ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ ΜΕ ΘΕΜΑ:

« Ταξινόμηση με Χρήση αλγορίθμων Data Mining και Ασαφούς Λογικής: Μια Εφαρμογή στο Μετρό του Παρισιού »

ΠΑΛΑΙΟΛΟΓΟΣ ΧΡΗΣΤΟΣ
Α.Μ. 2002030122

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΣΤΑΥΡΑΚΑΚΗΣ ΓΕΩΡΓΙΟΣ

ΧΑΝΙΑ
ΑΠΡΙΛΙΟΣ 2009

ΠΡΟΛΟΓΟΣ

Σκοπός της παρούσας διπλωματικής εργασίας είναι η ανάπτυξη και αξιολόγηση μοντέλων πρόβλεψης κλάσης μη ταξινομημένων δεδομένων αναγνωρίζοντας πρότυπα τα οποία θα δημιουργηθούν με τη χρήση αλγορίθμων Data Mining και ασαφούς λογικής. Τα δεδομένα με τα οποία θα πειραματιστούμε προέρχονται από ερωτηματολόγιο ικανοποίησης που αφορά το μετρό του Παρισιού.

Το πρόβλημα της ταξινόμησης παρουσίαζε ανέκαθεν μεγάλο ερευνητικό και πρακτικό ενδιαφέρον με σκοπό την εφαρμογή κατάλληλων τεχνικών για την όσο καλύτερη αναγνώριση των κλάσεων στις οποίες ανήκουν τα εκάστοτε δεδομένα. Η ταξινόμηση λαμβάνει χώρα σε πολλές εφαρμογές επιστημονικών πεδίων όπως στην επιστήμη των υπολογιστών, στην ιατρική, στην μετεωρολογία, στα οικονομικά και σε πολλά άλλα. Ετσι λοιπόν γίνεται κατανοητό το πόσο σημαντικό είναι η ανάπτυξη τεχνικών οι οποίες θα μπορούν να μας προσφέρουν αξιόπιστη ταξινόμηση.

Απώτερος στόχος της εργασίας είναι η σύγκριση τεχνικών ταξινόμησης και και η βελτίωσή τους σε όσο μεγαλύτερο βαθμό μπορούμε. Αρχικά θα γίνει χρήση του λογισμικού μάθησης WEKA που περιέχει αλγορίθμους Data Mining έτσι ώστε να εξαχθούν τα πρώτα αποτελέσματα ‘αφετηρία’. Στην συνέχεια, στην προσπάθεια σύγκρισης και εξαγωγής βελτιωμένων αποτελεσμάτων θα υλοποιήσουμε ένα σύστημα ταξινομητή με χρήση ασαφούς λογικής και τέλος θα κάνουμε χρήση γενετικών αλγορίθμων που αποτελεί μια μέθοδο βελτιστοποίησης προβλημάτων.

Η παρούσα διπλωματική εργασία αφιερώνεται στην αδερφή μου Ευαγγελία και στους γονείς μου, Κωνσταντίνο και Φανή

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ.Σταυρακάκη και τον κ.Ματσατσίνη για την επιμέλεια του θέματος της διπλωματικής μου εργασίας.

Επίσης ευχαριστώ τον κ.Μαρινάκη για την καθοδήγησή του και τις επικοδομητικές παρατηρήσεις του.

Τέλος ευχαριστώ πολύ τους φίλους μου για τις υπέροχες στιγμές που περάσαμε και την ψυχολογική τους υποστήριξή όταν τους χρειάστηκα.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1 :ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΟΜΑΔΟΠΟΙΗΣΗ

1.1 Εισαγωγή

Η πρακτική της κατάταξης των αντικείμενων σε κλάσεις σύμφωνα με κάποιες αντιλαμβανόμενες ομοιότητες αποτελεί σημαντικό κομμάτι σε πολλές εφαρμογές στο πεδίο της επιστήμης των υπολογιστών. Η κατηγοριοποίηση των απλών και καθημερινών δεδομένων σε αισθητές κατηγορίες είναι ένας από τους πιο θεμελιώδεις τρόπους με τους οποίους οι άνθρωποι αντιλαμβάνονται τον κόσμο, μαθαίνουν, σκέφτονται και ενεργούν με βάση αυτές. Από πολύ μικρή ηλικία μαθαίνουμε υποσυνείδητα να ταξινομούμε τα αντικείμενα του περιβάλλοντος και να συνδέουμε τις προκύπτουσες κατηγορίες με ουσιαστικά της γλώσσας μας. Για παράδειγμα όταν βλέπουμε ένα σκύλο, αμέσως αναγνωρίζουμε την οντότητα «σκύλος » ως ένα αντικείμενο της κατηγορίας «σκύλος » και έτσι μπορούμε να συμπεράνουμε ότι το αντικείμενο που εντοπίσαμε θα γαυγίζει ή θα τρώει κόκαλα χωρίς να το δούμε να το κάνει καθώς αυτά θα είναι τα χαρακτηριστικά τα οποία θα χαρακτηρίζουν την κατηγορία «σκύλος » και θα την διαχωρίζουν από μια άλλη κατηγορία ενός άλλου ζώου.

Ο όρος ***Ομαδοποίηση*** είναι ένα γενικό όνομα για μια ευρεία ποικιλία διαδικασιών που χρησιμοποιούνται για την δημιουργία μιας ταξινόμησης. Στόχος της ομαδοποίησης είναι η ανάκτηση «λογικών» ομάδων οι οποίες προϋπάρχουν στα δεδομένα και οι οποίες επιτρέπουν την ανακάλυψη ομοιοτήτων και διαφορών ανάμεσα στα δεδομένα έτσι ώστε να παράγονται χρήσιμα συμπεράσματα για αυτά. Η Ομαδοποίηση χρησιμοποιείται για τον χειρισμό του μεγάλου πλήθους των δεδομένων που λαμβάνουν καθημερινά οι άνθρωποι. Κάθε ομάδα χαρακτηρίζεται μέσω του κοινού χαρακτηριστικού των οντοτήτων που περιέχει. Τα αντικείμενα που θα καταταχτούν σε ομάδες περιγράφονται είτε από ένα σύνολο μετρήσεων είτε από τις σχέσεις που έχει αυτό το αντικείμενο με τα άλλα υπό εξέταση αντικείμενα.

Δεν είναι τυχαίο ότι η ομαδοποίηση χρησιμοποιείται σε μια πλειάδα ετερόκλιτων μεταξύ τους επιστημών. Έχει χρησιμοποιηθεί στις επιστήμες ζωής (βιολογία, ζωολογία), στις ιατρικές επιστήμες (ψυχολογία, παθολογία), στις

κοινωνικές επιστήμες (κοινωνιολογία, αρχαιολογία), στις επιστήμες της γης (γεωγραφία, γεωλογία) αλλά και στις επιστήμες των μηχανών. Ο όρος «ομαδοποίηση» παίρνει διαφορετικά ονόματα στις διαφορετικές επιστήμες. Έτσι παραδείγματος χάριν στην αναγνώριση προτύπων ονομάζεται εκπαίδευση με ή χωρίς επίβλεψη, στην βιολογία και στην οικολογία αναφέρεται ως αριθμητική ταξινόμηση (numerical taxonomy), στις κοινωνικές επιστήμες ως τυπολογία (typology), ενώ στην θεωρία γράφων ως τμηματοποίηση (partition). Τέλος στο πεδίο της εξόρυξης δεδομένων ονομάζεται και μη κατευθυνόμενη ανακάλυψη γνώσης (undirected knowledge discovery) .

Αναφέραμε παραπάνω ότι ο άνθρωπος είναι από την φύση του πάρα πολύ ικανός στην δημιουργία κατηγοριοποιήσεων και στην κατάταξη οντοτήτων σε διαφορετικές ομάδες. Παρακάτω παρουσιάζονται δύο βασικοί λόγοι για τους οποίους χρησιμοποιούμε προγράμματα ομαδοποίησης και γενικότερα προτιμούμε την χρήση υπολογιστών για την επίλυση τέτοιου είδους προβλημάτων.

- Κατά πρώτο λόγο, θα πρέπει να αναφέρουμε, ότι ένα πρόγραμμα ομαδοποίησης μπορεί να χρησιμοποιήσει ένα συγκεκριμένο αντικειμενικό κριτήριο κατάταξης πολύ πιο συνειδητά και αντικειμενικά απ' ότι ένας άνθρωπος. Οι άνθρωποι έχουν την ικανότητα της διαίρεσης των αντικειμένων που τους περιβάλλουν σε διάφορες κατηγορίες, όμως το αποτέλεσμα είναι υποκειμενικό καθώς τα διαφορετικά άτομα δεν θα αναγνωρίσουν πάντα τις ίδιες ομάδες στα ίδια σύνολα δεδομένων. Αυτό συμβαίνει καθώς κάθε άτομο θα έχει διαφορετικό μέτρο ομοιότητας που εξαρτάται από τον χαρακτήρα του και την προσωπική του εμπειρία. Έτσι είναι πολύ πιθανό, δυο διαφορετικά άτομα να κατηγοριοποιήσουν ίδια χαρακτηριστικά σε διαφορετικές ομάδες και ειδικά όταν οι διαχωριστικές γραμμές των ομάδων αυτών είναι λεπτές και είναι δύσκολο να τις διακρίνει ένας άνθρωπος

- Κατά δεύτερο λόγο, αλλά εξίσου σημαντικό, θα πρέπει να αναφέρουμε ότι ένας υπολογιστής μπορεί να ανακαλύψει ομάδες πολύ πιο γρήγορα και αξιόπιστα σ' ένα σύνολο δεδομένων απ' ότι ένας άνθρωπος, ειδικά εάν οι οντότητες που υπάρχουν στο σύνολο των δεδομένων μας, χαρακτηρίζονται

από έναν μεγάλο αριθμό χαρακτηριστικών. Η ταχύτητα, η αξιοπιστία και η συνέπεια με την οποία ένας αλγόριθμος ομαδοποίησης θα ανακαλύψει ομάδες σε ένα τεράστιο σύνολο από δεδομένα είναι ένα ισχυρό κίνητρο για να τον χρησιμοποιήσουμε. Ένας αλγόριθμος ομαδοποίησης θα απελευθερώσει τον μηχανικό δεδομένων από την κουραστική δουλειά της ομαδοποίησης έτσι ώστε να μπορέσει αυτός να ξοδέψει περισσότερο χρόνο στην ανάλυση των ομάδων που θα παραχθούν.

1.2 Τι είναι ομάδα: Ορισμοί.

Ας προσπαθήσουμε τώρα να δώσουμε έναν ορισμό για το τι είναι **Ομάδα**. Ο Everitt δίνει μεταξύ άλλων τους εξής ορισμούς της Ομάδας:

- Ομάδα είναι το σύνολο των οντοτήτων που είναι *όμοιες*, και οντότητες από διαφορετικά σύνολα δεν είναι όμοιες.

- Ομάδα είναι μια συσσωμάτωση σημείων του πεδίου αναφοράς τέτοια ώστε η απόσταση κάθε δυο σημείων της Ομάδας να είναι μικρότερη από την απόσταση μεταξύ κάθε σημείου της Ομάδας και οποιουδήποτε άλλο σημείου.

- Οι ομάδες μπορούν να περιγραφούν ως συνδεδεμένες περιοχές ενός πολυδιάστατου χώρου, που περιέχουν σχετικά υψηλή πυκνότητα σημείων και που είναι χωρισμένες μεταξύ τους με περιοχές, η πυκνότητα των οποίων είναι σχετικά μικρότερη.

Μπορούμε πολύ εύκολα να αναγνωρίσουμε μια Ομάδα όταν την δούμε στο επίπεδο αλλά δεν είναι ξεκάθαρο το πως το κάνουμε. Πολλές φορές οι ομάδες εξαρτώνται και από την λεπτομέρεια με την οποία κοιτάμε τα δεδομένα μας. Η Εικόνα 1.1 παρουσιάζει μερικά σημεία στο δυσδιάστατο επίπεδο. Αν παρατηρήσουμε την εικόνα με όχι μεγάλη λεπτομέρεια θα διακρίνουμε 4 ομάδες. Αν τώρα παρατηρήσουμε τα δεδομένα μας με μεγαλύτερη ακρίβεια θα τα χωρίζαμε σε 12 ομάδες. Βλέποντας λοιπόν τα δεδομένα υπό διαφορετικές κλίμακες μας βοηθάει να

κατανοήσουμε καλύτερα την δομή τους. Έτσι ένα από τα κρίσιμα προβλήματα στην εύρεση των Ομάδων είναι να καθορίσουμε τι είναι εγγύτητα και να βρούμε έναν τρόπο να την μετρήσουμε.

Εικόνα 1.1: Παραδείγματα ομάδων στο επίπεδο

Για το λόγο αυτό θα παραθέσουμε τώρα και μερικούς ακόμα ορισμούς για το τι είναι μία ομάδα.

- Είναι ένα σύνολο από συνεχόμενα στοιχεία ενός στατιστικού πληθυσμού, για παράδειγμα το σύνολο από ανθρώπους που κατοικούν σε ένα μόνο σπίτι, μια συνεχή ροή παρατηρήσεων σε μια ταξινομημένη σειρά και ένα σύνολο από γειτονικά κομμάτια σε ένα πεδίο.(Kendall και Buckland Dictionary of Statistical Terms).

- Είναι ένα σύνολο περισσότερο όμοιων μεταξύ τους οντοτήτων σε σχέση με οντότητες από άλλες ομάδες.

- Είναι ένα υποσύνολο από οντότητες οι οποίες μπορούν να χρησιμοποιηθούν κατά μια έννοια ως ισοδύναμες.(Wallace και Boulton 1968)

Το κοινό χαρακτηριστικό όλων αυτών των ορισμών είναι ότι χρησιμοποιούν τις έννοιες της ομοιότητας, της απόστασης και της ταύτισης χωρίς να ορίζονται εκ των προτέρων αυτές οι έννοιες. Για τον λόγο αυτό ο Bonner (1964) πρότεινε ως βασικό κριτήριο για τον καθορισμό τέτοιων εννοιών, όπως ομάδα και ομοιότητα, να είναι στη κρίση του χρήστη. Παρά το γεγονός όμως πως δεν υπάρχει ένας κοινά αποδεχτός ορισμός της ομάδας, εντούτοις θα πρέπει οι ομάδες να έχουν κάποιες κοινές ιδιότητες. Αυτές περιγράφονται από τους Sneath και Sokal (1973). Οι πιο σημαντικές από αυτές τις ιδιότητες είναι οι εξής:

Πυκνότητα (Density): είναι εκείνη η ιδιότητα που καθορίζει την ομάδα ως ένα σχετικά παχύ σμήνος από σημεία σε έναν χώρο, συγκρινόμενη με άλλες περιοχές του χώρου, που μπορεί να έχουν λιγότερα αν όχι καθόλου σημεία. Δεν υπάρχει απόλυτο μέτρο πυκνότητας.

