως προς το συνολικό αριθμό των περιπτώσεων που είναι θετικές στην πραγματικότητα.

$$\text{ευαισθησία} = \frac{\text{αριθμός ορθών θετικών}}{\text{αριθμός ορθών θετικών + λανθασμένων αρνητικών}}$$

Από την εξίσωση παρατηρούμε ότι όσο πιο ψηλή είναι η τιμή της ευαισθησίας τόσο πιο πιθανό είναι ένα αρνητικό αποτέλεσμα να έχει κατηγοριοποιειθεί σωστά [71].

Προσδιοριστικότητα

Η προσδιοριστικότητα (Specificity) είναι το στατιστικό μέτρο που εκφράζει το πόσο καλά ένας κατηγοριοποιητής εντοπίζει τα δείγματα που είναι αρνητικά στην πραγματικότητα. Δηλαδή ισούται με το ποσοστό των ορθών αρνητικών αποτελεσμάτων ως προς το συνολικό αριθμό των περιπτώσεων που είναι αρνητικές στην πραγματικότητα.

$$\text{προσδιοριστικότητα} = \frac{\text{αριθμός ορθών αρνητικών}}{\text{αριθμός ορθών αρνητικών} + \text{ λανθασμένων θετικών}}$$

Από την εξίσωση παρατηρούμε ότι όσο πιο ψηλή είναι η τιμή της προσδιοριστικότητας τόσο πιο πιθανό είναι ένα θετικό αποτέλεσμα να έχει κατηγοριοποιειθεί σωστά [71].

Πίνακας Σύγχυσης

Ο πίνακας σύγχυσης (Confusion Matrix) βοηθά στην οπτικοποίηση της απόδοσης του κατηγοριοποιητή. Αποτελείται από δύο γραμμές και δύο στήλες που αναφέρουν τον αριθμό των ορθά θετικών, λανθασμένα θετικών, ορθά αρνητικών και λανθασμένα αρνητικών. Με βάση το πιο κάτω σχήμα παρατηρούμε ότι όλες οι ορθές προβλέψεις βρίσκονται στην δεξιά διαγώνιο του πίνακα και όλες οι υπόλοιπες τιμές που βρίσκονται έξω από την διαγώνιο είναι λανθασμένες [72].

	προβλεπόμενη θετική κατάσταση	**προβλεπόμενη αρνητική κατάσταση**
θετική κατάσταση	αριθμός ορθών θετικών	αριθμός λανθασμένων αρνητικών
αρνητική κατάσταση	αριθμός λανθασμένων θετικών	αριθμός ορθών αρνητικών

Τιμή Σιλουέτας

Η τιμή σιλουέτας (Silhouette value), συνδυάζει ένα στατιστικό μέτρο για την συνοχή και τον διαχωρισμό μιας συσταδοποίησης. Με πιο απλά λόγια μετρά την ομοιότητα κάθε αντικειμένου στην δική του συστάδα σε σύγκριση με τις άλλες συστάδες. Αυτό το μέτρο υπολογίζεται ως εξής για κάθε σημείο: (b - a) / max(a, b) όπου, **b** είναι η ελάχιστη μέση απόσταση του σημείου από τα σημεία που βρίσκονται σε διαφορετική συστάδα (the mean nearest-cluster distance) και **a** είναι η μέση απόσταση του σημείου από όλα τα υπόλοιπα σημεία που ανήκουν στην ίδια συστάδα (the mean intra-cluster distance).

Το μέτρο αυτό μπορεί να υπολογισθεί με οποιαδήποτε μετρική απόστασης, όπως η Ευκλείδεια απόσταση και η απόσταση Manhattan.

Η καλύτερη τιμή που μπορεί να δώσει ως αποτέλεσμα είναι το +1 και η χειρότερη τιμή είναι το -1. Τιμές κοντά στο 0 δείχνουν ότι οι συστάδες επικαλύπτονται (δηλαδή κάποια σημεία είναι στο όριο των δύο συστάδων). Από την άλλη οι αρνητικές τιμές δείχνουν ότι κάποιο σημείο είναι πιο παρόμοιο με διαφορετική συστάδα. Με λίγα λόγια επιδιώκουμε το a να είναι όσο το δυνατόν πιο μικρό και το b όσο το δυνατόν πιο μεγάλο.

Στην βιβλιοθήκη sklearn υπάρχει το μέτρο **silhouette_score**, το οποίο επιστρέφει το μέσο συντελεστή όλων των δειγμάτων. Επίσης υπάρχει το μέτρο **silhouette_samples**, το οποίο επιστρέφει μια τιμή silhouette για κάθε δείγμα [73,74].

Κεφάλαιο 6

Μελέτη Περίπτωσης 1 : Διάγνωση Εμπειρικής Αποφυγής σε Καπνιστές

Σε αυτό το κεφάλαιο θα εφαρμοστούν αλγόριθμοι επιβλεπόμενης και μη επιβλεπόμενης μάθησης σε ένα σύνολο δεδομένων καπνιστών που σχετίζονται με την εμπειρική αποφυγή (EA-experiential avoidance).Τα δεδομένα αυτά δημιουργήθηκαν από το τμήμα Ψυχολογίας του Πανεπιστημίου. Το σύνολο των δεδομένων αποτελείτο από 36 δείγματα καπνιστών. Για τον κάθε καπνιστή πάρθηκε ένας όγκος μετρήσεων από τα σήματα ECG, SCR και COR. Εμείς, μετά από κάποια πειράματα, αποφασίσαμε να χρησιμοποιήσουμε μόνο τις μετρήσεις του δεύτερου σήματος, SCR. Έγιναν αρκετοί συνδυασμοί μέχρι να καταλήξουμε σε αυτή την απόφαση. Δηλαδή πρώτα αξιολογήθηκε η απόδοση του αλγορίθμου με την χρήση κάθε σήματος ξεχωριστά και μετά έγιναν συνδυασμοί αυτών των σημάτων. Οι μετρήσεις δεν πάρθηκαν σε ένα συνεχόμενο διάστημα, αλλά, κατά την διάρκεια πέντε διαφορετικών διαστημάτων (tasks). Για τον λόγο αυτό ανάμεσα σε κάθε δύο χρονικά διαστήματα, δηλαδή από το τέλος του ενός μέχρι την έναρξη του άλλου, υπήρχαν ανεπιθύμητα σήματα, τα οποία και αφαιρέσαμε. Ο στόχος ύπαρξης του πρώτου διαστήματος ήταν απλά για να χαλαρώσει ο ασθενής. Στα επόμενα τέσσερα διαστήματα ο ασθενής κλήθηκε να απαντήσει σε κάποια γνωστικά τεστς. Όλο το πείραμα

διάρκεσε οκτώ λεπτά. Στόχος των αλγορίθμων είναι η κατηγοριοποίηση-ομαδοποίηση των διανυσμάτων σε δύο κλάσεις, **την κλάση 0 για τους καπνιστές με χαμηλή εμπειρική αποφυγή (αρνητική κατάσταση) και 1 για τους καπνιστές με ψηλή εμπειρική αποφυγή (θετική κατάσταση).** Το Τμήμα Ψυχολογίας μου παρείχε την πραγματική κλάση στην οποία ανήκει ο κάθε ασθενής, δηλαδή ποιοι από τους ασθενείς αποδέχονται την εμπειρική αποφυγή και ποιοι δεν την αποδέχονται. Έτσι οι κλάσεις που θα προβλέψει ο κάθε αλγόριθμους για κάθε ασθενή, θα συγκριθούν με την χρήση κάποιων μετρικών αξιολόγησης, με τις πραγματικές κλάσεις, έτσι ώστε να μετρηθεί η απόδοση του αλγορίθμου.

6.1 Εφαρμογή Αλγορίθμων Επιβλεπόμενης Μάθησης

Οι πιο κάτω αλγόριθμοι επιβλεπόμενης μάθησης παίρνουν ως είσοδο ένα σύνολο με δείγματα εκπαίδευσης (training data set) με 25 ασθενείς και τις κλάσεις (labels) στις οποίες ανήκουν. Μετά από την εκπαίδευση του κατηγοριοποιητή, του δίνουμε ένα σύνολο με δείγματα δοκιμής (test data set) με 8 ασθενείς για να δούμε την ακρίβεια, την ευαισθησία και τη προσδιοριστικότητα, με την οποία θα προβλέψει τις κλάσεις των δειγμάτων δοκιμής. Χρησιμοποιήσαμε ως ορθές κλάσεις για κάθε δείγμα ασθενή αυτές που μας δόθηκαν από το Τμήμα Ψυχολογίας.

Κατηγοροποιητής SVM

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	3	0
αρνητική κατάσταση	1	4

Ακρίβεια : 7 / 8 = 0.875

Ευαισθησία : 3 / 3 = 1

Προσδιοριστικότητα : 4 / 5 = 0.8

Κατηγοροποιητής KNN (k = 3)

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	3	0
αρνητική κατάσταση	2	3

Ακρίβεια : 6 / 8 = 0.75

Ευαισθησία : 3 / 3 = 1

Προσδιοριστικότητα : 3 / 5 = 0.6

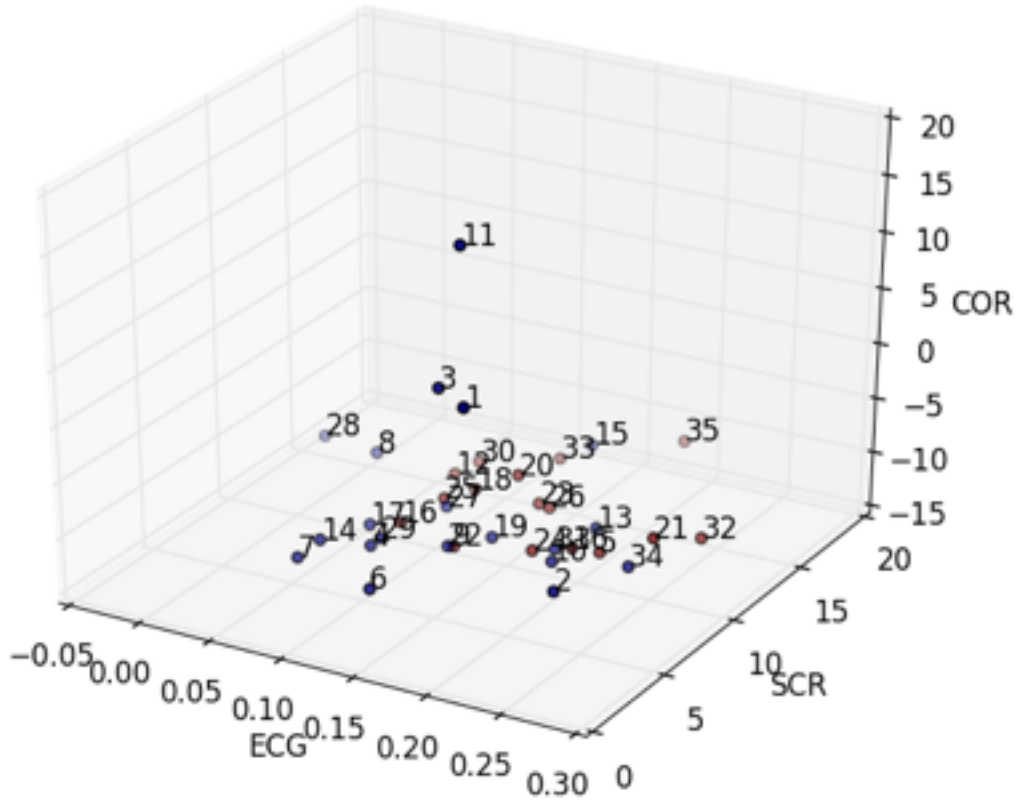
Αξιολόγηση Κατηγοριοποιητών

Η ακρίβεια του κατηγοριοποιητή SVM είναι υψηλή. Αυτό οφείλεται τόσο στην ευαισθησία όσο και στην προσδιοριστικότητα οι οποίες είναι πολύ ψηλές, με περισσότερη έμφαση στην ευαισθησία του κατηγοριοποιητή, η οποία είναι τέλεια και άρα συμπεραίνουμε ότι με μεγάλη πιθανότητα οι άνθρωποι με χαμηλή εμπειρική αποφυγή έχουν κατηγοριοποιηθεί σωστά. Αλλά και η προσδιοριστικότητα είναι πολύ ψηλή και άρα με μεγάλη πιθανότητα οι άνθρωποι με ψηλή εμπειρική αποφυγή έχουν κατηγοριοποιηθεί σωστά.

Η ακρίβεια του κατηγοριοποιητή KNN είναι υψηλή. Αυτό οφείλεται στην ευαισθησία του κατηγοριοποιητή, η οποία είναι τέλεια, κατηγοριοποιώντας ορθά με μεγάλη πιθανότητα όλους τους ανθρώπους με χαμηλή εμπειρική αποφυγή. Αυτό που μειώνει την τιμή της ακρίβειας είναι η προσδιοριστικότητα, η οποία είναι αρκετά χαμηλή, δεν διακρίνει ορθά ασθενείς με ψηλή εμπειρική αποφυγή.

Και οι δύο κατηγοριοποιητές έχουν τέλεια ευαισθησία, αλλά διαφέρουν ως προς την ακρίβεια και την προσδιοριστικότητα. Επομένως, ο καλύτερος από τους δύο κατηγοριοποιητές με συνδυασμό όλων των μέτρων αξιολόγησης είναι ο SVM, ο οποίος προσφέρει τόσο ψηλή ακρίβεια, όσο και ψηλή προσδιοριστικότητα.

6.2 Εφαρμογή Αλγορίθμων μη Επιβλεπόμενης Μάθησης



Σχήμα 6.1 : Στο πιο πάνω τρισδιάστατο σχήμα φαίνεται η πραγματική κατηγοριοποίηση τον δεδομένων. Όπου με μπλε χρώμα φαίνονται οι ασθενείς με χαμηλή εμπειρική αποφυγή και με κόκκινο χρώμα οι ασθενείς με ψηλή εμπειρική αποφυγή. Τα αρχεία των ασθενών ήταν αριθμημένα με μεγάλους αριθμούς. Για χάριν ευκολίας χρησιμοποίησα αριθμούς από το 1 ως το 36, που αντιστοιχούν στην σειρά με την οποία διαβάστηκε κάθε αρχείο. (**Η κλάση 0 για τους**

καπνιστές με χαμηλή εμπειρική αποφυγή (αρνητική κατάσταση) και 1 για τους καπνιστές με ψηλή εμπειρική αποφυγή (θετική κατάσταση)).

Ομαδοποιητής k-means με βάση τον μέσο όρο των μετρήσεων

Από κάθε τεστ υπολογίζεται ο μέσος όρος (average) των μετρήσεων SCR για κάθε ασθενή για τα δύο τελευταία διαστήματα (tasks). Και άρα υπολογίζονται δύο μέσοι όροι για κάθε ασθενή, δηλαδή δύο χαρακτηριστικά. Ο αλγόριθμος kmeans παίρνει ως είσοδο 36 ασθενείς και τους διαχωρίζει σε δύο συστάδες με βάση αυτό το χαρακτηριστικό.

Αποτελέσματα συσταδοποίησης

Με βάση την αρίθμηση των ασθενών στην γραφική παράσταση, οι ασθενείς με χαμηλή εμπειρική αποφυγή είναι οι εξής **[1, 2, 3, 4, 6, 7, 9, 10, 11, 14, 16, 17, 19, 22, 24, 29, 31, 34, 36].** Και οι ασθενείς με ψηλή εμπειρική αποφυγή είναι οι **[5, 8, 12, 13, 15, 18, 20, 21, 23, 25, 26, 27, 28, 30, 32, 33, 35]**.

Αξιολόγηση

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	12	4
αρνητική κατάσταση	5	15

Ακρίβεια : 27 / 36 = 0.75

Ευαισθησία : 12 / 16 = 0.75

Προσδιοριστικότητα : 15 / 20 = 0.75

silhouette_score = 0.3150847571

silhouette_samples :

[0.68518519 0.68518519 0.68518519 0.68518519 0.60416667 0.68518519
0.68518519 -0.71929825 -0.64705882 0.68518519 0.68518519 0.60416667
-0.71929825 0.68518519 -0.71929825 -0.64705882 0.68518519 0.60416667
0.68518519 0.60416667 0.60416667 0.68518519 0.60416667 -0.64705882
0.60416667 0.60416667 -0.71929825 -0.71929825 0.68518519 0.60416667
0.68518519 0.60416667 0.60416667 0.68518519 0.60416667 -0.64705882]

Κάθε αρχείο ασθενή ήταν αριθμημένο με κάποιον αριθμό. Στον πιο κάτω πίνακα φαίνονται οι αριθμοί των ασθενών με ψηλή εμπειρική αποφυγή και οι αριθμοί των ασθενών με χαμηλή εμπειρική αποφυγή.

Χαμηλή εμπειρική αποφυγή	Ψηλή εμπειρική αποφυγή
12200	62205
32202	91208
42203	132212
51204	142213
72206	161215
82207	482247
102209	502249
112210	512250
122211	531252
152214	552254
172216	562255
432242	5722255
492248	581257
522251	622261
542253	641263
592258	651264
631262	442243
671266	
472246	

Ομαδοποιητής k-means με βάση την διάμεσο

Από κάθε τεστ υπολογίζεται η διάμεσος (median) των μετρήσεων SCR του κάθε ασθενή για τα δύο τελευταία διαστήματα (tasks). Και άρα υπολογίζονται δύο διάμεσοι για κάθε ασθενή, δηλαδή δύο χαρακτηριστικά. Ο αλγόριθμος kmeans παίρνει ως είσοδο 36 ασθενείς και τους διαχωρίζει σε δύο συστάδες με βάση αυτά τα χαρακτηριστικά.

Αποτελέσματα συσταδοποίησης

Με βάση την αρίθμηση των ασθενών στην γραφική παράσταση, οι ασθενείς με χαμηλή εμπειρική αποφυγή είναι οι εξής **[1, 2, 3, 4, 6, 7, 9, 10, 11, 14, 16, 17, 19, 22, 24, 29, 31, 34, 36]**. Και οι ασθενείς με ψηλή εμπειρική αποφυγή είναι οι **[5, 8, 12, 13, 15, 18, 20, 21, 23, 25, 26, 27, 28, 30, 32, 33, 35]**.

Αξιολόγηση

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	12	4
αρνητική κατάσταση	5	15

Ακρίβεια : 27 / 36 = 0.75

Ευαισθησία : 12 / 16 = 0.75

Προσδιοριστικότητα : 15 / 20 = 0.75

silhouette_score = 0.3150847571

silhouette_samples :

[0.68518519 0.68518519 0.68518519 0.68518519 0.60416667 0.68518519
  0.68518519 -0.71929825 -0.64705882 0.68518519 0.68518519 0.60416667
 -0.71929825 0.68518519 -0.71929825 -0.64705882 0.68518519 0.60416667
  0.68518519 0.60416667 0.60416667 0.68518519 0.60416667 -0.64705882
  0.60416667 0.60416667 -0.71929825 -0.71929825 0.68518519 0.60416667
  0.68518519 0.60416667 0.60416667 0.68518519 0.60416667 -0.64705882]

Κάθε αρχείο ασθενή ήταν αριθμημένο με κάποιον αριθμό. Στον πιο κάτω πίνακα φαίνονται οι αριθμοί των ασθενών με ψηλή εμπειρική αποφυγή και οι αριθμοί των ασθενών με χαμηλή εμπειρική αποφυγή.

Χαμηλή εμπειρική αποφυγή	Ψηλή εμπειρική αποφυγή
12200	62205
32202	91208
42203	132212
51204	142213
62205	161215
72206	482247
82207	502249
102209	512250
112210	531252
122211	552254
152214	562255

Χαμηλή εμπειρική αποφυγή	Ψηλή εμπειρική αποφυγή
172216	5722255
432242	581257
492248	622261
522251	641263
542253	651264
631262	442243
671266	
472246	

Ομαδοποιητής k-means με βάση το ιστόγραμμα

Για κάθε ασθενή δημιουργούμε ένα histogram με τις μετρήσεις SCR. Ο αλγόριθμος kmeans παίρνει ως είσοδο 36 ασθενείς και τους διαχωρίζει σε δύο συστάδες με βάση αυτά τα histograms.

Αποτελέσματα συσταδοποίησης

Με βάση την αρίθμηση των ασθενών στην γραφική παράσταση, οι ασθενείς με χαμηλή εμπειρική αποφυγή είναι οι εξής **[1, 2, 3, 4, 6, 7, 9, 10, 14, 16, 17, 22, 29].** Και οι ασθενείς με ψηλή εμπειρική αποφυγή είναι οι **[5, 8, 11, 12, 13, 15, 18, 19, 20, 21, 23, 24, 25, 26, 27, 28, 30, 31, 32, 33, 34, 35, 36]**.

Αξιολόγηση

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	14	2
αρνητική κατάσταση	9	11

Ακρίβεια : 25 / 36 = 0.70

Ευαισθησία : 14 / 16 = 0.88

Προσδιοριστικότητα : 11 / 20 = 0.55

silhouette_score = 0.201363314259

silhouette_samples :

[0.72619048 0.72619048 0.72619048 0.72619048 0.51652893 0.72619048
0.72619048 -0.75824176 -0.57312253 0.72619048 -0.75824176 0.51652893
-0.75824176 0.72619048 -0.75824176 -0.57312253 0.72619048 0.51652893
-0.75824176 0.51652893 0.51652893 0.72619048 0.51652893 0.51652893
0.51652893 0.51652893 -0.75824176 -0.75824176 0.72619048 0.51652893
-0.75824176 0.51652893 0.51652893 -0.75824176 0.51652893 0.51652893]

Κάθε αρχείο ασθενή ήταν αριθμημένο με κάποιον αριθμό. Στον πιο κάτω πίνακα φαίνονται οι αριθμοί των ασθενών με ψηλή εμπειρική αποφυγή και οι αριθμοί των ασθενών με χαμηλή εμπειρική αποφυγή.

Χαμηλή εμπειρική αποφυγή	Ψηλή εμπειρική αποφυγή
12200	62205
32202	91208
42203	122211
51204	132212
72206	142213
82207	161215
102209	482247
112210	492248
152214	502249
172216	512250
432242	531252
522251	542253
592258	552254
	562255
	5722255
	581257
	622261
	631262
	641263
	651264
	671266
	442243
	472246

Αξιολόγηση Συσταδοποιητών

Η ακρίβεια του ομαδοποιητή k-means με βάση τον μέσο όρο είναι ικανοποιητική. Αυτό οφείλεται τόσο στην ευαισθησία όσο και στην προσδιοριστικότητα οι οποίες είναι αρκετά ικανοποιητικές. Αυτό σημαίνει ότι οι άνθρωποι με χαμηλή εμπειρική αποφυγή και οι ασθενείς με ψηλή εμπειρική αποφυγή έχουν κατηγοριοποιούνται το ίδιο αποτελεσματικά. Όπως βλέπουμε, η τιμή silhouette_score είναι ενδιάμεσα από το 0 και το +1. Αυτό συμβαίνει γιατί μερικοί ασθενείς είναι στα όρια του να έχουν ψηλή εμπειρική αποφυγή και άρα βρίσκονται ενδιάμεσα στις δύο κλάσεις (βλέπε Σχήμα 6.1).