Διασπορά (Variance): είναι ο βαθμός διασκόρπισης των σημείων σε σχέση με το κέντρο της ομάδας. Η ιδιότητα της ομάδας θα μπορούσε να θεωρηθεί ως η σχετική εγγύτητα των σημείων στον χώρο. Επομένως οι ομάδες χαρακτηρίζονται ως «σφικτές» (“tight”) όταν όλα τα σημεία της ομάδας είναι συγκεντρωμένα κοντά στο κέντρο ή ως «χαλαρές» (“loose”) όταν τα σημεία της ομάδας είναι διασκορπισμένα σε σχέση με το κέντρο.

Σχήμα (Shape): είναι ο σχηματισμός που έχουν τα σημεία στον χώρο.Υπάρχουν πολλών ειδών σχήματα όπως υπερσφαιρικά, ελλειψοειδή, επιμήκη κ.λ.π. Αν οι ομάδες σχηματίζονται με τέτοιον τρόπο ώστε η έννοια της διαμέτρου ή της ακτίνας να μην έχει νόημα τότε μπορεί να υπολογιστεί η έννοια της συνεκτικότητας (connectivity) των σημείων της ομάδας, η οποία είναι ένα σχετικό μέτρο της απόστασης μεταξύ αυτών.

Διαχωρισμός (Separation): είναι ο βαθμός υπερκάλυψης των ομάδων. Οι ομάδες μπορεί να υπερκαλύπτονται (overlap) ή να κείνται χωριστά στον χώρο. Παραδείγματος χάριν οι ομάδες μπορεί να είναι η μία κοντά στην άλλη χωρίς να έχουν πολύ ξεκάθαρα όρια ή να βρίσκονται σχετικά μακριά μεταξύ τους με ξεκαθαρισμένα όρια.

1.3 Εφαρμογές της Ομαδοποίησης.

Η ομαδοποίηση χρησιμοποιείται, εκτός από τις προαναφερθείσες εφαρμογές στην στατιστική ανάλυση δεδομένων, στην επεξεργασία εικόνας (image possessing), στην ανάκτηση πληροφοριών (data retrieval), στην επιχειρησιακή έρευνα (marketing research) – για την ομαδοποίηση παραδείγματος χάριν πελατών σε ομάδες, στην χημεία (chemistry) – για την ομαδοποίηση στοιχείων σε ομάδες σύμφωνα με τις ιδιότητες τους, στην εξόρυξη δεδομένων (data mining), στην αναγνώριση προτύπων (pattern recognition) και στον καθαρισμό δεδομένων (data cleansing). Δύο από τις σημαντικότερες εφαρμογές της ομαδοποίησης είναι οι παρακάτω:

1. **Μείωση Δεδομένων (Data reduction):** Το πρόβλημα του μεγάλου όγκου των δεδομένων εμφανίζεται σε πολλά ερευνητικά πεδία. Αυτό είναι ένα αρκετά μεγάλο πρόβλημα ιδιαίτερα εάν τα δεδομένα δεν οργανωθούν και ταξινομηθούν σε πιο εύχρηστες ομάδες, οι οποίες θα μπορούσαν να χρησιμοποιηθούν ως μονάδες. Οι τεχνικές της ομαδοποίησης μπορούν να χρησιμοποιηθούν για την μείωση των δεδομένων. Όλη η προσπάθεια λοιπόν είναι στο να μειώσουμε την πληροφορία που μας παρέχουν τα πλήθους N δεδομένα, στην πληροφορία που μας δίνουν τα $g < N$ δεδομένα χωρίς σημαντικές απώλειες πληροφορίας. Με άλλα λόγια, αναζητούμε εκείνη την απλοποίηση των δεδομένων που θα μας αποφέρει την μικρότερη απώλεια πληροφορίας.

2. **Πρόβλεψη βασισμένη σε ομάδες.** Στην περίπτωση αυτή η ομαδοποίηση εφαρμόζεται σε ένα σύνολο προτύπων και κάθε ομάδα χαρακτηρίζεται από τα χαρακτηριστικά των προτύπων τα οποία ανήκουν σε κάθε ομάδα. Αν ένα άγνωστο

πρότυπο εμφανιστεί, τότε αποφασίζεται σε ποια ομάδα είναι πιο πιθανό να ανήκει εξετάζοντας την ομοιότητα του με κάποιον προκαθορισμένο αντιπρόσωπο της ομάδας. Έτσι το πρότυπο αυτό αναγνωρίζεται και χαρακτηρίζεται σύμφωνα με τα χαρακτηριστικά της ομάδας. Ας υποθέσουμε πως σε ένα σύνολο από ασθενείς, μολυσμένους από την ίδια αρρώστια, εφαρμόζουμε την ομαδοποίηση. Το αποτέλεσμα θα είναι ένα σύνολο ομάδων ασθενών που θα χαρακτηρίζονται από τις αντιδράσεις των ασθενών σε συγκεκριμένες φαρμακευτικές αγωγές. Αν τώρα ένας νέος ασθενής εμφανιστεί τότε θα πρέπει να αναγνωριστεί η πιο πιθανή ομάδα στην οποία θα μπορούσε να ανήκει και με αυτόν τον τρόπο αποφασίζεται ποια φαρμακευτική αγωγή θα του χορηγηθεί.

1.4 Ομαδοποίηση στην εξόρυξη δεδομένων

Το συνεχές αυξανόμενο πλήθος των δεδομένων και η συνεπακόλουθη αύξηση του μεγέθους των βάσεων δεδομένων, όπου αυτά είναι αποθηκευμένα, κάνουν επιτατική την ανάγκη για ανάπτυξη νέων εργαλείων και νέων τεχνικών για την ανάλυση και την εξαγωγή (εξόρυξη) γνώσης από αυτά. Η εποχή της πληροφορίας στην οποία ζούμε και το τεράστιο μέγεθος των αποθηκευμένων πληροφοριών που συνεπάγεται, κάνουν τις παραδοσιακές στατιστικές τεχνικές ανάλυσης των δεδομένων ανεπαρκείς και χρονοβόρες και για αυτό νέες τεχνικές για την ανακάλυψη γνώσης θα πρέπει να βοηθούν τους ανθρώπους στην ανάλυση τεράστιων ποσοτήτων πληροφορίας. Αυτές οι τεχνικές και τα εργαλεία είναι το αντικείμενο ενός νέου πεδίου που καλείται ανακάλυψη γνώσης σε βάσεις δεδομένων (knowledge discovery in databases - KDD).

Ένας ορισμός για το τι είναι ανακάλυψη γνώσης σε βάσεις δεδομένων που προτάθηκε από τους Fayyad, Piatesky – Shapino, Smyth και Uthurusamy το 1996 είναι ο εξής:

> **Ανακάλυψη Γνώσης σε Βάσεις Δεδομένων (Knowledge Discovery in Databases)** είναι μια μη – τετριμμένη διαδικασία αναγνώρισης έγκυρων νέων, ενδεχομένως χρήσιμων και τελικά κατανοήσιμων προτύπων στα δεδομένα

Οι κυριότερες λειτουργίες στην εξόρυξη δεδομένων είναι η ταξινόμηση (classification) και η συσταδιοποίηση (clustering). Σκοπός της ταξινόμησης είναι η παραγωγή κανόνων από μεγάλες σχεσιακές βάσεις δεδομένων που να μπορούν να ταξινομήσουν καινούργια άγνωστα δεδομένα σε προκαθορισμένες κλάσεις οι οποίες να περιγράφονται από ένα σύνολο χαρακτηριστικών. Η εξαγωγή των κανόνων γίνεται με την χρήση μεθόδων μάθησης με επίβλεψη (supervised learning methods) .

Από τους πιο γνωστούς αλγόριθμους κατάταξης είναι οι ID3, C45, CN2 και CART. Το βασικό μειονέκτημα αυτών των μεθόδων είναι η υπολογιστική δυσκαμψία και η συνεπακόλουθη δαπάνη σε υπολογιστικό χρόνο. Από την άλλη πλευρά όμως, η εξόρυξη δεδομένων απαιτεί την επεξεργασία μεγάλων ποσοτήτων δεδομένων με ικανοποιητική ακρίβεια. Για αυτό και η πρόκληση είναι η ανάπτυξη μεθόδων που θα επεξεργάζονται τεράστια ποσά δεδομένων που υπερβαίνουν κατά πολύ την μνήμη ενός επεξεργαστή, με αποτελεσματικό και χρονικά ικανοποιητικό τρόπο. Σ' αυτήν την κατεύθυνση έχουν αναπτυχθεί σύγχρονες μέθοδοι όπως η στρατηγική meta – classifying ή meta – learning .

Η δεύτερη βασική λειτουργία στην εξόρυξη δεδομένων είναι η ομαδοποίηση των εγγραφών μια βάσης δεδομένων σε υποομάδες-συστάδες (clustering). Η συσταδιοποίηση είναι μια περιγραφική λειτουργία που σκοπό έχει την ανίχνευση ενός πεπερασμένου πλήθους ομάδων ή κατηγοριών (clusters) που περιέχονται στα δεδομένα .

Θα πρέπει να αναφέρουμε εδώ ότι οι αλγόριθμοι ομαδοποίησης διαχειρίζονται μεγάλο πλήθος δεδομένων και απαιτούν έναν αρκετά μεγάλο αριθμό υπολογισμών. Συνεπώς η πολυπλοκότητά τους εξαρτάται από το πλήθος των δεδομένων που επεξεργάζεται ο κάθε αλγόριθμος. Από την άλλη, το τεράστιο μέγεθος των δεδομένων που αποθηκεύονται στις βάσεις δεδομένων ωθεί σήμερα το ερευνητικό ενδιαφέρον κυρίως σε αλγορίθμους ομαδοποίησης, που μπορούν αν χειριστούν δεδομένα πολύ μεγαλύτερα από την κύρια μνήμη ενός επεξεργαστή.

ΚΕΦΑΛΑΙΟ 2: ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

2.1 Εισαγωγή και ιστορική αναδρομή

Στα πλαίσια της επιστημονικής περιοχής Εξόρυξης Δεδομένων (data mining) ή αλλιώς ¨Ανακάλυψης Γνώσης σε Βάσεις Δεδομένων ¨ (knowledge discovery in databases) η μηχανική μάθηση αποτέλεσε κινητήριο μοχλό για την ανάπτυξη της άνω περιοχής και αναπόσπαστο τμήμα της. Η μηχανική μάθηση (machine learning) είναι ένα υποπεδίο της τεχνητής νοημοσύνης που ασχολείται με τη σχεδίαση αυτόματων διαδικασιών ικανών να μαθαίνουν από παραδείγματα. Η εκμάθηση από παραδείγματα αφορά στην παραγωγή μιας λογικής περιγραφής των αναγκαίων και ικανών συνθηκών που αντιστοιχούν σε μια κλάση αντικειμένων, π.χ. σε έναν κανόνα δοθείσης μιας γλώσσας αναπαράστασης. Μία σημαντική φροντίδα είναι η εύρεση των αναγκαίων συμβιβασμών ανάμεσα στην πολυπλοκότητα των κανόνων και στην προσαρμογή των δεδομένων, έτσι ώστε να αποφευχθεί η υπερβολική προσαρμογή (overfitting) και να καταστεί δυνατή η ερμηνεία των αποτελεσμάτων.

Έτσι η μηχανική μάθηση έχει ως σκοπό τη δημιουργία μηχανών ικανών να μαθαίνουν, δηλαδή ικανών να βελτιώνουν την απόδοση τους σε κάποιους τομείς μέσω της αξιοποίησης προηγούμενης γνώσης και εμπειρίας. Αν και απέχουμε πάρα πολύ από τη δημιουργία μηχανών που να μαθαίνουν τόσο καλά και τόσο μεγάλη ποικιλία πραγμάτων όσο και ο άνθρωπος, έχουν αναπτυχθεί αλγόριθμοι για συγκεκριμένες περιοχές μάθησης, οι οποίοι έχουν επιτρέψει την εμφάνιση εμπορικών εφαρμογών με σημαντική επιτυχία. Για προβλήματα όπως η αναγνώριση φωνής (speech recognition) και η εξόρυξη γνώσης (data mining) από μεγάλες βάσεις δεδομένων, η χρήση αλγορίθμων μηχανικής μάθησης αποτελεί πλέον ρουτίνα, ενώ έχουν σχεδιαστεί προγράμματα ικανά από το να μαθαίνουν να παίζουν τάβλι σε επίπεδο ανάλογο με των παγκόσμιων πρωταθλητών [Tesauro 1995] μέχρι να μαθαίνουν να οδηγούν αυτόνομα οχήματα σε δημόσιες λεωφόρους [Pomerlau 1989]. Επίσης, έχουν δημοσιευθεί θεωρητικά αποτελέσματα σχετικά με τις θεμελιώδεις σχέσεις μεταξύ του όγκου της εμπειρίας που είναι διαθέσιμος, του αριθμού των υπό θεώρηση υποθέσεων και του προβλεπόμενου λάθους στην επιλεχθείσα υπόθεση, ενώ

έχουν αρχίσει να εμφανίζονται μοντέλα μάθησης για τον άνθρωπο και τα ζώα και να συσχετίζονται με τους αλγόριθμους που έχουν αναπτυχθεί για υπολογιστές.

Τα τελευταία χρόνια πληθαίνουν οι εφαρμογές της μηχανικής μάθησης. Τεχνικές όπως οι γενετικοί αλγόριθμοι, νευρωνικά δίκτυα, η ασαφής λογική και άλλες τεχνικές από τους τομείς της επεξεργασίας σημάτων και από την αναγνώριση προτύπων αντιμετωπίζουν με επιτυχία τη δύσκολη φύση πολλών προβλημάτων.