Η ακρίβεια του ομαδοποιητή k-means με βάση την διάμεσο είναι ικανοποιητική. Αυτό οφείλεται τόσο στην ευαισθησία όσο και στην προσδιοριστικότητα οι οποίες είναι αρκετά ικανοποιητικές. Αυτό σημαίνει ότι οι άνθρωποι με χαμηλή εμπειρική αποφυγή και οι ασθενείς με ψηλή εμπειρική αποφυγή έχουν κατηγοριοποιούνται το ίδιο αποτελεσματικά. Όπως βλέπουμε, η τιμή silhouette_score είναι ενδιάμεσα από το 0 και το +1, το οποίο, όπως αναφέραμε και πιο πάνω, δείχνει ότι οι κλάσεις επικαλύπτονται. Αυτό συμβαίνει γιατί μερικοί ασθενείς είναι στα όρια του να έχουν ψηλή εμπειρική αποφυγή και άρα βρίσκονται ενδιάμεσα στις δύο κλάσεις (βλέπε Σχήμα 6.1).

Η ακρίβεια του ομαδοποιητή k-means με βάση το ιστόγραμμα είναι μέτρια. Αυτό οφείλεται στο ότι η ευαισθησία είναι ψηλή, άρα συμπεραίνουμε ότι με μεγάλη πιθανότητα κατηγοριοποιεί ορθά τους ανθρώπους με χαμηλή εμπειρική αποφυγή. Αυτό που μειώνει την τιμή της ακρίβειας είναι η προσδιοριστικότητα, η οποία είναι αρκετά χαμηλή, συμπεραίνοντας ότι δεν διακρίνει ορθά ασθενείς με ψηλή εμπειρική αποφυγή. Όπως βλέπουμε, η τιμή silhouette_score είναι ενδιάμεσα από το 0 και το +1, το οποίο, όπως αναφέραμε και πιο πάνω, δείχνει ότι οι συστάδες

επικαλύπτονται. Αυτό συμβαίνει γιατί μερικοί ασθενείς είναι στα όρια του να έχουν ψηλή εμπειρική αποφυγή και άρα βρίσκονται ενδιάμεσα στις δύο κλάσεις (βλέπε Σχήμα 6.1).

Οι δύο πρώτοι ομαδοποιητές έχουν τις ίδιες τιμές σε όλες τις μετρικές. Ο τρίτος ομαδοποιητής έχει πιο χαμηλή ακρίβεια από τους δύο πρώτους αλλά αυτό οφείλεται στην πολύ χαμηλή προσδιοριστικότητα. Παρά την πολύ χαμηλή τους προσδιοριστικότητα έχει πιο ψηλή ευαισθησία και πιο χαμηλό silhouette_score. Ως καλύτερο ομαδοποιητή επιλέγουμε τους πρώτους δύο, επειδή έχουν ικανοποιητικές τιμές για όλες τις μετρικές, ενώ ο τρίτος έχει πολύ χαμηλή προσδιοριστικότητα.

Κεφάλαιο 7

Μελέτη Περίπτωσης 2 : Διάγνωση Διατροφικών Διαταραχών

Σε αυτό το κεφάλαιο θα εφαρμοστούν αλγόριθμοι επιβλεπόμενης και μη επιβλεπόμενης μάθησης σε ένα σύνολο δεδομένων για την εύρεση ανθρώπων με διατροφικές διαταραχές. Τα δεδομένα αυτά δημιουργήθηκαν από το τμήμα Ψυχολογίας του Πανεπιστημίου. Το σύνολο των δεδομένων αποτελείτο από 71 δείγματα ασθενών. Για κάθε ασθενή πάρθηκε ένας όγκος μετρήσεων από τα σήματα ECG, SCR και COR. Εμείς, μετά από κάποια πειράματα, αποφασίσαμε να χρησιμοποιήσουμε μόνο τις μετρήσεις του δεύτερου σήματος, ECG. Έγιναν αρκετοί συνδυασμοί μέχρι να καταλήξουμε σε αυτή την απόφαση. Δηλαδή πρώτα αξιολογήθηκε η απόδοση του αλγορίθμου με την χρήση κάθε σήματος ξεχωριστά και μετά έγιναν συνδυασμοί αυτών των σημάτων. Οι μετρήσεις δεν πάρθηκαν σε ένα συνεχόμενο διάστημα, αλλά, κατά την διάρκεια πέντε διαφορετικών διαστημάτων (tasks). Για τον λόγο αυτό ανάμεσα σε κάθε δύο χρονικά διαστήματα, δηλαδή από το τέλος του ενός μέχρι την έναρξη του άλλου, υπήρχαν ανεπιθύμητα σήματα, τα οποία και αφαιρέσαμε. Ο στόχος ύπαρξης του πρώτου διαστήματος ήταν απλά για να χαλαρώσει ο ασθενής. Στα επόμενα τέσσερα διαστήματα ο ασθενής κλήθηκε να δει κάποιες ταινίες μικρού μήκους. Ο στόχος των δύο πρώτων ταινιών ήταν να

δημιουργήσουν στον ασθενή ουδέτερο συναίσθημα, ενώ των υπόλοιπων δύο ήταν να του δημιουργήσουν αρνητικό συναίσθημα. Η χρονική διάρκεια κάθε διαστήματος ήταν 2 λεπτά και 60 δευτερόλεπτα. Στόχος των αλγορίθμων είναι η κατηγοριοποίηση-ομαδοποίηση των διανυσμάτων σε δύο κλάσης, **την κλάση 0 για τους ασθενείς με ψηλό ρίσκο (θετική κατάσταση) και 1 για τους ασθενείς με χαμηλό ρίσκο (αρνητική κατάσταση).** Το Τμήμα Ψυχολογίας μου παρείχε την πραγματική κλάση στην οποία ανήκει ο κάθε ασθενής, δηλαδή ποιοι από τους ασθενείς έχουν ψηλό ρίσκο ως προς τις διατροφικές διαταραχές και ποιοι έχουν χαμηλό ρίσκο. Έτσι οι κλάσεις που θα προβλέψει ο κάθε αλγόριθμους για κάθε ασθενή θα συγκριθούν, με την χρήση κάποιων μετρικών αξιολόγησης, με τις πραγματικές κλάσεις, έτσι ώστε να μετρηθεί η απόδοση του αλγορίθμου.

7.1 Εφαρμογή αλγορίθμων επιβλεπόμενης μάθησης

Οι πιο κάτω αλγόριθμοι επιβλεπόμενης μάθησης παίρνουν ως είσοδο ένα σύνολο με δείγματα εκπαίδευσης (training data set) με 57 ασθενείς και τις κλάσεις (labels) στις οποίες ανήκουν. Μετά από την εκπαίδευση του κατηγοριοποιητή, του δίνουμε ένα σύνολο με δείγματα δοκιμής (test data set) με 14 ασθενείς για να δούμε την ακρίβεια, την ευαισθησία και τη προσδιοριστικότητα, με την οποία θα προβλέψει τις κλάσεις των δειγμάτων δοκιμής. Χρησιμοποιήσαμε ως ορθές κλάσεις για κάθε δείγμα ασθενή αυτές που μας δόθηκαν από το Τμήμα Ψυχολογίας.

Κατηγοροποιητής SVM

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	8	1
αρνητική κατάσταση	0	5

Ακρίβεια : 13 / 14 = 0.93

Ευαισθησία : 8 / 9 = 0.89

Προσδιοριστικότητα : 5 / 5 = 1

Κατηγοροποιητής KNN (k = 3)

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	6	3
αρνητική κατάσταση	0	5

Ακρίβεια : 11 / 14 = 0.79

Ευαισθησία : 6 / 9 = 0.67

Προσδιοριστικότητα : 5 / 5 = 1

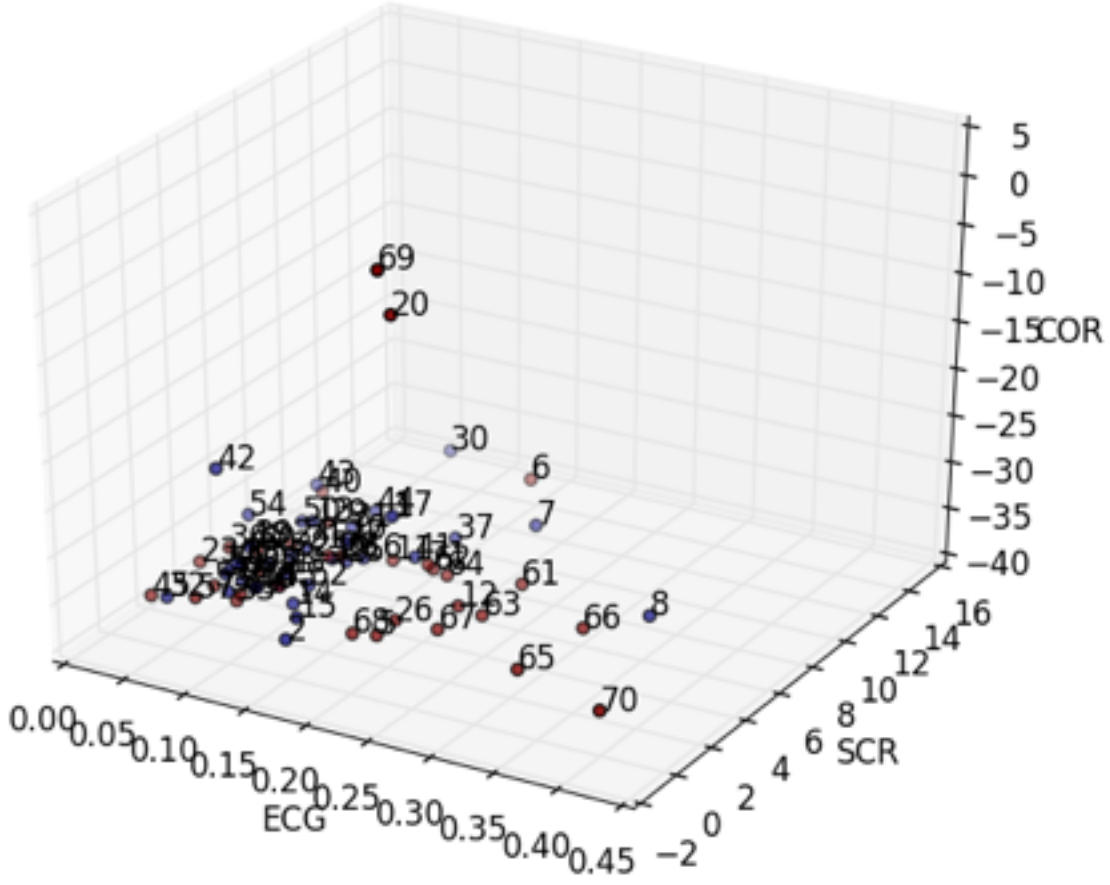
Αξιολόγηση Κατηγοριοποιητών

Όλα τα μέτρα αξιολόγησης του κατηγοριοποιητή SVM είνα ψηλά. Το μόνο που φέρνει μια ελάχιστη μείωση της ακρίβειας είναι η ευαισθησία.

Η ακρίβεια του κατηγοριοποιητή KNN είναι αρκετά καλή. Αυτό οφείλεται στην προσδιοριστικότητα του κατηγοριοποιητή, η οποία είναι τέλεια, κατηγοριοποιώντας ορθά με μεγάλη πιθανότητα τους ανθρώπους με ψηλό ρίσκο. Αυτό που μειώνει την τιμή της ακρίβειας είναι η ευαισθησία, η οποία είναι ελάχιστα πιο χαμηλή, συμπεραίνοντας ότι διακρίνει λιγότερο ορθά ασθενείς με χαμηλό ρίσκο.

Και οι δύο κατηγοριοποιητές έχουν τέλεια προσδιοριστικότητα, αλλά διαφέρουν ως προς την ακρίβεια και την ευαισθησία. Επομένως, ο καλύτερος από τους δύο κατηγοριοποιητές με συνδυασμό όλων των μέτρων αξιολόγησης είναι ο SVM, ο οποίος προσφέρει τόσο ψηλή απόδοση, όσο και ψηλή ευαισθησία.