Διάφοροι ορισμοί για την μηχανική μάθηση είναι οι παρακάτω:

- **Carbonell(1987):** ¨...η μελέτη υπολογιστικών μεθόδων για την απόκτηση νέας γνώσης, νέων δεξιοτήτων και νέων τρόπων οργάνωσης της υπάρχουσας γνώση¨

- **Mitchell(1997):**¨...Ένα πρόγραμμα υπολογιστή θεωρείται ότι μαθαίνει από την εμπειρία E σε σχέση με μια κατηγορία εργασιών T και μια μετρική απόδοσης P, αν απόδοσή του σε εργασίες της T, όπως μετρούνται από την P, βελτιώνονται με την εμπειρία E¨

- **Witten & Frank(2000)**:¨...Κάτι μαθαίνει όταν αλλάζει η συμπεριφορά του κατά τέτοιο τρόπο ώστε να αποδίδει καλύτερα στο μέλλον¨

Πολλές φορές, το πρόβλημα της βελτίωσης της απόδοσης σε μια εργασία μπορεί να αναχθεί στο πρόβλημα της προσέγγισης μιας συνάρτησης-στόχου (target function) ή αντικειμενικής συνάρτησης (object function), γεγονός που απλοποιεί τους περαιτέρω συλλογισμούς. Σε κάποια προβλήματα η συνάρτηση-στόχος είναι προφανής, ενώ σε άλλα δεν είναι και η επιλογή της αποτελεί καίρια σχεδιαστική επιλογή. Πεδίο ορισμού αυτής της συνάρτησης είναι ένα σύνολο οντοτήτων σε κάποια δεδομένη αναπαράσταση, η οποία αποτελεί το χώρο στιγμιοτύπων (instance space) του προβλήματος. Η πλέον συνηθισμένη αναπαράσταση είναι αυτή που παρέχει το μοντέλο του διανυσματικού το χώρο στιγμιοτύπων χώρου (vector space model, [Salton & McGill,1983]). Σύμφωνα με αυτό το μοντέλο, οι οντότητες αναπαρίστανται ως διανύσματα, τα στοιχεία των οποίων αναπαριστούν τα

χαρακτηριστικά (features ή attributes) της οντότητας που έχουν επιλεγεί ως σχετικά για το συγκεκριμένο πρόβλημα. Τα χαρακτηριστικά μπορούν να παίρνουν συμβολικές ή αριθμητικές τιμές. Για παράδειγμα, αν οι οντότητες αντιπροσωπεύουν μανιτάρια και το ζητούμενο είναι το αν αυτά είναι δηλητηριώδη, το διάνυσμα που αντιστοιχεί σε κάθε μανιτάρι είναι δυνατόν να περιλαμβάνει χαρακτηριστικά όπως την οσμή του, την προέλευσή του, το βάρος του κ.α. Οι τιμές της συνάρτησης-στόχου μπορεί να είναι πρακτικά οτιδήποτε: Αριθμητικές ή συμβολικές, διακριτές ή συνεχείς, βαθμωτές ή διανυσματικές, κ.ο.κ. Ακόμα είναι δυνατόν να έχουν φυσική σημασία ή να μην έχουν (π.χ. ένας αριθμός που εκτιμά πόσο καλή είναι η κατάσταση σε μια σκακιέρα για κάθε παίκτη).

Οι δύο στόχοι των προβλημάτων με τα οποία ασχολείται η μηχανική μάθηση είναι:
1) Η πρόβλεψη
2) Η ανακάλυψη γνώσης

Η πρόβλεψη εμπλέκει κάποιες μεταβλητές ή κάποια πεδία της βάσης δεδομένων έτσι ώστε να προβλεφθούν άγνωστες ή μελλοντικές τιμές ή και άλλες μεταβλητές ενδιαφέροντος. Πάνω στη πρόβλεψη στηρίζονται εφαρμογές σε διάφορα πεδία όπως αυτό της ιατρικής για πρόβλεψη καρκίνου, για χρηματοοικονομικές προβλέψεις, ακόμα και στην μετεωρολογία και σε πολλά άλλα..

Η ανακάλυψη γνώσης από δεδομένα (KDD) είναι μία σύνθετη διαδικασία για τον προσδιορισμό έγκυρων, νέων, χρήσιμων και κατανοητών σχέσεων-προτύπων σε δεδομένα. Αποτελεί μια σημαντική εφαρμογή σε πραγματικές συνθήκες και σε μεγάλη κλίμακα των ερευνητικών αποτελεσμάτων της Στατιστικής, των Βάσεων δεδομένων και της Μηχανικής μάθησης Η εξόρυξη γνώσης είναι η χρήση αλγορίθμων και τεχνικών για την εξαγωγή προτύπων κατά την διάρκεια της διαδικασίας KDD.

2.2 Μεθοδολογίες της μηχανικής μάθησης

Κύριες εργασίες της διαδικασίας εξόρυξης γνώσης είναι η εύρεση συσχετίσεων μεταξύ των δεδομένων (κανόνες συσχέτισης), η κατηγοριοποίηση σε

προκαθορισμένες κλάσεις (δέντρα απόφασης, νευρωνικά δίκτυα, κατηγοριοποίηση bayesian) και η συσταδιοποίηση-ομαδοποίηση (ιεραρχικοί, διαμεριστικοί, με βάση την ποιότητα).

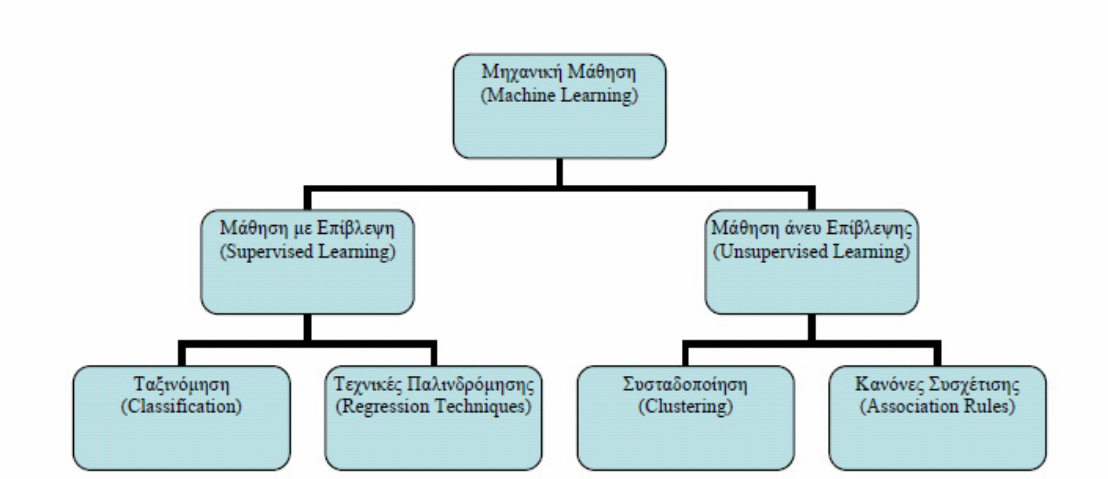
- **Ταξινόμηση (classification)**

  Ανάπτυξη ενός μοντέλου πρόβλεψης της κλάσης των στιγμιοτύπων ενός προβλήματος. Το μοντέλο αυτό χτίζεται με βάση ένα σύνολο δεδομένων εκπαίδευσης .Η απόδοσή του αξιολογείται με βάση ένα σύνολο δεδομένων ελέγχου

- **Συσταδιοποίηση (clustering)**

  Διαχωρισμός των δεδομένων σε ομάδες-συστάδες έτσι ώστε για κάθε εγγραφή που περιλαμβάνει μια συστάδα να είναι μεγαλύτερη από την ομοιότητά της με οποιαδήποτε εγγραφή από άλλες συστάδες. Στην μη εποπτευόμενη μάθηση δεν γνωρίζουμε την κλάση στην οποία ανήκουν τα δεδομένα εκπαίδευσης. Μας δίνεται ένα σύνολο μετρήσεων και παρατηρήσεων με στόχο να ανακαλύψουμε κλάσεις ή ομάδες μέσα στα δεδομένα.

Η μηχανική μάθηση περιλαμβάνει μία ιεραρχία μεθοδολογιών, όπως απεικονίζονται στην παρακάτω εικόνα.



Εικόνα 2.1 Οι κατηγορίες της μηχανικής μάθησης

Οι τεχνικές της μηχανικής μάθησης διαιρούνται σε δύο κατηγορίες: την εκμάθηση με επίβλεψη και την εκμάθηση χωρίς επίβλεψη. Στην πρώτη περίπτωση σε κάθε παράδειγμα είναι γνωστή η τιμή ή η κατηγορία στην οποία ανήκει η παρατήρηση y, και με βάση ένα σύνολο από ταξινομημένα στιγμιότυπα σκοπός είναι η εύρεση ενός προτύπου-μοντέλου με το οποίο θα μπορέσουμε να ταξινομήσουμε ένα άλλο σετ στιγμιοτύπων που είναι αταξινόμητα. Η πρόβλεψη της αυριανής μέσης θερμοκρασίας με δεδομένες τις μετρήσεις της μέσης ημερήσιας θερμοκρασίας είναι ένα παράδειγμα προβλήματος με επίβλεψη, γιατί στα παραδείγματα του παρελθόντος υπάρχει η ακριβής τιμή της μετρούμενης θερμοκρασίας. Στην περίπτωση που τιμή ή η κατηγορία στην οποία ανήκει η παρατήρηση είναι άγνωστη, η μηχανική μάθηση επιχειρεί να την προσεγγίσει κατηγοριοποιώντας τα παραδείγματα σε ομάδες που θα έχουν ομοιότητα με βάση τα χαρακτηριστικά τους ή δημιουργεί κανόνες συσχέτισης. Το πρόβλημα της εύρεσης της κατηγορίας ενός νέου προϊόντος σε σχέση με τα υπόλοιπα προϊόντα σε ένα πολυκατάστημα με βάση τα χαρακτηριστικά του και τις πωλήσεις του είναι ένα συνηθισμένο παράδειγμα προβλήματος χωρίς επίβλεψη.

Το ζητούμενο στην περίπτωση της μάθησης με επίβλεψη είναι η κατασκευή ενός μοντέλου-προτύπου που αναπαριστά τη γνώση που παρέχεται μέσω της εμπειρίας Ε και το οποίο στη συνέχεια χρησιμοποιείται για την αξιολόγηση νέων αταξινόμητων στιγμιοτύπων. Κατά κανόνα, οι προβλέψεις του προκύπτωντος μοντέλου θα επαληθεύονται για την πλειοψηφία από τα στοιχεία που περιλαμβάνονται στην Ε, τα οποία λέγονται και στιγμιότυπα εκπαίδευσης.

Η εκμάθηση με επίβλεψη αφορά σε δύο περαιτέρω κατηγορίες προβλημάτων, την παλινδρόμηση και την ταξινόμηση. Στην πρώτη η μεταβλητή y είναι μία συνεχής μεταβλητή ενώ στη δεύτερη μία διακριτή κατηγορική μεταβλητή. Το παράδειγμα της θερμοκρασίας είναι πρόβλημα παλινδρόμησης όταν η θερμοκρασία μετράται σαν πραγματικός συνεχής αριθμός, ενώ είναι πρόβλημα κατάταξης αν μετράται σαν θερμοκρασιακή ζώνη (π.χ. κρύο/ζέστη).

Τέλος τα προβλήματα χωρίς επίβλεψη χωρίζονται σε ομαδοποίησης και ανακάλυψης κανόνων. Η μεθοδολογία της ομαδοποίησης επιχειρεί να ξεχωρίσει τα δεδομένα σε άτυπες ομάδες, ενώ η ανακάλυψη κανόνων διερευνά τα δεδομένα για να

ανακαλύψει κανόνες που χαρακτηρίζουν τις ιδιότητες των παραδειγμάτων. Αν για παράδειγμα είχαμε στα χέρια μας ένα σύνολο δεδομένων που περιλάμβανε τις αγορές προϊόντων σε ένα κατάστημα από κάθε πελάτη, τότε ένας αλγόριθμος ανακάλυψης κανόνων θα συμπέρανε κανόνες του τύπου '34% των πελατών που αγόρασαν ψυγεία, αγόρασαν και ηλεκτρική σκούπα', ή '45% των πελατών που αγόρασαν σοκολατάκια ΚΑΙ αγόρασαν επίσης σαμπάνια, πλήρωσαν με πιστωτική κάρτα'.

2.3 Η διαδικασία των δεδομένων της Μηχανικής μάθησης

Τα στάδια τα οποία ακολουθούνται στην διαδικασία της μηχανικής μάθησης είναι τα παρακάτω:

1)Επιλογή Δεδομένων

Δημιουργείται το σύνολο δεδομένων στο οποίο θα εφαρμοστεί η αναζήτηση (training data set selection) με επιλογή στοιχείων (πινάκων, πεδίων) από σχεσιακές βάσεις δεδομένων.

2)Προεπεξεργασία (preprocessing) Δεδομένων

Αντιμετωπίζονται περιπτώσεις ελλιπών δεδομένων, πεδίων με τιμές που ουσιαστικά τα καθιστούν κενά και πεδίων με τιμές που υπονοούν κάτι άλλο.

3)Μετασχηματισμός δεδομένων

Τα δεδομένα μετασχηματίζονται ώστε να διευκολύνουν την ανακάλυψη γνώσης. Τέτοιοι μετασχηματισμοί μπορεί να περιλαμβάνουν για παράδειγμα:

- Τη μείωση του αριθμού των υπό-εξέταση χαρακτηριστικών με επιλογή ορισμένων από αυτά
- Την ομοιόμορφη κωδικοποίηση της ποιοτικά ίδιας πληροφορίας
- Την μετατροπή συνεχόμενων αριθμητικών τιμών σε διακριτές

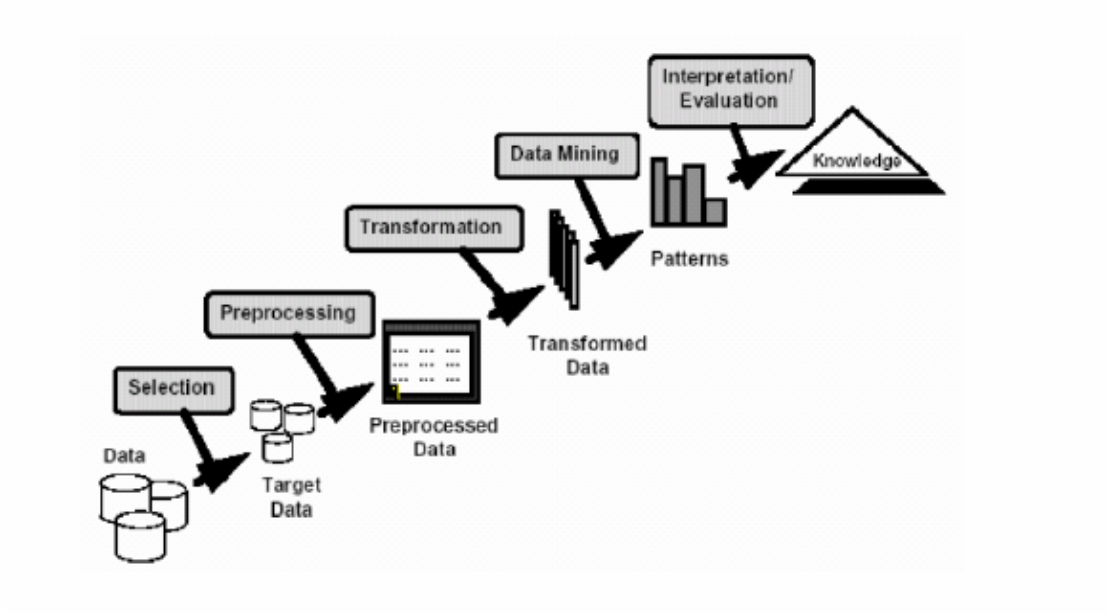
4)Επιλογή αλγόριθμου και εφαρμογή του

Καθορίζεται τι είδους γνώση θα αναζητηθεί, κάτι που έμμεσα προσδιορίζει και την κατηγορία αλγορίθμου που θα χρησιμοποιηθεί. Είναι ένα καθαρά υπολογιστικό

στάδιο, στο οποίο γίνεται η ουσιαστική αναζήτηση της γνώσης στα δεδομένα. Και περιγράφεται με τον όρο εξόρυξη σε δεδομένα (data mining).