7.2 Εφαρμογή αλγορίθμων μη επιβλεπόμενης μάθησης



Σχήμα 7.1 : Στο πιο πάνω τρισδιάστατο σχήμα φαίνεται η πραγματική κατηγοριοποίηση τον δεδομένων. Όπου με μπλε χρώμα φαίνονται οι ασθενείς με χαμηλό ρίσκο και με κόκκινο χρώμα οι ασθενείς με ψηλό ρίσκο. Τα αρχεία των ασθενών ήταν αριθμημένα με μεγάλους αριθμούς. Για

χάριν ευκολίας χρησιμοποίησα αριθμούς από το 1 ως το 71,που αντιστοιχούν στην σειρά με την οποία διαβάστηκε κάθε αρχείο.

Ομαδοποιητής k-means με βάση τον μέσο όρο των μετρήσεων

Από κάθε τεστ υπολογίζονται οι μέσοι όροι (average) των μετρήσεων ECG για κάθε ασθενή για το πρώτο task, για το δεύτερο μαζί με το τρίτο και για το τέταρτο μαζί με το πέμπτο. Ο αλγόριθμος kmeans παίρνει ως είσοδο 71 ασθενείς και τους διαχωρίζει σε δύο συστάδες με βάση αυτά τα χαρακτηριστικά.

Αποτελέσματα συσταδοποίησης

Με βάση την αρίθμηση των ασθενών στην γραφική παράσταση, οι ασθενείς με ψηλό ρίσκο είναι οι εξής **[1, 2, 3, 4, 9, 10, 11, 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 24, 25, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 38, 39, 40, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60].** Και οι ασθενείς με χαμηλό ρίσκο είναι οι **[5, 6, 7, 8, 12, 20, 26, 37, 41, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71]**.

Αξιολόγηση

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	35	4
αρνητική κατάσταση	16	16

Ακρίβεια : 51 / 71 = 0.72

Ευαισθησία : 35 / 39 = 0.90

Προσδιοριστικότητα : 16 / 32 = 0.50

silhouette_score = 0.255681041876

silhouette_samples :

[0.6 0.6 0.6 0.6 0.69323308 0.69323308
-0.62745098 -0.62745098 0.6 0.6 -0.71428571 0.69323308
0.6 0.6 0.6 0.6 0.6 0.6 0.6
0.69323308 0.6 0.6 -0.71428571 -0.71428571 0.6
0.69323308 0.6 -0.71428571 -0.71428571 0.6 0.6 0.6
-0.71428571 0.6 -0.71428571 -0.71428571 -0.62745098 0.6 0.6
-0.71428571 -0.62745098 0.6 0.6 0.6 -0.71428571
0.6 0.6 0.6 -0.71428571 0.6 0.6 0.6
-0.71428571 0.6 -0.71428571 -0.71428571 -0.71428571 0.6 0.6
-0.71428571 0.69323308 0.69323308 0.69323308 0.69323308 0.69323308
0.69323308 0.69323308 0.69323308 0.69323308 0.69323308 0.69323308]

Κάθε αρχείο ασθενή ήταν αριθμημένο με κάποιον αριθμό. Στον πιο κάτω πίνακα φαίνονται οι αριθμοί των ασθενών με ψηλό ρίσκο και οι αριθμοί των ασθενών με χαμηλό ρίσκο.

Ψηλό ρίσκο	Χαμηλό ρίσκο
425	62205
35	91208
39	122211
40	132212
336	142213
338	161215
339	482247
358	492248
395	502249
397	512250
413	531252
453	542253
460	552254
598	562255
1002	5722255
1004	581257
1009	622261
1021	631262
1027	641263

Ψηλό ρίσκο	Χαμηλό ρίσκο
1029	651264
1051	43
1053	135
1060	150
1063	155
1068	344
1071	603
1078	1028
2002	2014
2012	2022
2018	3052
2019	3054
2021	3061
2023	3062
2027	3063
2033	3065
2040	3066
2066	3067
2078	3068
2102	3073
2109	4001
2117	

Ψηλό ρίσκο	Χαμηλό ρίσκο
2118	
2156	
2192	
2207	
2212	
2239	
2243	
2248	
2261	
2264	

Ομαδοποιητής k-means με βάση την διάμεσο των μετρήσεων

Από κάθε τεστ υπολογίζονται οι διάμεσοι (median) των μετρήσεων ECG για κάθε ασθενή για το πρώτο task, για το δεύτερο μαζί με το τρίτο και για το τέταρτο μαζί με το πέμπτο. Ο αλγόριθμος kmeans παίρνει ως είσοδο 71 ασθενείς και τους διαχωρίζει σε δύο συστάδες με βάση αυτά τα χαρακτηριστικά.

Αποτελέσματα συσταδοποίησης

Με βάση την αρίθμηση των ασθενών στην γραφική παράσταση, οι ασθενείς με ψηλό ρίσκο είναι οι εξής **[1, 2, 3, 4, 6, 7, 9, 10, 11, 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 24, 25, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57,**

58, 59, 60]. Και οι ασθενείς με χαμηλό ρίσκο είναι οι **[5, 8, 12, 20, 26, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71]**.

Αξιολόγηση

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	38	1
αρνητική κατάσταση	17	15

Ακρίβεια : 53 / 71 = 0.75

Ευαισθησία : 38 / 39 = 0.97

Προσδιοριστικότητα : 15 / 32 = 0.47

silhouette_score = 0.318466156084

silhouette_samples :

[0.66419753 0.66419753 0.66419753 0.66419753 0.90350877 -0.91118421
0.66419753 -0.69090909 0.66419753 0.66419753 -0.91118421 0.90350877
0.66419753 0.66419753 0.66419753 0.66419753 0.66419753 0.66419753
0.66419753 0.90350877 0.66419753 0.66419753 -0.91118421 -0.91118421
0.66419753 0.90350877 0.66419753 -0.91118421 -0.91118421 0.66419753
0.66419753 0.66419753 -0.91118421 0.66419753 -0.91118421 -0.91118421
0.66419753 0.66419753 0.66419753 -0.91118421 0.66419753 0.66419753
0.66419753 0.66419753 -0.91118421 0.66419753 0.66419753 0.66419753

-0.91118421 0.66419753 0.66419753 0.66419753 -0.91118421 0.66419753
-0.91118421 -0.91118421 -0.91118421 0.66419753 0.66419753 -0.91118421
 0.90350877 0.90350877 0.90350877 0.90350877 0.90350877 0.90350877
 0.90350877 0.90350877 0.90350877 0.90350877 0.90350877]

Κάθε αρχείο ασθενή ήταν αριθμημένο με κάποιον αριθμό. Στον πιο κάτω πίνακα φαίνονται οι αριθμοί των ασθενών με ψηλό ρίσκο και οι αριθμοί των ασθενών με χαμηλό ρίσκο.

Ψηλό ρίσκο	Χαμηλό ρίσκο
425	43
35	155
39	344
40	603
135	1028
150	3052
336	3054
338	3061
339	3062
358	3063
395	3065
397	3066
413	3067
453	3068
460	3073

Ψηλό ρίσκο	Χαμηλό ρίσκο
598	4001
1002	
1004	
1009	
1021	
1027	
1029	
1051	
1053	
1060	
1063	
1068	
1071	
1078	
2002	
2012	
2014	
2018	
2019	
2021	
2022	
2023	

Ψηλό ρίσκο	Χαμηλό ρίσκο
2027	
2033	
2040	
2066	
2078	
2102	
2109	
2117	
2118	
2156	
2192	
2207	
2212	
2239	
2243	
2248	
2261	
2264	

Ομαδοποιητής k-means με βάση το ιστόγραμμα

Για κάθε ασθενή δημιουργούμε ένα ιστόγραμμα (histogram) με τις μετρήσεις ECG από τα δύο τελευταία task. Ο αλγόριθμος kmeans παίρνει ως είσοδο 71 ασθενείς και τους διαχωρίζει σε δύο συστάδες με βάση αυτά τα histograms.

Αποτελέσματα συσταδοποίησης

Με βάση την αρίθμηση των ασθενών στην γραφική παράσταση, οι ασθενείς με χαμηλή εμπειρική αποφυγή είναι οι εξής **[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 24, 25, 27, 28, 29, 30, 31, 33, 34, 35, 36, 37, 38, 40, 41, 42, 44, 46, 48, 50, 52, 53, 55, 56, 58, 59, 61, 63, 64, 65, 69, 71].** Και οι ασθενείς με ψηλή εμπειρική αποφυγή είναι οι **[23, 26, 32, 39, 43, 45, 47, 49, 51, 54, 57, 60, 62, 66, 67, 68, 70].**

Αξιολόγηση

Πίνακας Σύγχυσης

	προβλεπόμενη θετική κατάσταση	προβλεπόμενη αρνητική κατάσταση
θετική κατάσταση	33	6
αρνητική κατάσταση	21	11

Ακρίβεια : 44 / 71 = 0.62

Ευαισθησία : 33 / 39 = 0.85

Προσδιοριστικότητα : 11 / 32 = 0.34

silhouette_score = 0.0752131766146

silhouette_samples :

[0.38765009 0.38765009 0.38765009 0.38765009 -0.43315508 -0.43315508
  0.38765009 0.38765009 0.38765009 0.38765009 -0.43315508 -0.43315508
  0.38765009 0.38765009 0.38765009 0.38765009 0.38765009 0.38765009
  0.38765009 -0.43315508 0.38765009 0.38765009 0.38636364 -0.43315508
  0.38765009 0.38636364 0.38765009 -0.43315508 -0.43315508 0.38765009
  0.38765009 -0.43434343 -0.43315508 0.38765009 -0.43315508 -0.43315508
  0.38765009 0.38765009 -0.43434343 -0.43315508 0.38765009 0.38765009
 -0.43434343 0.38765009 0.38636364 0.38765009 -0.43434343 0.38765009
  0.38636364 0.38765009 -0.43434343 0.38765009 -0.43315508 -0.43434343
 -0.43315508 -0.43315508 0.38636364 0.38765009 0.38765009 0.38636364
 -0.43315508 0.38636364 -0.43315508 -0.43315508 -0.43315508 0.38636364
  0.38636364 0.38636364 -0.43315508 0.38636364 -0.43315508]

Κάθε αρχείο ασθενή ήταν αριθμημένο με κάποιον αριθμό. Στον πιο κάτω πίνακα φαίνονται οι αριθμοί των ασθενών με ψηλό ρίσκο και οι αριθμοί των ασθενών με χαμηλό ρίσκο.

Ψηλό ρίσκο	Χαμηλό ρίσκο
425	1009
35	1028
39	1068
40	2019
43	
135	
150	

Ψηλό ρίσκο	Χαμηλό ρίσκο
155	
336	
338	
339	
344	
358	
395	
397	
413	
453	
460	
598	
603	
1002	
1004	
1021	
1027	
1029	
1051	
1053	
1060	
1063	

Ψηλό ρίσκο	Χαμηλό ρίσκο
1071	
1078	
2002	
2012	
2014	
2018	
2021	
2022	
2023	
2033	
2066	
2102	
2117	
2156	
2192	
2212	
2239	
2248	
2261	
3052	
3061	
3062	

Ψηλό ρίσκο	Χαμηλό ρίσκο
3063	
3068	
4001	

Αξιολόγηση Συσταδοποιητών

Η ακρίβεια του ομαδοποιητή k-means με βάση το μέσο όρο είναι ικανοποιητική. Αυτό οφείλεται στην ευαισθησία του ομαδοποιητή, η οποία είναι ψηλή, κατηγοριοποιώντας ορθά με μεγάλη πιθανότητα τους ανθρώπους με χαμηλό ρίσκο. Αυτό που μειώνει την τιμή της ακρίβειας είναι η προσδιοριστικότητα, η οποία είναι πολύ χαμηλή, συμπεραίνοντας ότι δεν διακρίνει ορθά ασθενείς με ψηλό ρίσκο. Όπως βλέπουμε, η τιμή silhouette_score είναι ενδιάμεσα από το 0 και το +1. Αυτό συμβαίνει γιατί μερικοί ασθενείς είναι στα όρια του να έχουν ψηλό ρίσκο και άρα βρίσκονται ενδιάμεσα στις δύο κλάσεις (βλέπε Σχήμα 7.1).

Η ακρίβεια του ομαδοποιητή k-means με βάση τη διάμεσο είναι ικανοποιητική. Αυτό οφείλεται στην ευαισθησία του ομαδοποιητή, η οποία είναι ψηλή, κατηγοριοποιώντας ορθά με μεγάλη πιθανότητα τους ανθρώπους με χαμηλό ρίσκο. Αυτό που μειώνει την τιμή της ακρίβειας είναι η προσδιοριστικότητα, η οποία είναι πολύ χαμηλή, συμπεραίνοντας ότι δεν διακρίνει ορθά ασθενείς με ψηλό ρίσκο. Όπως βλέπουμε, η τιμή silhouette_score είναι ενδιάμεσα από το 0 και το +1. Αυτό συμβαίνει γιατί μερικοί ασθενείς είναι στα όρια του να έχουν ψηλό ρίσκο και άρα βρίσκονται ενδιάμεσα στις δύο κλάσεις (βλέπε Σχήμα 7.1).

Η ακρίβεια του ομαδοποιητή k-means με βάση το ιστόγραμμα είναι αρκετά χαμηλή. Αυτό οφείλεται στο ότι η προσδιοριστικότητα είναι πολύ χαμηλή, συμπεραίνοντας ότι δεν διακρίνει ορθά ασθενείς με ψηλό ρίσκο. Όπως βλέπουμε, η τιμή silhouette_score είναι κοντά στο 0,το οποίο δείχνει ότι οι συστάδες επικαλύπτονται. Αυτό συμβαίνει γιατί μερικοί ασθενείς είναι στα όρια του να έχουν ψηλό ρίσκο και άρα βρίσκονται ενδιάμεσα στις δύο κλάσεις (βλέπε Σχήμα 7.1).

Ο δεύτερος ομαδοποιητής έχει καλύτερη ακρίβεια και ευαισθησία από τους υπόλοιπους δύο. Από την άλλη ο πρώτος έχει καλύτερη προσδιοριστικότητα και silhouette_score από τους άλλους δύο. Λόγω του ότι και οι τρεις ομαδοποιητές δεν έχουν καθόλου καλή προσδιοριστικότητα, θα επιλέξουμε ως καλύτερο κατηγοριοποιητή αυτόν με την καλύτερη ευαισθησία, δηλαδή τον δεύτερο.

Κεφάλαιο 8

Συμπεράσματα

8.1 Συμπεράσματα

Στην παρούσα εργασία αρχικά μελετήθηκαν θεωρητικά αλγόριθμοι επιβλεπόμενης και μη επιβλεπόμενης μάθησης. Στην συνέχεια εφαρμόσαμε κάποιους από αυτούς τους αλγορίθμους σε πραγματικά δεδομένα. Κατά την αξιολόγηση των αλγορίθμων στα δύο διαφορετικά σύνολα δεδομένων που εξετάσαμε, καταλήξαμε στο συμπέρασμα ότι δεν μπορεί να θεωρηθεί κάποιος αλγόριθμος καλύτερος από όλους, αφού η απόδοση του εξαρτάται από τα ίδια τα δεδομένα. Δηλαδή παίζει ρόλο η συνοχή των δεδομένων και κατά πόσο οι δύο κλάσεις επικαλύπτονται, δηλαδή αν κάποια δείγματα βρίσκονται στο όριο των δύο κλάσεων. Με βάση τα κριτήρια που χρησιμοποιήσαμε παρατηρήσαμε ότι για την απόδοση ενός αλγορίθμου δεν παίζει ρόλο μόνο η ακρίβειά του, αλλά πιο σημαντικό ρόλο παίζουν οι παράγοντες που επηρεάζουν την ακρίβεια, δηλαδή η ευαισθησία και η προσδιοριστικότητά του. Όσο μεγαλύτερα είναι τα δύο τελευταία κριτήρια τόσο πιο πολύ μεγάλη είναι η πιθανότητα το πρότυπο να ανήκει στην σωστή κλάση.

Με βάση αυτά τα κριτήρια αξιολόγησης ο καλύτερος κατηγοριοποιητής επιβλεπόμενης μάθησης που εφαρμόστηκε και στις δύο μελέτες περίπτωσης ήταν ο SVM. Ο KNN

κατηγοριοποιητής είχε ικανοποιητικές τιμές και στις δύο περιπτώσεις με την μια από τις δύο κλάσεις να κατηγοριοποιείται τέλεια.

Για τον λόγο ότι δεν ήμασταν σίγουροι για την ορθότητα των κλάσεων στις οποίες χωρίστηκαν τα δεδομένα από το Τμήμα Ψυχολογίας, εφαρμόσαμε και τον κατηγοριοποιητή μη επιβλεπόμενης μάθησης k-means. Μετασχηματίσαμε τα δεδομένα με διαφορετικούς τρόπους ώστε να τα περάσουμε αυτά ως χαρακτηριστικά στον αλγόριθμο. Οι πρώτοι δύο τρόποι που ήταν να λάβουμε κάποια στατιστικά στοιχεία από τα δεδομένα, ήταν και οι πιο ικανοποιητικοί. Ενώ το να λάβουμε κάποιο ιστόγραμμα των τιμών, δεν έφερε τόσο ικανοποιητικά αποτελέσματα.

8.2 Προβλήματα

Το κύριο πρόβλημα που αντιμετώπισα κατά την εφαρμογή των αλγορίθμου ήταν ότι για κάθε χαρακτηριστικό κάθε ασθενή υπήρχε ένας μεγάλος αριθμός μετρήσεων. Έτσι ήταν αδύνατον να εξάγουμε κάποια γνώση από αυτό το μεγάλο όγκο δεδομένων. Αυτός ο αριθμός μετρήσεων διέφερε σε κάθε ασθενή, αν και το χρονικό διάστημα που πάρθηκαν οι μετρήσεις ήταν το ίδιο για κάθε ασθενή. Αυτό συνέβη για τον λόγο ότι η επιτάχυνση με την οποία πάρθηκαν οι μετρήσεις ήταν διαφορετική σε κάθε ασθενή. Η προσέγγιση που χρησιμοποίησα για τους κατηγοριοποιητές ήταν για κάθε χρονικό διάστημα να κόψω τις μετρήσεις των ασθενών στον ελάχιστο αριθμό μετρήσεων που υπήρχαν στο συγκεκριμένο χρονικό διάστημα. Ενώ η προσέγγιση που ακολούθησα για τον αλγόριθμο k-means ήταν να εξάγω κάποια στατιστικά ή άλλα χαρακτηριστικά από τις μετρήσεις, ώστε να συνοψίσω όλες τις τιμές ενός χαρακτηριστικού σε μία (μέσος όρος, διάμεσος, ιστόγραμμα). Ακόμα να αναφέρω ότι πριν να επιλέξω τελικά αυτά τα δύο στατιστικά, δοκίμασα και άλλα χωρίς όμως κάποιο να καταλήξω σε κάποιο καλύτερο αποτέλεσμα.

8.3 Μελλοντικές Επεκτάσεις

Τα φυσιολογικά σήματα είναι δεδομένα χρονοσειρών (time series data), δηλαδή είναι μια ακολουθία διανυσμάτων τα οποία εξαρτώνται από το χρόνο. Οι μέθοδοι Μηχανικής Μάθησης που εφαρμόσαμε στις δύο μελέτες περίπτωσης, υποθέτουν ότι τα δεδομένα είναι ανεξάρτητα. Άρα ως επέκταση της παρούσας εργασίας μπορούμε να εφαρμόσουμε άλλες μεθόδους που να λαμβάνουν υπόψη τους την συσχέτιση των δεδομένων. Ένα παράδειγμα αλγορίθμου που το πετυχαίνει αυτό είναι ο Dynamic Time Warping (DTW), ο οποίος μετρά την ομοιότητα μεταξύ δύο χρονοσειρών, ακόμα και αν αυτές διαφέρουν είτε ως προς το χρόνο είτε ως προς την ταχύτητα [76]. Για παράδειγμα, και στις δύο μελέτες περίπτωσης, πάρθηκαν μετρήσεις σημάτων για το ίδιο χρονικό διάστημα από όλους τους ασθενείς, εν τούτοις ο αριθμός των μετρήσεων διέφερε για κάθε ασθενή.

Βιβλιογραφία

[1] Arel, I., Rose, D. C., & Karnowski, T. P. (2010). Deep Machine Learning - A New Frontier in Artificial Intelligence Research [Research Frontier]. IEEE Computational Intelligence Magazine, 5(4), 13–18. http://doi.org/10.1109/MCI.2010.938364

[2] Deng, L., & Yu, D. (2013). Deep Learning: Methods and Aopplications. Foundations and Trends in Signal Processing, 7(3-4), 197–387. http://doi.org/10.1136/bmj.319.7209.0a

[3] Guerra, E., de Lara, J., Malizia, A., & Díaz, P. (2009). Supporting user-oriented analysis for multi-view domain-specific visual languages. Information and Software Technology. http://doi.org/10.1016/j.infsof.2008.09.005

[3] http://deeplearning.net/reading-list/, (last accepted September 2015)

[4] http://www.iro.umontreal.ca/~bengioy/dlbook/intro.html, (last acceptance September 2015)

[5] https://books.google.com.cy/books?hl=el&lr=&id=f44hLefOz6UC&oi=fnd&pg=PT4&dq=deep+learning+EEG+brain+signals&ots=Fsh0wAdTA1&sig=Sd8kAacKlkJKOUmbs77XOJR6P3E&redir_esc=y#v=onepage&q&f=false, (last acceptance September 2015)

[6] https://en.wikipedia.org/wiki/Electrodermal_activity, (last acceptance September 2015)

[7] http://www.megaemg.com/knowledge/heart-rate-variability-hrv/, (last acceptance September 2015)

[8] https://en.wikipedia.org/wiki/Electrocardiography, (last acceptance September 2015)

[9] Alomari, M., Samaha, A., & AlKamha, K. (2013). Automated Classification of L/R Hand Movement EEG Signals using Advanced Feature Extraction and Machine Learning. arXiv Preprint arXiv:1312.2877, 4(6), 207–212. http://doi.org/10.14569/IJACSA.2013.040628

[10] Bartaula, Jyoti Prasad, M. M. A. T. B. (2013). Emotion Recognition from EEG Signals using Machine Learning Mohammadshakib Moshfeghi Aliye Tuke Bedasso, (March).

[11] Blankertz, B., Dornhege, G., Krauledat, M., Kunzmann, V., Losch, F., Curio, G., & Müller, K.-R. (2007). The Berlin brain-computer interface: Machine-learning based detection of user specific brain states, 12(6), 581–607. Retrieved from http://eprints.pascal-network.org/archive/00003320/

[12] Jirayucharoensak, S., & Israsena, P. (2014). EEG-based Emotion Recognition using Deep Learning Network with Principal Component based Covariate Shift Adaptation, 2014. http://doi.org/10.1155/2014/627892

[13] Larsen, E. A., & Wang, A. I. (2011). Classification of EEG Signals in a Brain- Computer Interface System. Norwegian University of Science and Technology, (June), 1–72.