5)Ερμηνεία (interpretation)-αξιολόγηση (evaluation)

Γίνεται ερμηνεία και αξιολόγηση των ευρεθέντων προτύπων, πιθανώς με υποβοήθηση γραφικών απεικονίσεων των προτύπων ή και των δεδομένων που περιγράφονται από το πρότυπο.



Εικόνα 2.2 Τα στάδια της ανακάλυψης γνώσης

Υπάρχει μια σημαντική διαφορά ανάμεσα στους όρους KDD και Data mining. Ο όρος KDD αναφέρεται στην ολική διαδικασία ανακάλυψης γνώσης από τα δεδομένα, σε αντίθεση με τον όρο Data Mining ο οποίος αναφέρεται κυρίως στην εφαρμογή των αλγορίθμων για την εξαγωγή της γνώσης από τα δεδομένα χωρίς να αναφέρεται και στα άλλα στάδια της KDD διαδικασίας. Ένας ορισμός για το τι είναι εξόρυξη δεδομένων δίνεται παρακάτω.

> **Εξόρυξη Δεδομένων (Data Mining)** είναι ένα βήμα της KDD διαδικασίας που αναφέρεται στην ανακάλυψη και εφαρμογή αλγορίθμων για την ανάλυση δεδομένων, οι οποίοι κάτω από υπολογιστικά αποδεκτούς περιορισμούς,

παράγουν μια συγκεκριμένη απαρίθμηση (enumeration) των προτύπων στα δεδομένα

Πρέπει να τονιστεί ότι η επιλογή των κατάλληλων αλγορίθμων περιλαμβάνει αποφάσεις για το ποιο μοντέλο και ποιες παράμετροι είναι οι πλέον κατάλληλες να χρησιμοποιηθούν και θα πρέπει να μπορούν να χειριστούν σετ δεδομένων μεγάλου μεγέθους. Έτσι αυτό που αναζητείται είναι ποια από τις μεθόδους θα έχει τις μεγαλύτερες πιθανότητες να επιτύχει την σωστή ταξινόμηση ενός αταξινόμητου στιγμιοτύπου.

Ωστόσο η επίτευξη ενός επιτυχημένου data mining προϋποθέτει περισσότερα από την απλή επιλογή ενός αλγορίθμου και την εφαρμογή του στα δεδομένα που διαθέτουμε. Η τεχνητή νοημοσύνη των αλγορίθμων πολλές φορές δοκιμάζεται από την ποιότητα των δεδομένων που εισάγονται προς επεξεργασία και είναι δεδομένος ο κίνδυνος κάποια ελλατωματικά δεδομένα να τον οδηγήσουν σε λάθη. Όπως γίνεται κατανοητό λοιπόν, πέραν της επιλογής του αλγορίθμου και των παραμέτρων που θα οριστούν σ'αυτόν, είναι και άλλες ενέργειες που μπορούν να βελτιώσουν ουσιαστικά την εφαρμογή τεχνικών μηχανικής μάθησης σε ένα πρακτικό πρόβλημα εξόρυξης γνώσης.

Έτσι σημαντική υπόθεση για την επιτυχία μίας εφαρμογής μηχανικής μάθησης είναι η σωστή εκμετάλλευση των παρεχόμενων δεδομένων. Όλοι οι αλγόριθμοι της μηχανικής μάθησης απαιτούν από αυτά να είναι ενοποιημένα σε ένα αρχείο εισόδου ακολουθώντας συγκεκριμένη τυποποίηση. Το ζητούμενο είναι τα παραδείγματα να αντιπροσωπεύονται από τις γραμμές του αρχείου και οι ιδιότητες από τις στήλες. Πολλές δυσκολίες προκύπτουν προς την κατεύθυνση της ενοποίησης των δεδομένων. Αυτά μπορούν να υπάρχουν σε διάφορες μορφές και σε διαφορετικά αποθηκευτικά μέσα. Μπορεί να είναι κατανεμημένες ή συγκεντρωμένες, σε αρχεία κειμένου, σε βάσεις δεδομένων, σε λογιστικά φύλλα, ταξινομημένες ή αταξινόμητες, χρονικά διατεταγμένες ή χωρικά διεσπαρμένες.

ΚΕΦΑΛΑΙΟ 3: ΕΠΕΞΕΡΓΑΣΙΑ ΔΕΔΟΜΕΝΩΝ ΜΕ ΤΟ ΛΟΓΙΣΜΙΚΟ ΜΑΘΗΣΗΣ WEKA

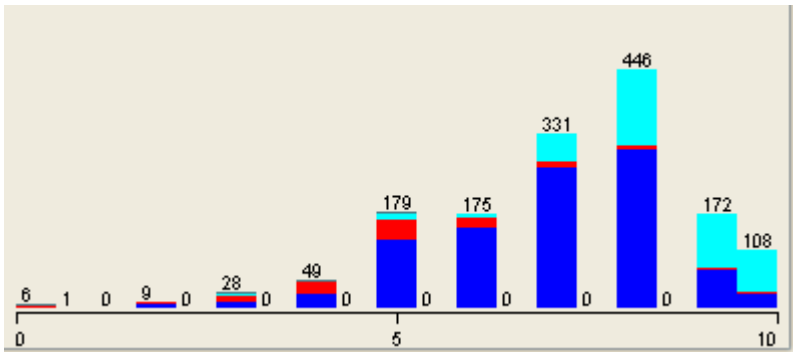
3.1 Παρουσίαση σετ δεδομένων

Στην παρούσα εργασία θα χρησιμοποιηθεί ένα σετ δεδομένων το οποίο έχει εξαχθεί από ένα ερωτηματολόγιο που αφορά στοιχεία για το μετρό του Παρισίου. Τα δεδομένα τα οποία και διαθέτουμε είναι δεδομένα ικανοποίησης, δηλαδή βαθμολογίες πολιτών για διάφορες στατιστικές κατηγορίες που αφορούν το μετρό ανάλογα με το πόσο ικανοποιημένοι ή όχι είναι με την συγκεκριμένη κατηγορία. Για κάθε ομάδα κατηγοριών υπάρχει και μία τελική κατηγορία η οποία θα αποτελεί και την κλάση μας και η οποία προκύπτει ανάλογα με τις τιμές των προηγούμενων κατηγοριών οι οποίες αποτελούν τα χαρακτηριστικά των στιγμιοτύπων μας.

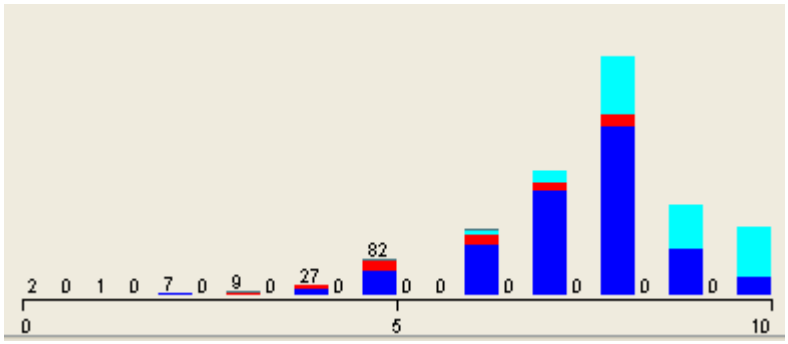
Η βάση δεδομένων την οποία θα χρησιμοποιήσουμε περιλαμβάνει 1504 στιγμιότυπα από 10 χαρακτηριστικά συνολικά, οι συνδυασμοί των οποίων ανά τετράδες και τριάδες μας δίνουν τις τιμές (κλάσεις) οι οποίες εκφράζουν την ικανοποίηση για τρεις διαφορετικές κατηγορίες. Τα χαρακτηριστικά παίρνουν τιμές από 0 έως 10 που εκφράζουν τον βαθμό ικανοποίησης και οι κλάσεις τιμές από 1 έως 4 που αντιστοιχούν στις ονομαστικές τιμές bad, not good, enough, good. Τέλος χρησιμοποιώντας τις τιμές των κλάσεων αυτών σαν χαρακτηριστικά προκύπτει η τελική ικανοποίηση για κάθε άτομο που περιέχει τιμές (κλάσεις) από 1 έως 10. Έτσι θα έχουμε την δυνατότητα να πειραματιστούμε με στιγμιότυπα τεσσάρων και δέκα κλάσεων.

Θα δημιουργήσουμε 4 σετ υποομάδων δεδομένων. Παρακάτω θα παρουσιάσουμε αναλυτικά όλα τα σετ και τα χαρακτηριστικά τους καθώς και τα στατιστικά στοιχεία που προκύπτουν μέσω του εργαλείου WEKA. Το πρώτο σετ περιλαμβάνει τα εξής χαρακτηριστικά:

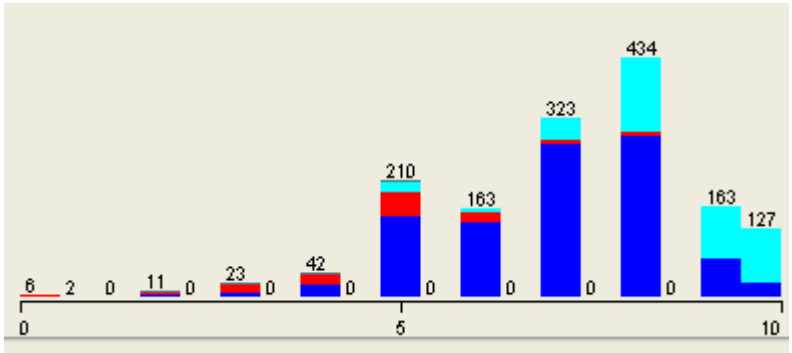
Latency: εκφράζει την καθυστέρηση των δρομολογίων ως προς τον χρόνο αναχώρησης και άφιξης των τραίνων.



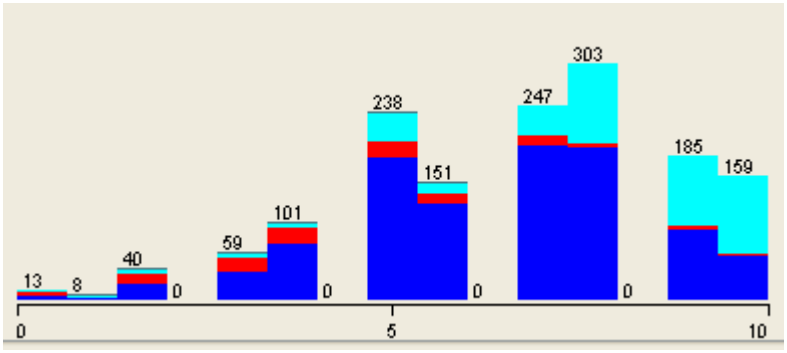
Circulation: κατά πόσο είναι ικανοποιητική η συχνότητα των δρομολογίων



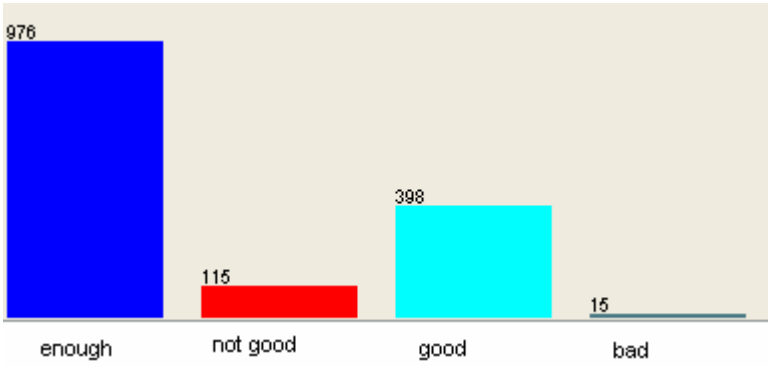
Punctuality: εκφράζει την ακρίβεια των δρομολογίων



Breakdown: ικανοποίηση ως προς τα προβλήματα που οφείλονται σε μηχανικές βλάβες

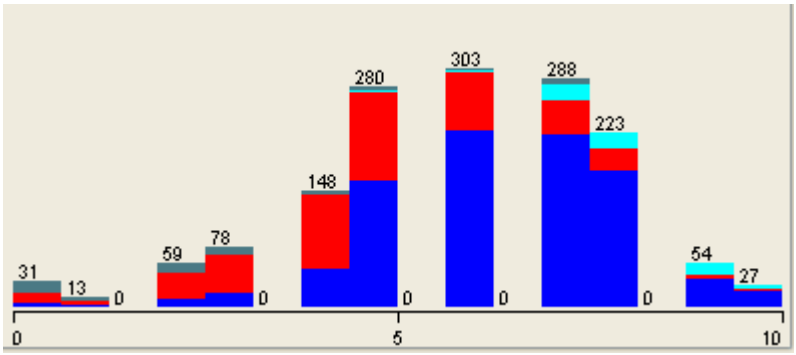


GlobalA:Το χαρακτηριστικό κλάση που προσδιορίζει τον βαθμό ικανοποίησης ως προς την λειτουργία στο μετρό με χαρακτηριστικά τα 4 προηγούμενα.

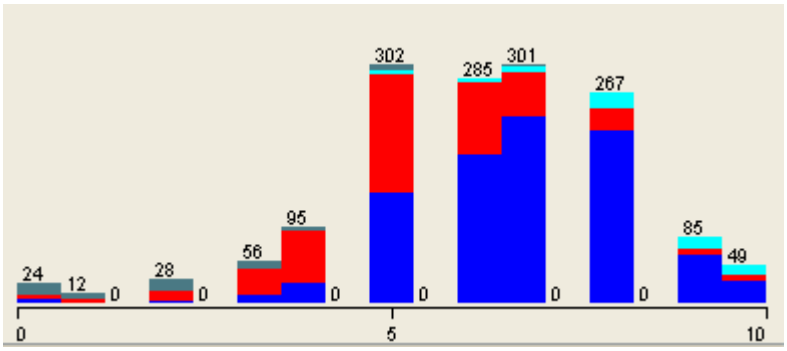


Το δεύτερο σετ μας δίνει πληροφορία για τις συνθήκες που επικρατούν στο σταθμό.