[14] Makeig, S., Kothe, C., Mullen, T., Bigdely-Shamlo, N., Zhang, Z., & Kreutz-Delgado, K. (2012). Evolving signal processing for brain-computer interfaces. Proceedings of the IEEE, 100(SPL CONTENT), 1567–1584. http://doi.org/10.1109/JPROC.2012.2185009

[15] Mikhail, M. (2008). Real Time Emotion Detection using EEG Mina Mikhail Computer Science and Engineering Department The American University in Cairo Seminar. Seminar.

[16] Nu, C. S., & Wellman, R. (n.d.). An exploratory investigation using electroencephalography and machine learning techniques for fine motor classification in the EggLink brain-computer interface . Data Transformations (MATLAB) Machine Learning Algorithms (MATLAB) Support-Vector Machine Optimizers :, 8.

[17] Podgorelec, V. (2012). Analyzing EEG signals with machine learning for diagnosing alzheimer's disease. Elektronika Ir Elektrotechnika, 18(8), 61–64. http://doi.org/10.5755/j01.eee.18.8.2627

[18] PROJECT PROFILE: Automatic Generation of EEG Reports Using Deep Learning. (n.d.).

[19] Sanei, S., & Chambers, J. a. (2007). EEG Signal Processing. Chemistry & biodiversity (Vol. 1). http://doi.org/10.1002/9780470511923

[20] Shoeb, A. H., & Guttag, J. V. (2010). Application of machine learning to epileptic seizure detection. Proceedings of the International Conference on Machine Learning (ICML), 975–982. Retrieved from http://machinelearning.wustl.edu/mlpapers/paper_files/icml2010_ShoebG10.pdf

[21] Walter, C., Cierniak, G., & Gerjets, P. (2011). Classifying mental states with machine learning algorithms using alpha activity decline. ESANN 2011 Proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, (April), 27–29. Retrieved from http://material.bccn.uni-freiburg.de/publications-bccn/2011/Walter11_405.pdf

[22] Wulsin, D. F., Gupta, J. R., Mani, R., Blanco, J. a, & Litt, B. (2011). Modeling electroencephalography waveforms with semi-supervised deep belief nets: fast classification and anomaly measurement. Journal of Neural Engineering, 8(3), 036015. http://doi.org/10.1088/1741-2560/8/3/036015

[23] Zaremba, W. (n.d.). University of Warsaw Modeling the variability of EEG / MEG data through statistical machine learning, (September 2012).

[24] Alomari, M. H., Abubaker, A., Turani, A., Baniyounes, A. M., & Manasreh, A. (2014). EEG Mouse : A Machine Learning-Based Brain Computer Interface, 5(4), 193–198.

[24] Ayzenberg, Y., & Picard, R. W. (2014). FEEL: A system for frequent event and electrodermal activity labeling. IEEE Journal of Biomedical and Health Informatics, 18(1), 266–277. http://doi.org/10.1109/JBHI.2013.2278213

[25] Benefits, W. I. G. I. B., For, A., & Bill, W. I. G. I. (n.d.). Frequently Asked Questions, 1–9.

[26] Friedman, D., Suji, K., & Slater, M. (2007). SuperDreamCity: An Immersive Virtual Reality Experience that Responds to Electrodermal Activity, 1–12. http://doi.org/10.1007/978-3-540-74889-2_50

[27] Henriques, R., Paiva, A., & Antunes, C. (2013). Accessing emotion patterns from affective interactions using electrodermal activity. Proceedings - 2013 Humaine Association Conference on Affective Computing and Intelligent Interaction, ACII 2013, 43–48. http://doi.org/10.1109/ACII.2013.14

[28] https://en.wikipedia.org/wiki/Supervised_learning, (last acceptance December 2015)

[29] Ieee, N. J. I., Hedman, E., Wilder-smith, O., Goodwin, M. S., & Picard, R. (2014). iCalm : Measuring electrodermal activity in almost any setting The MIT Faculty has made this article openly available . Please share Citation activity in almost any setting . 3rd International Conference on and may be subject to US copyright law . Please .

[30] Kurniawan, H., Maslov, A. V., & Pechenizkiy, M. (2013). Stress detection from speech and galvanic skin response signals. Proceedings of CBMS 2013 - 26th IEEE International Symposium on Computer-Based Medical Systems, 209–214. http://doi.org/10.1109/CBMS.2013.6627790

[31] Peuscher, J. (2012). Galvanic skin response (GSR), (November). Retrieved from http://www.tmsi.com/products/accessories?task=callelement&format=raw&item_id=43&element=fe0c95f3-af08-4719-bc51-36917715660d&method=download

[32] Poh, M.-Z., Swenson, N., & Picard, R. W. (2010). A wearable sensor for unobtrusive, long-term assesment of electrodermal activity. IEEE Transactions on Biomedical Engineering, 57(5), 1243–1252.

[33] Taylor, S., Jaques, N., Chen, W., Fedor, S., Sano, A., & Picard, R. (n.d.). Automatic Identification of Artifacts in Electrodermal Activity Data.

[34] Vijaya, P. a., & Shivakumar, G. (2013). Galvanic Skin Response: A Physiological Sensor System for Affective Computing. International Journal of Machine Learning and Computing, 3(1), 31–34. http://doi.org/10.7763/IJMLC.2013.V3.267

[35] Villarejo, M. V., Zapirain, B. G., & Zorrilla, A. M. (2012). A stress sensor based on galvanic skin response (GSR) controlled by ZigBee. Sensors (Switzerland), 12(5), 6075–6101. http://doi.org/10.3390/s120506075

[36] Babikier, M. A., Izzeldin, M., Ishag, I. M., & Lee, D. G. (n.d.). Classification of Cardiac Arrhythmias using Machine learning techniques based on ECG Signal Matching.

[37] Banupriya, C. V, & Karpagavalli, S. (2014). Electrocardiogram Beat Classification Using Support Vector Machine and Extreme Learning Machine. ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India- Vol I SE - 22, 248, 187–193. http://doi.org/10.1007/978-3-319-03107-1_22

[38] Barrella, T., & Mccandlish, S. (2014). Identifying Arrhythmia from Electrocardiogram Data, 1–5.

[39] Bobbie, P. O., Chaudhari, H., Arif, C., & Pujari, S. (2004). Electrocardiogram (EKG) data acquisition and wireless transmission. WSEAS Transactions on Systems, 3(8), 2665–

2672. Retrieved from http://www.wseas.us/e-library/conferences/brazil2004/papers/470-152.pdf

[40] Congress, I. (2014). Electrocardiology 2014 - Proceedings of the 41, 249–252.

[41] Guvenir, H. a., Acar, B., Demiroz, G., & Cekin, a. (1997). A supervised machine learning algorithm for arrhythmia analysis. Computers in Cardiology 1997. http://doi.org/10.1109/CIC.1997.647926

[42] Kalkstein, N., Kinar, Y., Na'aman, M., Neumark, N., & Akiva, P. (2011). Using machine learning to detect problems in ECG data collection. 2011 Computing in Cardiology (CinC), 437–440.

[43] Karpagachelvi, S. (2011). Classification of Electrocardiogram Signals With Extreme Learning Machine and Relevance Vector Machine. Journal of Computer Science, 8(1).

[44] Kim, J., Shin, H. S., Shin, K., & Lee, M. (2009a). Robust algorithm for arrhythmia classification in ECG using extreme learning machine. Biomedical Engineering Online, 8, 31. http://doi.org/10.1186/1475-925X-8-31

[45] Kim, J., Shin, H. S., Shin, K., & Lee, M. (2009b). Robust algorithm for arrhythmia classification in ECG using extreme learning machine. Biomedical Engineering Online, 8, 31. http://doi.org/10.1186/1475-925X-8-31

[46] Oleksy, W., Tkacz, E., & Budzianowski, Z. (2012). Improving EASI ECG Method Using Various Machine Learning and Regression Techniques to Obtain New EASI ECG Model, 1(3), 287–289.

[47] Salem, A.-B. M., Revett, K., & El-Dahshan, E.-S. (2009). Machine learning techniques in electrocardiogram diagnosis, 429–433. Retrieved from http://westminsterresearch.wmin.ac.uk/7103/

[48] Science, E. (2015). Classification of Arrhythmias Using Support Vector Machine, 1–4.

[49] Soman, T., & Bobbie, P. (2005). Classification of arrhythmia using machine learning techniques. WSEAS Transactions on Computers, 4(6), 548–552. Retrieved from http://cse.spsu.edu/pbobbie/SharedFile/ECGDiagnosis_ICOSSE_2005_VFinal.pdf

[50] Vishwa, A., Lal, M. K., Dixit, S., & Vardwaj, P. (2011). Clasification Of Arrhythmic ECG Data Using Machine Learning Techniques. International Journal of Interactive Multimedia and Artificial Intelligence, 1(4), 67. http://doi.org/10.9781/ijimai.2011.1411

[51] Zhu, M., Cheng, L., Armstrong, J. J., Poss, J. W., Hirdes, J. P., & Stolee, P. (2014). Machine Learning in Healthcare Informatics, 56, 40017. http://doi.org/10.1007/978-3-642-40017-9

[52] Kavzoglu, T., & Vieira, C. a O. (1998). An analysis of artificial neural network pruning algorithms in relation to land cover classification accuracy. Proceedings of the Remote Sensing Society Student Conference, (May), 53–58.

[53] Burges, C. J. C. J. C. J. C. (1998). A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2(2), 121–167. http://doi.org/10.1023/A:1009715923555

[54] https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm, (last acceptance December 2015)

[55] Song, Y., Huang, J., Zhou, D., Zha, H., & Giles, C. L. (2007). IKNN : Informative K-Nearest Neighbor Pattern Classification, 248–264.

[56] http://deeplearning.net/tutorial/lenet.html, (last acceptance December 2015)

[57] https://en.wikipedia.org/wiki/Convolutional_neural_network, (last acceptance December 2015)

[58]http://www.aihorizon.com/essays/generalai/supervised_unsupervised_machine_learning.htm, (last acceptance December 2015)

[59] https://el.wikipedia.org/wiki/Συσταδοποίηση, (last acceptance December 2015)

[60] https://datasciencelab.wordpress.com/2013/12/27/finding-the-k-in-k-means-clustering/, (last acceptance December 2015)

[61] http://scikit-learn.org/stable/modules/clustering.html#overview-of-clustering-methods, (last acceptance December 2015)

[62] http://trendsofcode.net/kmeans/, (last acceptance December 2015)

[63] http://scikit-learn.org/stable/modules/clustering.html#k-means, (last acceptance December 2015)

[64] https://en.wikipedia.org/wiki/Determining_the_number_of_clusters_in_a_data_set, (last acceptance December 2015)

[65] Ringner, M. (2008). What is principal component analysis? Nat Biotechnol, 26(3), 303–304. http://doi.org/10.1038/nbt0308-303

[66] https://en.wikipedia.org/wiki/Principal_component_analysis, (last acceptance December 2015)

[67] http://ufldl.stanford.edu/tutorial/supervised/ConvolutionalNeuralNetwork/, (last acceptance January 2015)

[68] https://el.wikipedia.org/wiki/Νευρωνικό_δίκτυο, (last acceptance January 2015)

[69] http://neuralnetworksanddeeplearning.com, (last acceptance January 2015)

[70] https://en.wikipedia.org/wiki/Accuracy_and_precision, (last acceptance January 2015)

[71] https://en.wikipedia.org/wiki/Sensitivity_and_specificity,(last acceptance January 2015)

[72] https://en.wikipedia.org/wiki/Confusion_matrix,(last acceptance January 2015)

[73] http://scikit-learn.org/stable/modules/classes.html#clustering-metrics

[74] https://en.wikipedia.org/wiki/Silhouette_(clustering), (last acceptance January 2015)

[75] https://en.wikipedia.org/wiki/Artificial_intelligence, (last acceptance May 2016)

[76] https://en.wikipedia.org/wiki/Dynamic_time_warping, (last acceptance May 2016)

[77] https://www.quora.com/What-are-the-main-differences-between-artificial-intelligence-and-machine-learning, (last acceptance May 2016)

[78] http://scikit-learn.org/stable/, (last acceptance May 2016)

Παράρτημα Α

Μικρά παραδείγματα προγραμμάτων που χρησιμοποιήθηκαν για την επεξήγηση διάφορων εννοιών.

1. Για την εξαγωγή του σχήματος 4.3 γράφτηκε ο πιο κάτω κώδικας στην γλώσσα python. Εφαρμόστηκε ο αλγόριθμος kmeans για την συσταδοποίηση σημείων με συνιστώσες (x,y), ο οποίος διαχώρισε τα δεδομένα σε δύο συστάδες. Τα δεδομένα της μια συστάδας εμφανίζονται με κόκκινο χρώμα και της άλλης με πράσινο.

kmeans.py

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import style
style.use("ggplot")
from sklearn.cluster import KMeans


x=[1,5,1.5,8,1,9]
y=[2,8,1.8,8,0.6,11]

plt.scatter(x,y)

plt.show()

X= np.array([[1,2],
             [5,8],
             [1.5,1.8],
             [8,8],
             [1,0.6],
             [9,11]])
kmeans =KMeans(n_clusters=2)
kmeans.fit(X)

centroids= kmeans.cluster_centers_
labels= kmeans.labels_

print(centroids)
print(labels)

colors= ["g.","r."]

for i in range(len(X)):
```

```
    print("coordinate:",X[i], "label:", labels[i])
    plt.plot(X[i][0],  X[i][1], colors[labels[i]], markersize=10)

plt.scatter(centroids[:, 0], centroids[:, 1], marker="x", s=150, linewidths=5,zorder=10)
plt.show()
```

2. Για την εξαγωγή του σχήματος 4.4 γράφτηκε ο πιο κάτω κώδικας στην γλώσσα python. Εφαρμόστηκε ο αλγόριθμος kmeans για την συσταδοποίηση λουλουδιών ίρις με βάση τρία χαρακτηριστικά, το μήκος του φύλλου, το πλάτος του φύλλου και το μήκος του πετάλου. Τα δεδομένα διαχωρίστηκαν σε τρεις συστάδες, οι οποίες αναπαρίστανται με πράσινο, κόκκινο και μπλε χρώμα.

iris.py

```
from sklearn.cluster import KMeans
from sklearn import datasets

# load the iris dataset
iris = datasets.load_iris()
X = iris.data
Y=iris.target

import numpy as np

kmeans =KMeans(n_clusters=3)
kmeans.fit(X)

centroids= kmeans.cluster_centers_
labels= kmeans.labels_

#colors= ["g.","r.", "b."]

import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D


fig = plt.figure()
ax = plt.axes(projection='3d')
ax.scatter(X[:,0],  X[:,1],X[:,3],c=kmeans.labels_)
plt.show()
```

3. Για την εξαγωγή του σχήματος 4.5 γράφτηκε ο πιο κάτω κώδικας στην γλώσσα python. Εφαρμόστηκε η μέθοδος elbow στα δεδομένα iris για την εύρεση του βέλτιστου αριθμού συστάδων-k.

```
import numpy as np
from sklearn import cluster
from scipy.spatial.distance import cdist
import matplotlib.pyplot as plt
import scipy.io
import numpy as np


# load the iris dataset
from sklearn import datasets
iris = datasets.load_iris()
X = iris.data

#cluster data into K=1..8 clusters
K = range(1,8)

kmeans = [cluster.KMeans(n_clusters = i, init="k-means++").fit(X) for i in K]
centroids= [kmeans[i].cluster_centers_ for i in range(0,len(kmeans))]


D_k = [cdist(X, cent, 'euclidean') for cent in centroids]
cIdx = [np.argmin(D,axis=1) for D in D_k]
dist = [np.min(D,axis=1) for D in D_k]
avgWithinSS = [sum(d)/X.shape[0] for d in dist]


# elbow curve
fig = plt.figure()
ax = fig.add_subplot(111)
ax.plot(K, avgWithinSS, 'b*-')
plt.grid(True)
plt.xlabel('Number of clusters')
plt.ylabel('Average within-cluster sum of squares')
plt.title('Elbow for KMeans clustering (Iris)')

plt.show()
```

Παράρτημα Β

Κώδικας κεφαλαίου 6.1

Κατηγοροποιητής SVM

```
from sklearn import svm
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath
import warnings
warnings.filterwarnings("ignore")

s = []
e =[]
D=[]
names=[]
for path in Path('/Users/andria/Desktop/first project/hist2').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    data=[]
    for i,x in zip(range(0,len(X)),X):
        if int(X[i,7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(X[i,7])==0 and X[i-1,7]==5:
            e.append(i-1)



    if len(s)==5:


        data=X[s[0]:s[0]+250354,1].tolist()
        data.extend(X[s[1]:s[1]+419997,1].tolist())
        data.extend(X[s[2]:s[2]+419996,1].tolist())
        data.extend(X[s[3]:s[3]+420006,1].tolist())
        data.extend(X[s[4]:s[4]+2295,1].tolist())

        D.append(data)

        name=ntpath.basename(str(path))
        name=name[5:]
        name=name.split(".")[0]

        name=name.replace(" ", "")

        names.append(int(name))
```

```
    else:
        print(path)

print(len(D))



Y=[0,0,1,1,0,0,0,0,1,0,1,0,1,1,0,1,1,1,0,0,1,0,1,1,1]


clf=svm.SVC(kernel='linear',C=1.0)
clf.fit(D,Y)

P=[]
for path in Path('/Users/andria/Desktop/first project/hist').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    data=[]
    for i,x in zip(range(0,len(X)),X):
        if int(X[i,7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(X[i,7])==0 and X[i-1,7]==5:
            e.append(i-1)



    if len(s)==5:


        data=X[s[0]:s[0]+250354,1].tolist()
        data.extend(X[s[1]:s[1]+419997,1].tolist())
        data.extend(X[s[2]:s[2]+419996,1].tolist())
        data.extend(X[s[3]:s[3]+420006,1].tolist())
        data.extend(X[s[4]:s[4]+2295,1].tolist())

        P.append(data)

for p in P:
    print(clf.predict(p))
```

Κατηγοροποιητής KNN

```
from sklearn.neighbors import KNeighborsClassifier
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath
import warnings
warnings.filterwarnings("ignore")

s = []
e =[]
D=[]
names=[]
for path in Path('/Users/andria/Desktop/first project/hist2').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    data=[]
    for i,x in zip(range(0,len(X)),X):
        if int(X[i,7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(X[i,7])==0 and X[i-1,7]==5:
            e.append(i-1)

    if len(s)==5:
        data=X[s[0]:s[0]+250354,1].tolist()
        data.extend(X[s[1]:s[1]+419997,1].tolist())
        data.extend(X[s[2]:s[2]+419996,1].tolist())
        data.extend(X[s[3]:s[3]+420006,1].tolist())
        data.extend(X[s[4]:s[4]+2295,1].tolist())

        D.append(data)

        name=ntpath.basename(str(path))
        name=name[5:]
        name=name.split(".")[0]

        name=name.replace(" ", "")

        names.append(int(name))
    else:
        print(path)

    print(len(D))

y=[0,0,1,1,0,0,0,0,1,0,1,0,1,1,0,1,1,1,0,0,1,0,1,1,1]


neigh = KNeighborsClassifier(n_neighbors=3)
neigh.fit(D, y)
```

```
P=[]
for path in Path('/Users/andria/Desktop/first project/hist').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    data=[]
    for i,x in zip(range(0,len(X)),X):
        if int(X[i,7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(X[i,7])==0 and X[i-1,7]==5:
            e.append(i-1)



    if len(s)==5:


        data=X[s[0]:s[0]+250354,1].tolist()
        data.extend(X[s[1]:s[1]+419997,1].tolist())
        data.extend(X[s[2]:s[2]+419996,1].tolist())
        data.extend(X[s[3]:s[3]+420006,1].tolist())
        data.extend(X[s[4]:s[4]+2295,1].tolist())
       P.append(data)
        for p in P:
    print(neigh.predict(p))
```

Κώδικας του κεφαλαίου 6.2

Ομαδοποιητής k-means με βάση τον μέσο όρο (average) των μετρήσεων

```
from sklearn.cluster import KMeans
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath

s = []
e =[]
D=[]
names=[]
for path in Path('/Users/andria/Desktop/first project/bio-file/').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    avg=[]
    for i,x in zip(range(0,len(X)),X):
        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)

    if len(s)==5:

        avg = [np.average(np.hstack((X[s[3]:e[3],1],X[s[4]:e[4],1])))]
        D.append(avg)

        name=ntpath.basename(str(path))
        name=name[5:]
        name=name.split(".")[0]

        name=name.replace(" ", "")

        names.append(int(name))
    else:
        print(path)

kmeans =KMeans(n_clusters=2)

kmeans.fit(D)


np.savetxt("2_clusters_average_SCR_cut.csv",np.c_[names,kmeans.labels_],delimiter=",",fmt=" %d, %d “)
```

Ομαδοποιητής k-means με βάση την διάμεσο (median)

```
from sklearn.cluster import KMeans
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath

s = []
e =[]
D=[]
names=[]
for path in Path('/Users/andria/Desktop/first project/bio-file/').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    median=[]
    for i,x in zip(range(0,len(X)),X):
        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)

    if len(s)==5:

        median=[np.median(np.hstack((X[s[3]:e[3],1],X[s[4]:e[4],1])))]
        D.append(median)

        name=ntpath.basename(str(path))
        name=name[5:]
        name=name.split(".")[0]

        name=name.replace(" ", "")

        names.append(int(name))
    else:
        print(path)

print(len(D))

kmeans =KMeans(n_clusters=2)

kmeans.fit(D)


np.savetxt("2_clusters_median_SCR_cut.csv",np.c_[names,kmeans.labels_],delimiter=",",fmt=" %d, %d “)
```

Ομαδοποιητής k-means με βάση το ιστόγραμμα (histogram)

```
import numpy as np
from sklearn.cluster import KMeans
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath


s = []
e =[]
Hist=[]
names=[]

for path in Path('/Users/andria/Desktop/first project/bio-file/').glob('*.mat'):
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    Y=[]
    hist=[]

    for i,x in zip(range(0,len(X)),X):
        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)



    if len(s)==5:
        Y=X[s[0]:e[0],1].tolist()
        np.append(Y,X[s[1]:e[1],1].tolist())
        np.append(Y,X[s[2]:e[2],1].tolist())
        np.append(Y,X[s[3]:e[3],1].tolist())
        np.append(Y,X[s[4]:e[4],1].tolist())

        hist, bin_edges = np.histogram(Y,bins=50, normed=True)#Y,range=(-5,25)
        Hist.append(hist)

        name=ntpath.basename(str(path))
        name=name[5:]
        name=name.split(".")[0]

        name=name.replace(" ", "")

        names.append(int(name))
    else:
        print(path)
```

```
kmeans =KMeans(n_clusters=2)

kmeans.fit(Hist)


np.savetxt("histogram2.csv",np.c_[names,kmeans.labels_],delimiter=",",fmt=" %d, %d “)
```

Παράρτημα Γ

Εισάγεται το θέμα που προτίθεστε να διατυπώσετε στο παράρτημα αυτό. Εισάγεται το θέμα που προτίθεστε να διατυπώσετε στο παράρτημα αυτό. Εισάγεται το θέμα που προτίθεστε να διατυπώσετε στο παράρτημα αυτό.

Κώδικας κεφαλαίου 7.1

Κατηγοροποιητής SVM