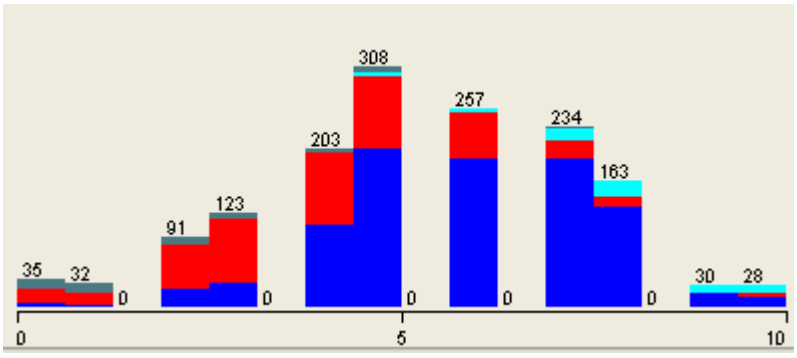
Cleanliness : δηλώνει κατά πόσον είναι καθαρός ο περιβάλλοντας χώρος



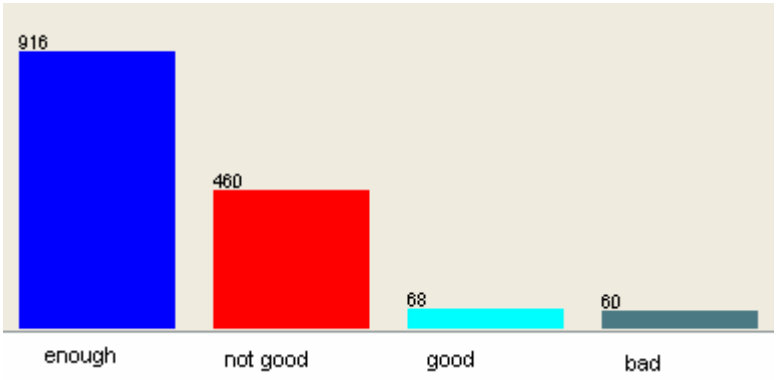
Safety: ικανοποίηση ως προς την ασφάλεια μέσα στο μετρό



Multitude: κατά πόσον μεγάλη είναι η κίνηση του πλήθους

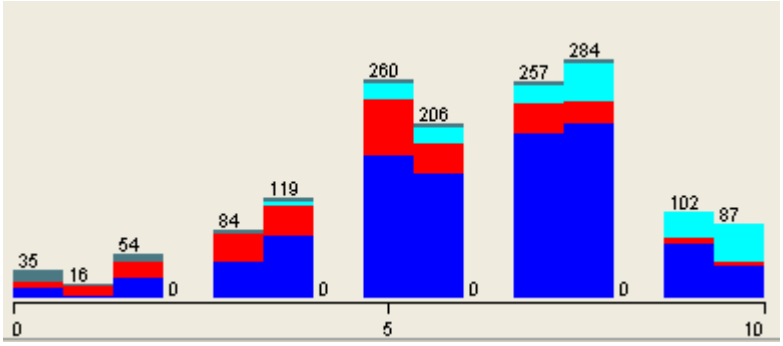


GlobalB : Το χαρακτηριστικό κλάση που μας δίνει τον βαθμό ικανοποίησης ως προς τις συνθήκες.

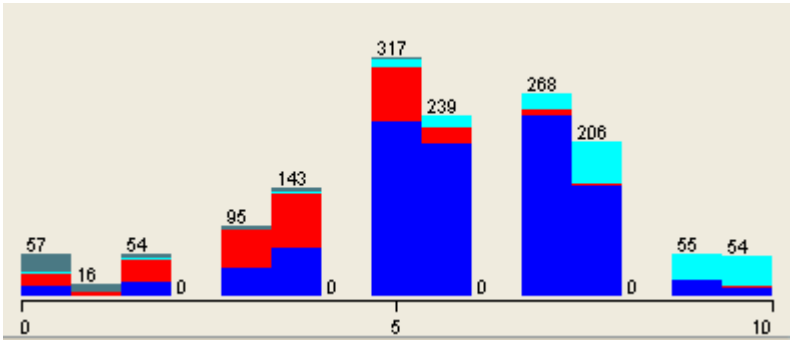


Το τρίτο σετ μας δίνει πληροφορία για την εξυπηρέτηση των πολιτών από το προσωπικό στο μετρό

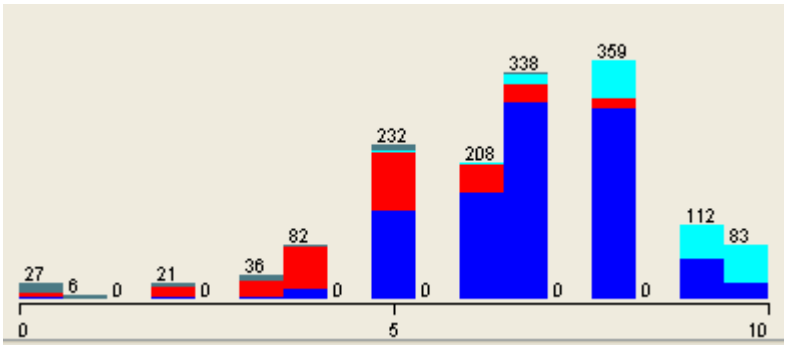
Information: ικανοποίηση για τις πληροφορίες που παρέχει το προσωπικό



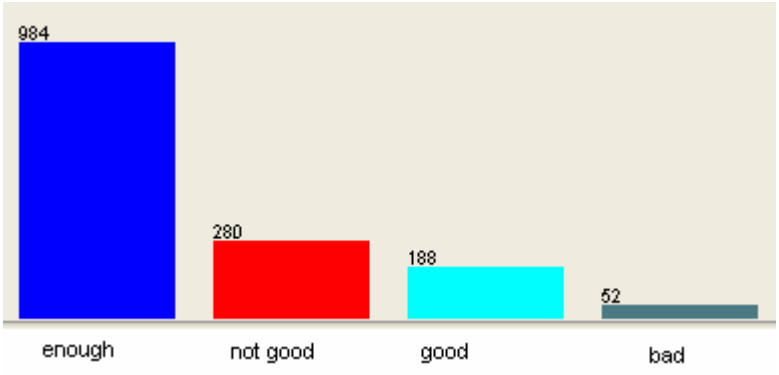
Personnel presence: ικανοποίηση από την παρουσία προσωπικού



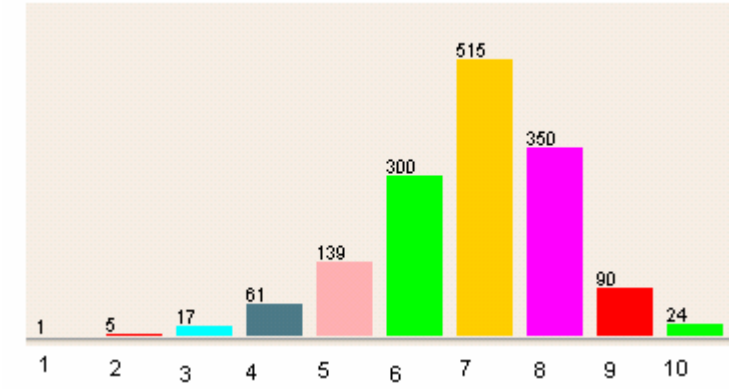
Personnel kindness: ευγένεια του προσωπικού



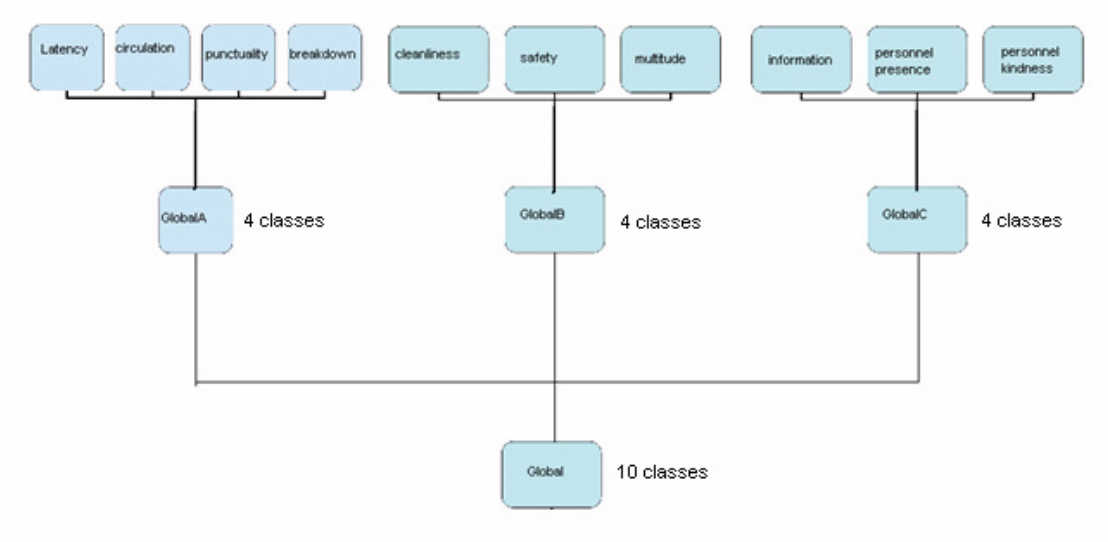
GlobalC: Το χαρακτηριστικό κλάση που προσδιορίζει την ικανοποίηση για την εξυπηρέτηση από το προσωπικό



Τέλος τα GlobalA, GlobalB, GlobalB χρησιμοποιούνται σαν χαρακτηριστικά τα οποία προσδιορίζουν το χαρακτηριστικό κλάση Global και το οποίο θα αποτελείται από 10 διαφορετικές κλάσεις από 1 εώς 10 και δείχνουν την συνολική ικανοποίηση του κάθε πολίτη.



Παρακάτω φαίνεται και ένα διάγραμμα με την ιεραρχική δομή των δεδομένων μας



Εικόνα 3.1 Δομή του σετ δεδομένων

3.2 Εισαγωγή στο λογισμικό μάθησης WEKA

Το πρώτο εργαλείο το οποίο θα χρησιμοποιήσουμε για την εξαγωγή αποτελεσμάτων ως προς την ταξινόμηση των δεδομένων μας θα είναι το weka. Το WEKA(Waikato Enviroment for Knowledge Analysis) είναι ένα περιβάλλον ανάπτυξης εφαρμογών μηχανικής μάθησης και εξόρυξης γνώσης, το οποίο αναπτύχθηκε από ερευνητές στο πανεπιστήμιο του Waikato στην Νέα Ζηλανδία. Επίσης είναι γραμμένο σε Java, τρέχει σχεδόν σε κάθε πλατφόρμα και έχει ελεγχθεί στα λειτουργικά συστήματα των Windows, Linux και Macintosh. Χρησιμοποιείται για έρευνα, εκπαίδευση και άλλες εφαρμογές και διαθέτει μια συλλογή από αλγορίθμους machine learning που μπορούν να κληθούν αμέσως από το περιβάλλον διεπαφής (GUI) ή από το δικό μας κώδικα Java.

Το WEKA παρέχει εκτενή υποστήριξη για ολόκληρη την διαδικασία data mining συμπεριλαμβανομένης την προετοιμασία των δεδομένων εισόδου, την εκτίμηση στατιστικών σχημάτων εκμάθησης και την εικονική απεικόνιση των δεδομένων εισόδου καθώς και το αποτέλεσμα της εκάστοτε εκμάθησης. Ως πρόγραμμα το οποίο διαθέτει

μια μεγάλη ποικιλία αλγορίθμων εκμάθησης περιλαμβάνει και ένα ευρύ φάσμα από εργαλεία προεπεξεργασίας δεδομένων. Σε αυτό το περιεκτικό εργαλείο υπάρχει πρόσβαση μέσω μιας κοινής διεπαφής έτσι ώστε οι χρήστες του να μπορούν να συγκρίνουν διαφορετικές μεθόδους και να προσδιορίσουν ποια από αυτές είναι η πιο κατάλληλη για το εκάστοτε πρόβλημα.

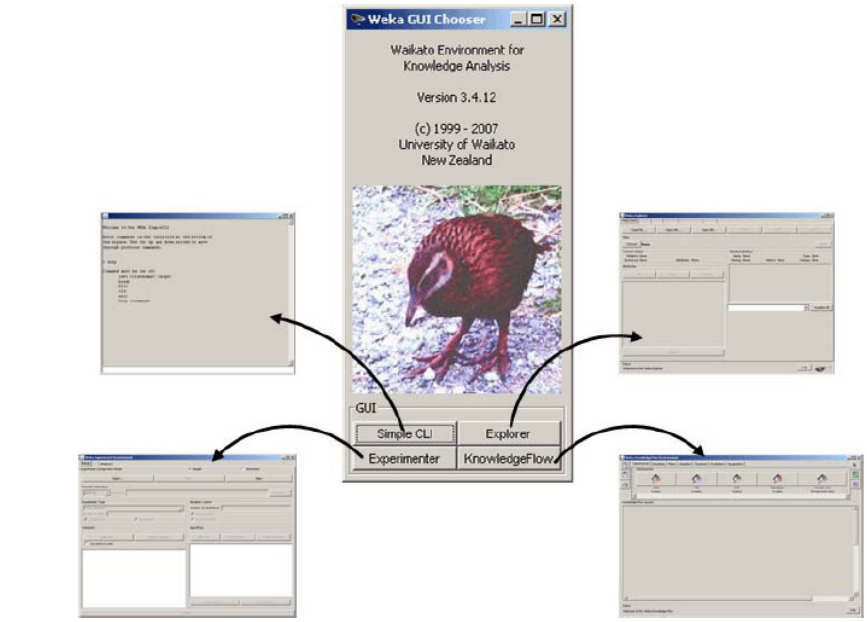
Το περιβάλλον διεπαφής (GUI) χρησιμοποιείται για να αρχίσει κάποιος να δουλέψει πάνω στο WEKA και περιέχει τις ακόλουθες 4 επιλογές:

1)Simple CLI: παρέχει γραμμή εντολών για τις ρουτίνες του weka και είναι περισσότερο για λειτουργικά συστήματα τα οποία δεν έχουν γραμμή εντολών

2)Explorer interface: Παρέχει γραφικό περιβάλλον για τις ρουτίνες του weka και τα συστατικά του μέρη ,περισσότερο για το exploring of data

3)Experimenter: Επιτρέπει στη δημιουργία πειραμάτων και στατιστικών αναλύσεων των σχημάτων που παρέχονται

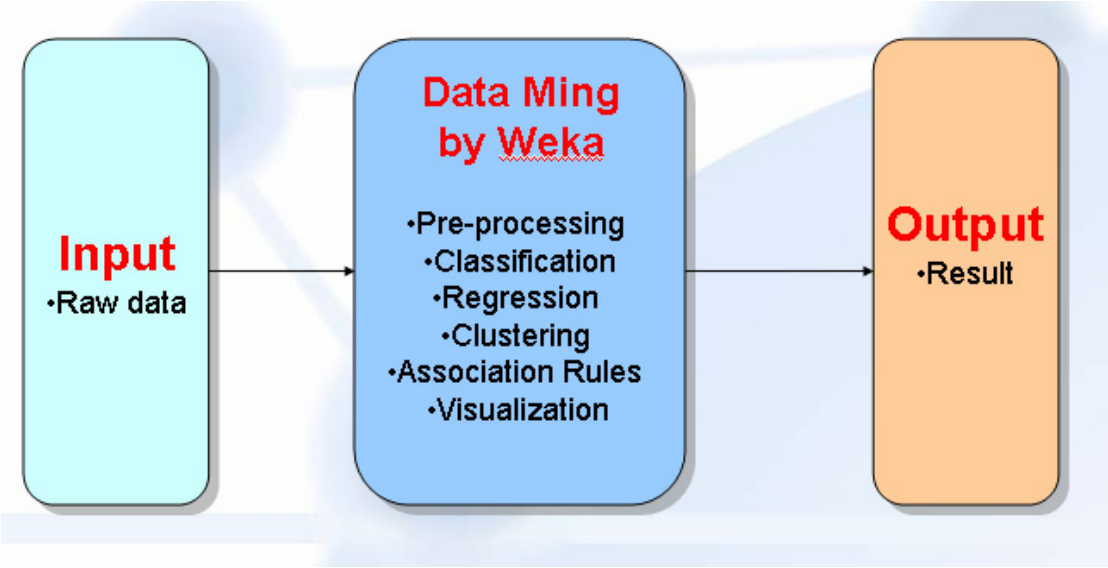
4)Knowledge flow: Προσφέρει την δυνατότητα ανάπτυξης σύνθετων μοντέλων.



Εικόνα 3.2 το περιβάλλον διεπαφής του weak

Το WEKA μας παρέχει εφαρμογές αλγορίθμων εκμάθησης τους οποίους μπορούμε εύκολα να τους εφαρμόσουμε στα δεδομένα μας. Επίσης μας δίνει την δυνατότητα προεπεξεργασίας δεδομένων, προσαρμογή τους σε σχήματα εκμάθησης και ανάλυση του προτεινόμενου ταξινομητή και του αποτελέσματος της ταξινόμησης χωρίς να γράψουμε τον οποιοδήποτε κώδικα προγράμματος. Έτσι μας προσφέρει την λύση πολλών προβλημάτων μηχανικής μάθησης περιέχοντας μεθόδους για:

- Προεπεξεργασία δεδομένων (Pre-proccesing)
- Ταξινόμηση (Classification)
- Παλινδρόμηση (Regression)
- Συσταδιοποίηση (Clustering)
- Εύρεση κανόνων συσχέτισης (Assoscation Rules)
- Εικονικό σχεδιασμό δεδομένων (Visualization)



Εικόνα 3.3 Εφαρμογές του εργαλείου weka

Η χρήση του WEKA ποικίλει ανάλογα με το πρόβλημα το οποίο έχουμε να αντιμετωπίσουμε. Έτσι μπορούμε να εφαρμόσουμε μια μέθοδο σε ένα σύνολο δεδομένων έτσι ώστε να αναλύσουμε την γνώση την οποία και θα αποκτήσουμε στην έξοδο, να χρησιμοποιήσουμε μοντελοποιημένα πρότυπα για να παράγουμε προβλέψεις

ταξινόμησης καινούργιων δεδομένων και τέλος να συγκρίνουμε την απόδοσή τους ώστε να επιλεγεί η βέλτιστη μέθοδος.

Η μέθοδος αποθήκευσης στοιχείων στο πρόγραμμα WEKA γίνεται συνήθως με ένα αρχείο arff το οποίο περιέχει όλα τα δεδομένα τα οποία πρέπει να χρησιμοποιηθούν και γράφονται και διαβάζονται μέσω του προγράμματος notepad. Ένα αρχείο arff αποτελείται από την λίστα των περιπτώσεων (instances) που διαθέτουμε, και οι τιμές των χαρακτηριστικών για κάθε περίπτωση διαχωρίζονται από κόμματα. Επίσης εκτός από την μορφή arff το πρόγραμμα αναγνωρίζει και την μορφή αρχείου csv. Το format του κάθε αρχείου θα είναι ως εξής:

Όνομα του αρχείου δεδομένων

@RELATION numcat2

Ονόματα χαρακτηριστικών και ο τύπος τους

@ATTRIBUTE num NUMERIC
@ATTRIBUTE pond NUMERIC
@ATTRIBUTE metro {Oui}
@ATTRIBUTE Q1 NUMERIC
@ATTRIBUTE Q2 NUMERIC
@ATTRIBUTE Q3 NUMERIC
@ATTRIBUTE Q4 NUMERIC
@ATTRIBUTE class {enough,little,lot,bad}

Δεδομένα

@DATA
1,1815.96,Oui,8,8,7,8,enough
8,5857.57,Oui,5,6,3,2,little
9,1401.72,Oui,6,7,6,5,enough
15,3554.80,Oui,7,8,7,8,lot
22,7776.88,Oui,8,8,8,6,enough

29,717.21,Oui,5,8,6,8,enough

30,1106.86,Oui,6,8,7,6,enough

38,3593.59,Oui,8,5,5,5,enough

44,1447.25,Oui,7,8,8,8,lot

52,4825.49,Oui,8,8,8,5,enough

64,1380.39,Oui,5,10,5,10,enough

67,1538.53,Oui,8,7,8,8,enough

70,1335.41,Oui,8,8,8,8,lot

73,1302.76,Oui,8,8,8,10,lot

..

..

..

Αυτό που θα μας απασχολήσει στην συνέχεια της παρούσας εργασίας θα είναι κυρίως η διαδικασία της ταξινόμησης δηλ. η προσπάθεια πρόβλεψης της κατηγορίας αταξινόμητων δεδομένων μέσω ενός μοντέλου το οποίο δημιουργείται βασισμένο σε κάποιες μεταβλητές και κανόνες πρόβλεψης. Έτσι θα διαπιστώσουμε κατά πόσο τα συγκεκριμένα μοντέλα μπορούν να μας εξασφαλίσουν μια αξιόπιστη ταξινόμηση η οποία θα μας προσφέρει την απαραίτητη γνώση για λήψη αποφάσεων αλλά και πρόβλεψη. Το κριτήριο σε αυτές τις περιπτώσεις είναι το ποσοστό των επιτυχόντων προσπαθειών ταξινόμησης διαφορετικών δεδομένων που ισούται με το κλάσμα των επιτυχόντων προς τον συνολικό αριθμό προσπαθειών ταξινόμησης. Αξίζει να σημειωθεί ότι όπως είναι φυσικό η επιτυχία της ταξινόμησης εξαρτάται σε μεγάλο βαθμό από την ποιότητα αλλά και την ποσότητα των δεδομένων. Έτσι ο όγκος των δεδομένων θα πρέπει να είναι επαρκής αλλά και η πληροφορία σημαντική έτσι ώστε να μπορεί να γίνει ο διαχωρισμός των ιδιοτήτων της κάθε ομάδας και συνεπώς η κατασκευή του μοντέλου το οποίο θα χρησιμοποιήσουμε για την αξιολόγηση της ταξινόμησης.

3.3 Σετ εκπαίδευης και ελέγχου

Όπως είναι ήδη γνωστό η ταξινόμηση(classification) είναι μια διαδικασία Μάθησης υπό Επίβλεψη (Supervised learning). Τα δεδομένα τα οποία θα χρησιμοποιηθούν αναπαρίστανται με πίνακα ταξινομημένων περιπτώσεων. Οι στήλες

του πίνακα δηλώνουν τα χαρακτηριστικά (features) και τον τύπο τους και οι γραμμές τις διαφορετικές περιπτώσεις χαρακτηριστικών, δηλ. τα στιγμιότυπα των δεδομένων. Επίσης η τελευταία στήλη μας δηλώνει την κλάση στην οποία ανήκει κάθε στιγμιότυπο.

Σε τέτοια προβλήματα ταξινόμησης η στρατηγική η οποία ακολουθείται είναι να χωρίσουμε τα δεδομένα μας σε σετ εκπαίδευσης (training set) και σετ ελέγχου (test set). Το σετ εκπαίδευσης περιέχει τα δεδομένα από τα οποία εκπαιδεύεται το μοντέλο μας ώστε να δημιουργήσουμε τα πρότυπα εκμάθησης και το σετ ελέγχου περιέχει δεδομένα τα οποία θα εξεταστούν για να διαπιστώσουμε κατά πόσο είναι δυνατόν να γίνει πρόβλεψη της κλάσης στην οποία ανήκουν τα δεδομένα αυτά και κατά συνέπεια η επιτυχία της ταξινόμησης.

Όπως αναφέραμε και προηγουμένως η αξιοπιστία ενός ταξινομητή αξιολογείται με την ποσοστιαία αναλογία σωστών ταξινομήσεων. Σημαντικό βήμα στην όλη διαδικασία είναι ο σωστός διαχωρισμός των σετ ελέγχου και εκπαίδευσης ο οποίος είναι και υποκειμενικός. Στόχος είναι να έχουμε ικανοποιητικό αλλά και ποιοτικό αριθμό δεδομένων εκπαίδευσης ώστε να δημιουργήσουμε τα κατάλληλα σχήματα και μεγάλο αριθμό δεδομένων ελέγχου ώστε η αξιολόγηση του μοντέλου να είναι όσο το δυνατόν πιο αντικειμενική και αξιόπιστη. Σημαντικό είναι επίσης να αναφερθεί ότι για την αξιολόγηση του μοντέλου δεν μπορούμε να χρησιμοποιήσουμε δεδομένα από το σετ εκπαίδευσης καθώς τα αποτελέσματα θα είναι προσαρμοσμένα στα δεδομένα εκπαίδευσης του μοντέλου και σαν επακόλουθο η αξιολόγησή του δεν θα είναι αξιόπιστη και αντικειμενική. Στην περίπτωση αυτή συναντάμε το φαινόμενο της υπέρ-προσαρμογής (overfitting), οπότε τα δεδομένα ελέγχου δεν θα πρέπει να έχουν παίξει ρόλο κατά την εκπαίδευση και αυτή είναι και η αιτία ύπαρξης των δύο σετ.

3.4 Παρουσίαση αλγορίθμων που επιλέχθηκαν

3.4.1 Μπαιεζιανοί (Bayes) ταξινομητές

Η Μάθηση κατά Μπαιεζ (Bayes) αποτελεί μια άλλη ιδιαίτερα δημοφιλή προσέγγιση για την επαγωγική κατασκευή ταξινομητών, αφενός διότι εκπορεύεται από τον χώρο των Πιθανοτήτων και αφετέρου διότι έχει επιδείξει σημαντικά

αποτελέσματα σε ένα ευρύτατο φάσμα εφαρμογών. Η λειτουργία αυτής της κατηγορίας αλγορίθμων στηρίζεται στην υπόθεση ότι η υπό εκμάθηση έννοια σχετίζεται άμεσα με την κατανομή των πιθανοτήτων που παρουσιάζουν τα στιγμιότυπα του προβλήματος αναφορικά με την κλάση στην οποία ανήκουν. Συνεπώς, υπάρχει μια τελείως διαφορετική αντιμετώπιση του χώρου υποθέσεων. Δεν κατασκευάζεται ένας ταξινομητής ο οποίος βελτιώνεται σύμφωνα με τις επιδόσεις του στο σύνολο υποθέσεων, αλλά αναζητείται η υπόθεση με την μεγαλύτερη πιθανότητα να ταξινομεί σωστά τα στιγμιότυπα του συνόλου εκπαίδευσης. Εν συνεχεία δίνονται κάποιες από τις βασικότερες έννοιες στο χώρο της μηχανικής μάθησης που βασίζονται στην στατιστική. Το σημαντικότερο από αυτά είναι το θεώρημα του Μπαιεζ.

Έστω μια υπόθεση h ενός χώρου υποθέσεων H και D το σύνολο δεδομένων που χρησιμοποιείται στην εκπαίδευση. Η πιθανότητα η υπόθεση h να ταξινομεί σωστά (ή τουλάχιστον με την επιζητούμενη ακρίβεια) τα στιγμιότυπα συμβολίζεται με P(h) ενώ η πιθανότητα η υπόθεση να ταξινομεί σωστά τα στιγμιότυπα του D συμβολίζεται με P(h|D) και καλείται δεσμευμένη πιθανότητα. Ο Bayes διατύπωσε το θεμελιώδες για την στατιστική θεώρημα που είναι γνωστό με το όνομά του και συνοψίζεται στον τύπο:

$$P(h \mid D) = \frac{P(D \mid h)P(h)}{P(D)} \qquad (3.4.1.1)$$

Παρατηρείται ότι η πιθανότητα της υπόθεσης που μελετάται να ταξινομεί σωστά το σύνολο εκπαίδευσης αυξάνει όταν αυξάνεται η πιθανότητα να ταξινομεί σωστά όλα τα πιθανά στιγμιότυπα. Στη συνέχεια θα δοθούν κάποιοι ορισμοί εννοιών που συναντώνται συχνά στις μεθόδους μηχανικής μάθησης που έχουν ως βάση το θεώρημα του Μπέυζ. Μια υπόθεση λέγεται MAP (μέγιστη a posteriori) και συμβολίζεται h_{MAP} αν και μόνο αν

$$h_{MAP} = \arg_{h \in H} \max P(h \mid D) = \arg_{h \in H} \max \frac{P(D \mid h)P(h)}{P(D)} = \arg_{h \in H} \max P(D \mid h)P(h)$$

(3.4.1.2)

Το P(D) παραλήφθηκε καθώς είναι σταθερά ως προς τις υποθέσεις. Επίσης μερικές φορές δεν έχουμε γνώση για τις υποθέσεις h.Τότε μπορούμε να θεωρήσουμε πως και ο όρος P(h) είναι σταθερός για όλες τις υποθέσεις και να τον απαλείψουμε από τον τύπο .Έτσι προκύπτει η μέγιστη πιθανοφάνεια (maximum likelihood)

$$h_{ML} = \arg_{h \in H} \max P(D \mid h) \qquad (3.4.1.3)$$

Προηγουμένως δόθηκε απάντηση στο ερώτημα της πιο πιθανής υπόθεσης. Τώρα θα δοθεί απάντηση και στο ερώτημα της καταλληλότερης ταξινόμησης. Μια υπόθεση πρέπει να καλύπτει το σύνολο των στιγμιοτύπων όμως κατά την αναζήτηση της κατάλληλης υπόθεσης είναι πολύ πιο απλό να κατασκευαστεί μια υπόθεση που να μπορεί να ταξινομεί τα περισσότερα (και όχι όλα) στιγμιότυπα σωστά. Κατά την ταξινόμηση ενός στιγμιότυπου, προτεραιότητα έχει η ορθή ταξινόμηση του συγκεκριμένου στιγμιότυπου, χωρίς να έχει κάποια σημασία η επίδοση του ταξινομητή στο σύνολο των στιγμιοτύπων.