```
from sklearn import svm
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath
import warnings
warnings.filterwarnings("ignore")

s = []
e =[]
D=[]
names=[]


for path in Path('/Volumes/VERBATIM HD/Data_mkoushiou/f2').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    data=[]



    for i,x in zip(range(0,len(X)),X):

        if int(x[6])==5 and X[i-1,6]==0:
            s.append(i)
        if int(x[6])==0 and X[i-1,6]==5:
            e.append(i-1)

        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
```

```
            e.append(i-1)

        if int(x[8])==5 and X[i-1,8]==0:
            s.append(i)
        if int(x[8])==0 and X[i-1,8]==5:
            e.append(i-1)

        if int(x[9])==5 and X[i-1,9]==0:
            s.append(i)
        if int(x[9])==0 and X[i-1,9]==5:
            e.append(i-1)

        if int(x[10])==5 and X[i-1,10]==0:
            s.append(i)
        if int(x[10])==0 and X[i-1,10]==5:
            e.append(i-1)



    if len(s)==5:


        data=X[s[0]:s[0]+299997,0].tolist()
        data.extend(X[s[1]:s[1]+137998,0].tolist())
        data.extend(X[s[2]:s[2]+137998,0].tolist())
        data.extend(X[s[3]:s[3]+137998,0].tolist())
        data.extend(X[s[4]:s[4]+126102,0].tolist())

        D.append(data)

        name=ntpath.basename(str(path))
        name=name.split(".")[0]
        name=name.split("_")[1]
        names.append(int(name))

    else:
        print(path)

    #maxlen = max([len(member) for member in Y])
    #[member.extend("" * (maxlen - len(member))) for member in Y]

print(len(D))

Y = [0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1]

clf=svm.SVC(kernel='linear',C=1.0)
clf.fit(D,Y)




P=[]
for path in Path('/Volumes/VERBATIM HD/Data_mkoushiou/f2_test').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
```

```
    s=[]
    e=[]
    data=[]
    for i,x in zip(range(0,len(X)),X):

        if int(x[6])==5 and X[i-1,6]==0:
            s.append(i)
        if int(x[6])==0 and X[i-1,6]==5:
            e.append(i-1)

        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)

        if int(x[8])==5 and X[i-1,8]==0:
            s.append(i)
        if int(x[8])==0 and X[i-1,8]==5:
            e.append(i-1)

        if int(x[9])==5 and X[i-1,9]==0:
            s.append(i)
        if int(x[9])==0 and X[i-1,9]==5:
            e.append(i-1)

        if int(x[10])==5 and X[i-1,10]==0:
            s.append(i)
        if int(x[10])==0 and X[i-1,10]==5:
            e.append(i-1)



    if len(s)==5:


        data=X[s[0]:s[0]+299997,0].tolist()
        data.extend(X[s[1]:s[1]+137998,0].tolist())
        data.extend(X[s[2]:s[2]+137998,0].tolist())
        data.extend(X[s[3]:s[3]+137998,0].tolist())
        data.extend(X[s[4]:s[4]+126102,0].tolist())

        P.append(data)



for p in P:
    print(clf.predict(p))
```

Κατηγοροποιητής KNN

```
from sklearn.neighbors import KNeighborsClassifier
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath
import warnings
warnings.filterwarnings("ignore")

s = []
e =[]
D=[]
names=[]


for path in Path('/Volumes/VERBATIM HD/Data_mkoushiou/f2').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    data=[]



    for i,x in zip(range(0,len(X)),X):

        if int(x[6])==5 and X[i-1,6]==0:
            s.append(i)
        if int(x[6])==0 and X[i-1,6]==5:
            e.append(i-1)

        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)

        if int(x[8])==5 and X[i-1,8]==0:
            s.append(i)
        if int(x[8])==0 and X[i-1,8]==5:
            e.append(i-1)

        if int(x[9])==5 and X[i-1,9]==0:
            s.append(i)
        if int(x[9])==0 and X[i-1,9]==5:
            e.append(i-1)

        if int(x[10])==5 and X[i-1,10]==0:
            s.append(i)
        if int(x[10])==0 and X[i-1,10]==5:
            e.append(i-1)
```

```
    if len(s)==5:


        data=X[s[0]:s[0]+299997,0].tolist()
        data.extend(X[s[1]:s[1]+137998,0].tolist())
        data.extend(X[s[2]:s[2]+137998,0].tolist())
        data.extend(X[s[3]:s[3]+137998,0].tolist())
        data.extend(X[s[4]:s[4]+126102,0].tolist())

        D.append(data)

        name=ntpath.basename(str(path))
        name=name.split(".")[0]
        name=name.split("_")[1]
        names.append(int(name))

    else:
        print(path)

    #maxlen = max([len(member) for member in Y])
    #[member.extend("" * (maxlen - len(member))) for member in Y]

print(len(D))

Y = [0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1]

neigh = KNeighborsClassifier(n_neighbors=3)
neigh.fit(D, Y)




P=[]
for path in Path('/Volumes/VERBATIM HD/Data_mkoushiou/f2_test').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    data=[]
    for i,x in zip(range(0,len(X)),X):

        if int(x[6])==5 and X[i-1,6]==0:
            s.append(i)
        if int(x[6])==0 and X[i-1,6]==5:
            e.append(i-1)

        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)

        if int(x[8])==5 and X[i-1,8]==0:
            s.append(i)
        if int(x[8])==0 and X[i-1,8]==5:
            e.append(i-1)
```

```
        if int(x[9])==5 and X[i-1,9]==0:
            s.append(i)
        if int(x[9])==0 and X[i-1,9]==5:
            e.append(i-1)

        if int(x[10])==5 and X[i-1,10]==0:
            s.append(i)
        if int(x[10])==0 and X[i-1,10]==5:
            e.append(i-1)



    if len(s)==5:


        data=X[s[0]:s[0]+299997,0].tolist()
        data.extend(X[s[1]:s[1]+137998,0].tolist())
        data.extend(X[s[2]:s[2]+137998,0].tolist())
        data.extend(X[s[3]:s[3]+137998,0].tolist())
        data.extend(X[s[4]:s[4]+126102,0].tolist())

        P.append(data)



for p in P:
    print(neigh.predict(p))
```

Κώδικας του κεφαλαίου 7.2

Ομαδοποιητής k-means με βάση τον μέσο όρο (average) των μετρήσεων

```
from sklearn.cluster import KMeans
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath
import pandas as pd
from sklearn import decomposition
from sklearn.preprocessing import normalize


D=[]
names=[]
for path in Path('/Volumes/VERBATIM HD/Data_mkoushiou/format').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    avg=[]

    for i,x in zip(range(0,len(X)),X):

        if int(x[6])==5 and X[i-1,6]==0:
            s.append(i)
        if int(x[6])==0 and X[i-1,6]==5:
            e.append(i-1)

        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)

        if int(x[8])==5 and X[i-1,8]==0:
            s.append(i)
        if int(x[8])==0 and X[i-1,8]==5:
            e.append(i-1)

        if int(x[9])==5 and X[i-1,9]==0:
            s.append(i)
        if int(x[9])==0 and X[i-1,9]==5:
            e.append(i-1)

        if int(x[10])==5 and X[i-1,10]==0:
            s.append(i)
        if int(x[10])==0 and X[i-1,10]==5:
            e.append(i-1)
```

```
    #print(len(e),len(s),"\n")

    if len(s)==5:

       avg = [np.average(np.hstack((X[s[0]:e[0],0]))),
              np.average(np.hstack((X[s[1]:e[1],0],X[s[2]:e[2],0]))),
              np.average(np.hstack((X[s[3]:e[3],0],X[s[4]:e[4],0])))]

        D.append(avg)

        name=ntpath.basename(str(path))
        name=name.split(".")[0]
        name=name.split("_")[1]
        names.append(int(name))

    else:
        print(path)



print(len(D))
kmeans =KMeans(n_clusters=2)

kmeans.fit(D)


np.savetxt("2_clusters_avg_second_ecg_last2tasks.csv",np.c_[names,kmeans.labels_],delimiter=",",fmt=" %d, %d ")
```

Ομαδοποιητής k-means με βάση την διάμεσο (median)

```
from sklearn.cluster import KMeans
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath
import pandas as pd
from sklearn import decomposition
from sklearn.preprocessing import normalize


D=[]
names=[]
for path in Path('/Volumes/VERBATIM HD/Data_mkoushiou/format').glob('*.mat'):#first project/bio-file/
    X=loadmat(str(path))['data']
    s=[]
    e=[]
    median=[]
```

```
    for i,x in zip(range(0,len(X)),X):

        if int(x[6])==5 and X[i-1,6]==0:
            s.append(i)
        if int(x[6])==0 and X[i-1,6]==5:
            e.append(i-1)

        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)

        if int(x[8])==5 and X[i-1,8]==0:
            s.append(i)
        if int(x[8])==0 and X[i-1,8]==5:
            e.append(i-1)

        if int(x[9])==5 and X[i-1,9]==0:
            s.append(i)
        if int(x[9])==0 and X[i-1,9]==5:
            e.append(i-1)

        if int(x[10])==5 and X[i-1,10]==0:
            s.append(i)
        if int(x[10])==0 and X[i-1,10]==5:
            e.append(i-1)


    if len(s)==5:
       median= [np.median(np.hstack((X[s[0]:e[0],0]))),
                np.median(np.hstack((X[s[1]:e[1],0],X[s[2]:e[2],0]))),
                np.median(np.hstack((X[s[3]:e[3],0],X[s[4]:e[4],0])))]
        D.append(median)

        name=ntpath.basename(str(path))
        name=name.split(".")[0]
        name=name.split("_")[1]
        names.append(int(name))

    else:
        print(path)

print(len(D))

kmeans =KMeans(n_clusters=2)

kmeans.fit(D)


np.savetxt("2_clusters_median_second_3features5tasks_norm.csv",np.c_[names,kmeans.labels_],delimiter=",",fmt=" %d, %d ")
```

Ομαδοποιητής k-means με βάση το ιστόγραμμα (histogram)

```
from sklearn.cluster import KMeans
from scipy.io import loadmat
import numpy as np
from pathlib import Path
import ntpath
import pandas as pd
from sklearn import decomposition


D=[]
names=[]
for path in Path('/Volumes/VERBATIM HD/Data_mkoushiou/format').glob('*.mat'):#first project/bio-file/
    X = loadmat(str(path))['data']
    s = []
    e = []
    hist = []
    X0 = []
    X1 = []
    X3 = []
    Y = []
    for i,x in zip(range(0,len(X)),X):

        if int(x[6])==5 and X[i-1,6]==0:
            s.append(i)
        if int(x[6])==0 and X[i-1,6]==5:
            e.append(i-1)
        if int(x[7])==5 and X[i-1,7]==0:
            s.append(i)
        if int(x[7])==0 and X[i-1,7]==5:
            e.append(i-1)
        if int(x[8])==5 and X[i-1,8]==0:
            s.append(i)
        if int(x[8])==0 and X[i-1,8]==5:
            e.append(i-1)
          if int(x[9])==5 and X[i-1,9]==0:
            s.append(i)
        if int(x[9])==0 and X[i-1,9]==5:
            e.append(i-1)
        if int(x[10])==5 and X[i-1,10]==0:
            s.append(i)
        if int(x[10])==0 and X[i-1,10]==5:
            e.append(i-1)
    if len(s)==5
        Y=X[s[3]:e[3],0].tolist()
        np.append(Y,X[s[4]:e[4],0].tolist())
          hist, bin_edges = np.histogram(Y,bins=50, normed=True )
        D.append(hist)
        name=ntpath.basename(str(path))
        name=name.split(".")[0]
        name=name.split("_")[1]
        names.append(int(name))
    else:
```

```
        print(path)
print(len(D))
kmeans =KMeans(n_clusters=2)
kmeans.fit(D)
np.savetxt("histo_second_50bins_new_notpca_range_1.csv",np.c_[names,kmeans.labels_],delimiter=",",fmt=" %d, %d ")
```