Έστω ένα τυχαίο παράδειγμα όπου *i h* είναι οι συνεπής υποθέσεις με το σύνολο υποθέσεων D. Τότε οι ταξινόμηση ενός τυχαίου στιγμιότυπου που ανήκει στη κλάση *i v* θα είναι

$$P(v_i \mid D) = \sum_{h_1 \in H} P(v_i \mid h_i) P(h_i \mid D) \qquad (3.4.1.4)$$

Η μέγιστη από αυτές τις ταξινομήσεις καλείται βέλτιστη ταξινόμηση Bayes και είναι

$$\arg_{v_i \in C} \sum_{h1 \in H} P(v_i \mid h_i) P(h_i \mid D) \qquad (3.4.1.5)$$

Δεν υπάρχει μέθοδος που να μπορεί να ταξινομήσει στιγμιότυπα με την καλύτερη ακρίβεια (κατά μέσο όρο). Κατά συνέπεια ο ταξινομητής αυτός μπορεί να μας δώσει ένα άνω φράγμα ακρίβειας για όλες τις μεθόδους μηχανικής μάθησης. Επιπλέον, παρά τον συμβολισμό, μπορεί να εφαρμοστεί ακόμα και αν οι συνεπής υποθέσεις δεν περιέχονται στο χώρο υποθέσεων. Δυστυχώς όμως απαιτεί μεγάλο αριθμό πράξεων.

Το μοντέλο του βέλτιστου κατά Μπαιεζ ταξινομητή που παρουσιάστηκε στην προηγούμενη παράγραφο, παρ' ότι είναι σε θέση να μας υποδείξει ένα άνω φράγμα των επιδόσεων ενός συστήματος ταξινόμησης, καθίσταται ιδιαίτερα δύσκολο να χρησιμοποιηθεί στην πράξη, καθώς προϋποθέτει τον υπολογισμό ενός μεγάλου αριθμού πιθανοτήτων. Αυτός ο υπολογισμός τις περισσότερες φορές είναι στην πράξη μη εφικτός, όχι μόνο λόγω του πλήθους των συναρτήσεων, αλλά και εξ αιτίας του μικρού μεγέθους του σώματος εκπαίδευσης που συχνά διαθέτουμε. Μια πιο πρακτική προσέγγιση της συλλογιστικής που αναπτύχθηκε είναι δυνατή μέσω του Αφελούς Μπεϋζιανού Ταξινομητή (Naive Bayes ή NBClassifier).

Αν a1,a2,…..,,an είναι το σύνολο των χαρακτηριστικών των στιγμιοτύπων, σύμφωνα με την προσέγγιση του Μπαιεζ η κλάση ενός τυχαίου στιγμιότυπου είναι

$$v_{MAP} = \arg_{v_i \in V} \max P(v_j \mid a_1, a_2, ..., a_n) \qquad (3.4.1.6)$$

Η οποία μέσω του θεωρήματος του Μπαιεζ μπορεί να εκφραστεί ως

$$v_{MAP} = \arg_{v_i \in V} \max P(a_1, a_2, ..., a_n \mid v_i) P(v_j) \qquad (3.4.1.7)$$

Αν τώρα υποτεθεί ότι οι τιμές των χαρακτηριστικών είναι ανεξάρτητες μεταξύ τους, τότε η

$$\arg_{v_i \in V} \max P(v_j) \prod_i P(a_i \mid v_j) \qquad (3.4.1.8)$$

είναι μια απλοποιημένη μορφή με λιγότερο κόστος για τον υπολογισμό της πιο πιθανής ταξινόμησης και καλείται αφελής ταξινομητής Μπέυζ. Το όνομα του ο τύπος το οφείλει στο γεγονός ότι η τιμή του ενός χαρακτηριστικού δεν επηρεάζει την τιμή που θα πάρει το άλλο κάτι που συχνά δεν ισχύει στην πραγματικότητα. Παρόλα αυτά οι μέθοδοι που χρησιμοποιούν τον αφελή ταξινομητή Μπέυζ έχουν απόδοση συγκρίσιμη με αυτή άλλων αλγορίθμων.

3.4.2 Αλγόριθμος k κοντινότερων Γειτόνων (k-Nearest Neighboor)

Η μέθοδος των κοντινότερων γειτόνων είναι μια από τις πιο δημοφιλείς, παλαιότερες και αξιόπιστες μεθόδους ταξινόμησης. Βασίζεται στα στιγμιότυπα του συνόλου των δεδομένων και προσεγγίζουν τη λύση σε ένα καινούργιο στιγμιότυπο. Έχει εφαρμοστεί στο πεδίο της στατιστικής, της μοντελοποίησης και της αναγνώρισης προτύπων και τελευταία χρησιμοποιείται και στον τομέα της μηχανικής μάθησης.

Ο αλγόριθμος αυτός σε αντίθεση με πολλούς άλλους, δεν κατασκευάζει ένα μοντέλο από τα δεδομένα εκπαίδευσης για να γίνει ο έλεγχος από τα δεδομένα ελέγχου για την σωστή πρόβλεψη των κλάσεων. Ο τρόπος λειτουργίας του είναι να προβλέπει την κλάση ενός νέου στιγμιοτύπου βασισμένος μόνο και μόνο από τις αντίστοιχες τιμές των k πιο κοντινών στιγμιότυπων γνωστής κλάσης τα οποία και θα είναι οι γείτονές του. Έτσι ανάλογα με την πλειοψηφία των κλάσεων των k κοντινότερων γειτόνων του προβλέπεται και η κλάση στο νέο στιγμιότυπο. Τα σημαντικά στοιχεία τα οποία καθορίζουν και την λειτουργία του αλγόριθμου είναι τα παρακάτω:

- Ο ορισμός της απόστασης ανάμεσα στα στιγμιότυπα, δηλαδή το μέτρο το οποίο θα μας εκφράσει την εγγύτητα ή αλλιώς την ομοιότητα ανάμεσά τους έτσι ώστε να μπορούμε να τα συγκρίνουμε μεταξύ τους

- Ο τρόπος συνδυασμού μεταξύ των γειτόνων

- Η τιμή του k, που μας δηλώνει πόσους γείτονες θα χρησιμοποιήσουμε για να επιλέξουμε την κλάση στην οποία ανήκει το νέο στιγμιότυπο.

Η έννοια της εγγύτητας εξαρτάται από τον τύπο των χαρακτηριστικών και συνήθως εκφράζεται από την ευκλείδεια απόσταση ανάμεσα στα χαρακτηριστικά. Ποιο συγκεκριμένα εάν τα στιγμιότυπα x αποτελούνται από διανύσματα που περιέχουν πραγματικούς αριθμούς με την μορφή

$$\langle a_1(x), a_2(x), ...a_n(x) \rangle \qquad (3.4.2.1)$$

Όπου το ak(x) είναι η τιμή του κ-οστού χαρακτηριστικού τότε η απόσταση μεταξύ δύο χαρακτηριστικών Xi , Xj ορίζεται ως

$$d(x_{i,}x_j) \equiv \sqrt{\sum_{r=1}^{n}(a_r(x_i) - a_r(x_j))^2} \qquad (3.4.2.2)$$

Στην περίπτωση όπου τα χαρακτηριστικά δεν είναι αριθμητικά αλλά ονομαστικά(nominal) τότε η ευκλείδεια απόσταση δεν θα έχει κανένα νόημα αφού δεν μπορούν να γίνουν πράξεις ανάμεσα σε συμβολικές ποσότητες. Τότε αυτό το οποίο χρησιμοποιείται συνήθως είναι το μέτρο επικάλυψης η αλλιώς απόσταση Hamming ή απόσταση Manhattan και ορίζεται ως εξής:

$$d(x_i, x_j) \equiv \sum_{r=1}^{n}\delta(a_r(x_i), a_r(x_j)) \qquad (3.4.2.3)$$

Αυτό το μέτρο μας δίνει ουσιαστικά τον αριθμό των χαρακτηριστικών στα οποία διαφέρουν τα στιγμιότυπα.

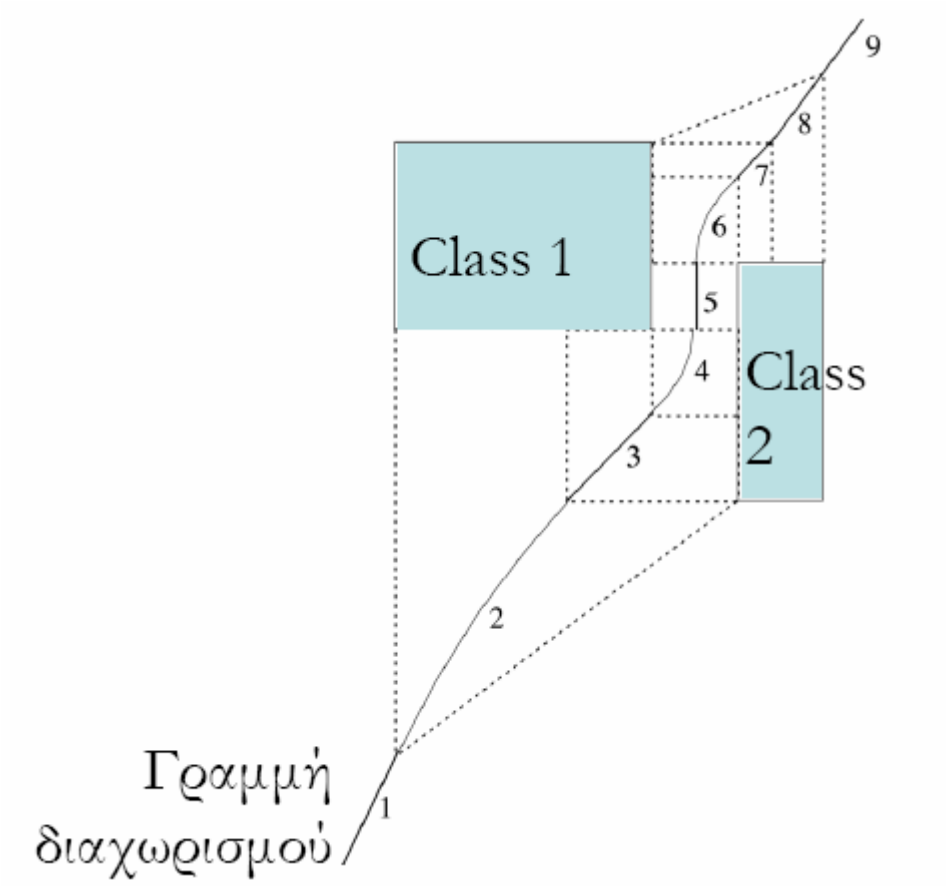
3.4.2.1 Ο αλγόριθμος IBK

Ο IBK είναι από τους ποιο σημαντικούς αλγόριθμους της κατηγορίας αυτής. Κρατά μια πλήρη μνήμη των στιγμιοτύπων κατάρτισης και ταξινομεί τις νέες περιπτώσεις χρησιμοποιώντας τις ποιο παρόμοιες περιπτώσεις κατάρτισης. Η νέα περίπτωση ταξινομείται μετά από την εύρεση της περίπτωσης με την μεγαλύτερη ομοιότητα όπου της προσαρτάται η αντίστοιχη κλάση. Το μειονέκτημα του αλγόριθμου αυτού είναι ότι διατηρεί μια μεγάλη μνήμη για την αποθήκευση των στιγμιοτύπων κατάρτισης. Για να προβλέψει την κλάση του καινούργιου στιγμιότυπου ο IBK χρησιμοποιεί την συνάρτηση ομοιότητας που ορίζεται ως:

$$Similarity(x, y) = -\sqrt{f\left(x_{i,}^{n} y_i\right)} \qquad (3.4.2.1.1)$$

3.4.2.2 Ο αλγόριθμος KSTAR

Ο αλγόριθμος KSTAR είναι άλλη μία μέθοδος κοντινότερων γειτόνων, η οποία χρησιμοποιεί γενικευμένη συνάρτηση απόστασης που βασίζεται σε μετασχηματισμό. Σύμφωνα με αυτή τη μέθοδο λαμβάνουμε υπόψιν ότι ένα στιγμιότυπο μετασχηματίζεται σε ένα άλλο μέσω μιας σειράς προκαθορισμένων διεργασιών και στην συνέχεια υπολογίζουμε την πιθανότητα της σειράς αυτής να συμβαίνει εάν οι διεργασίες επιλέγονται τυχαία. Έτσι έχουμε βελτίωση αν όλοι οι μετασχηματισμοί ληφθούν υπόψιν, αποκτήσουν την πιθανότητά τους και το σχήμα γενικεύεται σε πρόβλημα υπολογισμού της απόστασης ανάμεσα σε ένα στιγμιότυπο και ένα σύνολο άλλων στιγμιοτύπων με την εξέταση των μετασχηματισμών σε όλες τις περιπτώσεις στο σύνολο. Έτσι με αυτό το μέτρο, λαμβάνοντας υπόψη ένα στιγμιότυπο ελέγχου όπου η κλάση του είναι άγνωστη, η απόστασή της στο σύνολο των δεδομένων εκπαίδευσης υπολογίζεται σε κάθε κλάση ξεχωριστά και η ποιο κοντινή κλάση επιλέγεται. Αξίζει να σημειωθεί ότι και οι ονομαστικές και οι αριθμητικές τιμές μπορούν να αντιμετωπιστούν ομοιόμορφα με αυτήν την μέθοδο .



Εικόνα 3.4

3.4.3 Αλγόριθμοι Meta(Bagging)

Οι «μετα» μαθησιακοί αλγόριθμοι μπορούν να συνδυάσουν άλλους ταξινομητές και να βελτιώσουν την απόδοση. Η κύρια ιδέα είναι η κατασκευή πολλών «εμπειρογνωμόνων», που είναι μοντέλα που δημιουργούνται με τεχνικές μηχανικής μάθησης, για την ανάδειξη της πλειοψηφούσας γνώμης. Το πλεονέκτημά τους είναι ότι συχνά βελτιώνουν την προβλεπτική ικανότητα, ενώ το μειονέκτημα είναι ότι τα εξαγόμενα πολλές φορές είναι δύσκολο να αναλυθούν.

Bagging

Ο αλγόριθμος bagging(εμφωλίαση) συνδυάζει προβλέψεις μέσω καταμέτρησης ψήφων / εύρεσης μέσου όρου και κάθε μοντέλο λαμβάνει ισοδύναμη βαρύτητα. Αρχικά γίνεται δειγματοληψία αρκετών συνόλων εκπαίδευσης μεγέθους n, κατασκευάζεται

ένας ταξινομητής για κάθε σύνολο εκπαίδευσης και στο τέλος συνδυάζονται οι προβλέψεις των ταξινομητών. Η λειτουργία της εμφωλίασης έγκειται κυρίως στην μείωση της διακύμανσης με της καταμέτρησης ψήφων /εύρεσης μέσου όρου. Συνήθως η βελτίωση είναι ανάλογη του αριθμού των ταξινομητών. Τέλος για κάθε μαθησιακό σχήμα θα πρέπει να υπολογίζονται οι εξής παράμετροι:

- Bias(προκατάληψη):αναμενόμενο σφάλμα του μεταταξινομητή σε νέα δεδομένα
- Variance(διακύμανση):αναμενόμενο σφάλμα λόγω του συγκεκριμένου συνόλου εκπαίδευσης που χρησιμοποιήθηκε
- Συνολικά αναμενόμενο σφάλμα = bias + variance

3.4.4 Δέντρα Απόφασης

Τα δέντρα απόφασης αποτελούν μια από τις ποιο βασικές μεθόδους πρόβλεψης και ταξινόμησης και αποτελούν μια επαγωγική διαδικασία. Αναπαριστούν κανόνες και είναι αρκετά διαδεδομένα καθώς ο μηχανισμός της διαδικασίας απόφασης είναι αρκετά εμφανής και επιτρέπει αρκετά καλή ανάλυση της γνώσης την οποία λαμβάνουμε. Τα κύρια χαρακτηριστικά του είναι οι κόμβοι, τα κλαδιά και τα φύλλα. Σύμφωνα με έναν ορισμό, ένα δέντρο απόφασης είναι ένα δέντρο με τις ακόλουθες ιδιότητες:

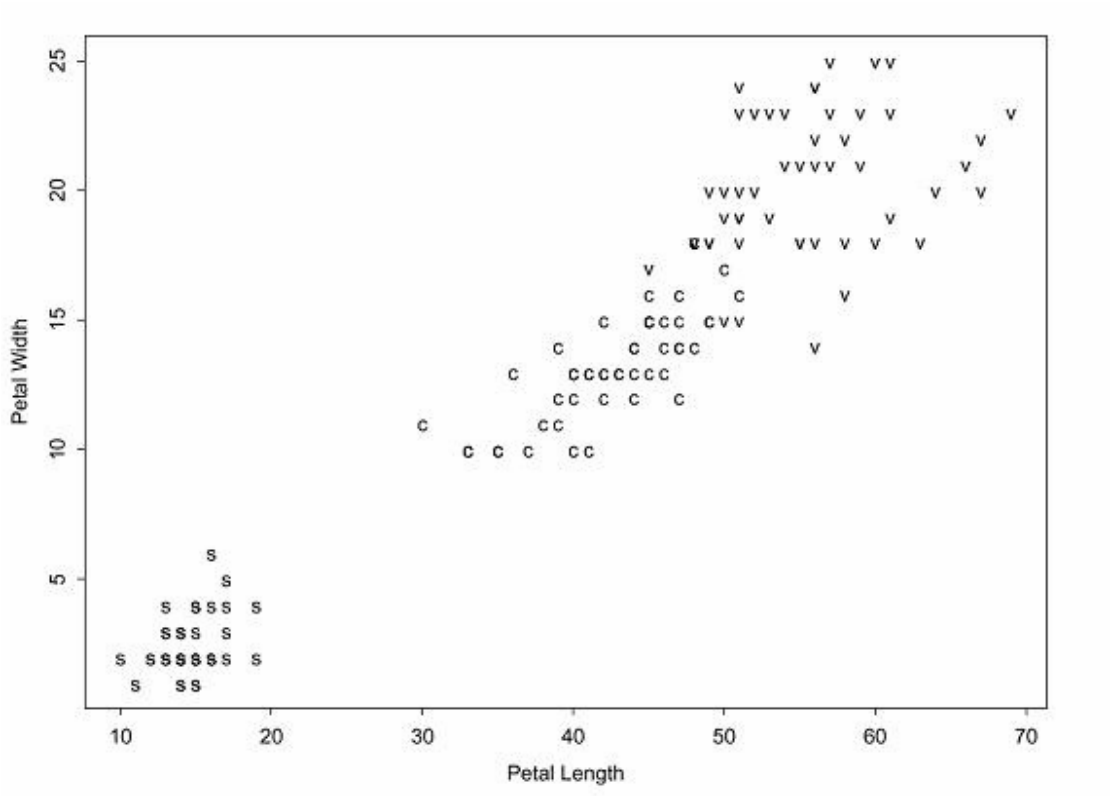
- Κάθε εσωτερικός κόμβος ονοματίζεται με το όνομα ενός χαρακτηριστικού x
- Κάθε κλαδί / σύνδεση ονοματίζεται με ένα κατηγόρημα που μπορεί να εφαρμοστεί στο χαρακτηριστικό που αποτελεί το όνομα του κόμβου
- Κάθε φύλλο ονοματίζεται με το όνομα μιας κλάσης

Οι κόμβοι του δέντρου αφορούν τον έλεγχο ενός συγκεκριμένου χαρακτηριστικού. Τα κλαδιά περιέχουν περιέχουν συνθήκες σύγκρισης (κυρίως ανισότητες) της τιμής που λαμβάνει το συγκεκριμένο χαρακτηριστικό με μια άλλη τιμή η οποία και θα προσδιορίσει σε ποιο κόμβο ¨ παιδί¨ θα συνεχίσουμε την αναζήτηση ώστε να φτάσουμε στην κλάση την οποία και θέλουμε να προβλέψουμε και αναπαρίσταται στα φύλλα του δέντρου.

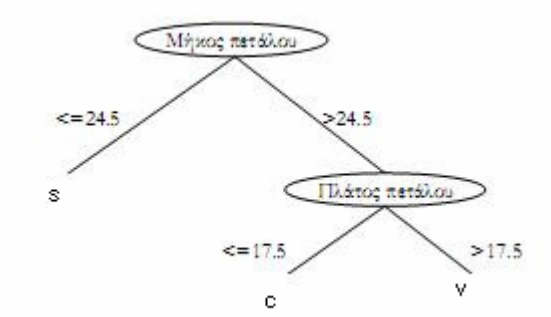
Έτσι λοιπόν για να ταξινομήσουμε μια νέα περίπτωση, ακολουθούμε μια πορεία από την κορυφή του δέντρου και καταλήγουμε στα φύλλα όπου μας καθορίζουν την κλάση στην οποία θα ανήκουν τα χαρακτηριστικά μας. Κατά την διαδρομή οι κόμβοι ελέγχουν τις τιμές που λαμβάνουν τα χαρακτηριστικά ώστε να υποδείξουν την συνέχεια της διαδρομής μέχρι τα φύλλα του δέντρου κάτι το οποίο θα σημαίνει και το τέλος της ταξινόμησης. Αν το χαρακτηριστικό που ελέγχεται σε έναν κόμβο είναι ονομαστική τιμή ,τότε ο αριθμός των κλαδιών τα οποία θα προκύψουν από αυτόν είναι συνήθως όσες και οι πιθανές απαντήσεις του ελεγχόμενου χαρακτηριστικού.

Επιθυμητό είναι να δημιουργούνται δέντρα που είναι ισορροπημένα και με τα λιγότερα επίπεδα. Αυτό όμως δεν είναι πάντα εφικτό αλλά ούτε και υπολογιστικά καλυτέραο. Η δημιουργία ενός δέντρου σταματά οπωσδήποτε όταν όλα τα δεδομένα του συνόλου εκπαίδευσης κατηγοριοποιούνται πλήρως. Μπορεί όμως να είναι απαραίτητο να σταματήσει νωρίτερα για να αποφευχθούν π.χ. μεγάλα δέντρα. Το πότε ή που θα σταματήσει είναι θέμα συναλλαγής μεταξύ ακρίβειας (accuracy) και απόδοσης (performance) του αλγορίθμου. Επίσης πρώιμος τερματισμός μπορεί να γίνει για την αποφυγή του φαινόμενου της προσαρμογής (overfitting). Τέλος μπορεί να προχωρήσει σε μεγαλύτερα δέντρα αν είναι γνωστό ότι υπάρχουν κατηγορίες δεδομένων που δεν αντιπροσωπεύονται στο σύνολο της εκπαίδευσης.

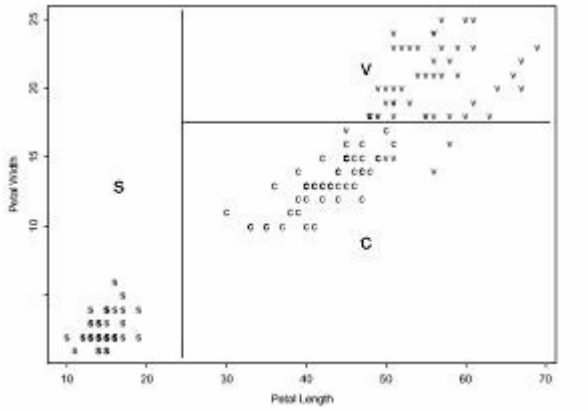
Ένα κλασσικό παράδειγμα εφαρμογής αλγορίθμου δέντρου απόφασης είναι το πρόβλημα της ταξινόμησης του άνθους του λουλουδιού της ίριδας σε ένα από τα 3 γνωστά είδη:iris setosa, iris versicolor ,iris virginica.Τα 4 χαρακτηριστικά τα οποία και προσδιορίζουν την κλάση του άνθους είναι το μήκος και το πλάτος των πετάλων και των σέπαλων του άνθους. Η απεικόνιση των κλάσεων των στιγμιοτύπων, το δέντρο απόφασης και ο διαχωρισμός τους από το δέντρο φαίνεται στις παρακάτω εικόνες:



Εικόνα 3.5 απεικόνιση των κλάσεων των στιγμιοτύπων



Εικόνα 3.6 Το δέντρο απόφασης της ίριδας



Εικόνα 3.7 διαχωρισμός των κλάσεων από το δέντρο απόφασης

3.4.4.1 Αλγόριθμος C4.5(j48)

Μια από τις ποιο γνωστές μεθόδους μηχανικής μάθησης για σχηματισμό κανόνων απόφασης μέσω δέντρου είναι ο αλγόριθμος C4.5 ο οποίος είναι μια εξέλιξη του αλγορίθμου ID3. Ο αλγόριθμος j48 είναι η έκδοση του C4.5 για την πλατφόρμα του WEKA.

Ο ID3 υπήρξε ο κυριότερος εκπρόσωπος των TDIDT δέντρων μέχρι την έλευση του C4.5. Ήταν ο πρώτος αλγόριθμος που χρησιμοποίησε για το κριτήριο καταλληλότητας τεμαχισμού το κέρδος Gain από τη θεωρία πληροφορίας.

Το αποτέλεσμα είναι μια δενδροειδής δομή που με γραφικό τρόπο αναπαριστά τις συσχετίσεις στα δεδομένα εκπαίδευσης ή διαφορετικά, περιγράφει τα δεδομένα. Αρχικά, μια από τις παραμέτρους του συνόλου εκπαίδευσης ορίζεται ως παράμετρος στόχος. Οι υπόλοιπες παράμετροι θεωρούνται παράμετροι εισόδου. Τα βήματα τα οποία ακολουθεί ο αλγόριθμος είναι τα παρακάτω:

1)Βρίσκει την ανεξάρτητη μεταβλητή η οποία αν χρησιμοποιηθεί ως κριτήριο διαχωρισμού των δεδομένων εκπαίδευσης θα οδηγήσει σε κόμβους κατά το δυνατό διαφορετικούς σε σχέση με την εξαρτημένη μεταβλητή

2)Κάνει το διαχωρισμό

3)Επαναλαμβάνει τη διαδικασία για κάθε έναν από τους κόμβους που προέκυψαν μέχρι να μην είναι δυνατός περαιτέρω διαχωρισμός.

Ένας από τους πιο διαδεδομένους μηχανισμούς διαχωρισμού είναι αυτός της εντροπίας της πληροφορίας (information entropy) ο οποίος επιλέγει εκείνη την ανεξάρτητη μεταβλητή που οδηγεί σε περισσότερο συμπαγές δένδρο. Η τιμή της εντροπίας της πληροφορίας δίνεται από τη σχέση:

$$E(S) = -p_{+}.\log_2(p_{+}) - p_{-}.\log_2(p\_) \qquad (3.4.4.1.1)$$

όπου *S* είναι το σύνολο των δεδομένων εκπαίδευσης στο στάδιο (κόμβο) του διαχωρισμού, *p*+ είναι το κλάσμα των θετικών παραδειγμάτων του *S* και *p*- είναι το κλάσμα των αρνητικών παραδειγμάτων του *S*.

Γενικότερα, για *c* διαφορετικές κατηγορίες, η εντροπία ορίζεται από τη σχέση:

$$E(S) = -\sum_{i=1}^{c} p_i . \log_2 p(i) \qquad (3.4.4.1.2)$$

όπου pi το ποσοστό των παραδειγμάτων του *S* που ανήκουν στην κατηγορία *i*.

Η εντροπία της πληροφορίας μετρά ουσιαστικά την ανομοιογένεια που υπάρχει στο *S* αναφορικά με την υπό εξέταση εξαρτημένη μεταβλητή και έχει τις ρίζες της θεωρίας των πληροφοριών (information theory). Στην περίπτωση που έχουμε δυο κατηγορίες, η τιμή της είναι 0 αν όλα τα μέλη του *S* ανήκουν στην ίδια κατηγορία και 1 αν τα μισά μέλη ανήκουν στην μια και τα άλλα μισά στην άλλη κατηγορία. Σε όλους δε τους υπολογισμούς, θεωρούμε την ποσότητα 0-log2(0) ίση με μηδέν.

Στην πράξη, χρησιμοποιείται το κέρδος πληροφορίας (information gain), Gain(S, A) ή G(S,A) που αναπαριστά τη μείωση της εντροπίας του συνόλου εκπαίδευσης *S* αν επιλεγεί ως παράμετρος διαχωρισμού η μεταβλητή *A*. Όταν μειώνεται η πληροφοριακή εντροπία, αυξάνεται η πυκνότητα πληροφορίας και άρα η περιγραφή γίνεται περισσότερο συμπαγής. Το κέρδος πληροφορίας δίνεται από τη σχέση:

$$G(S, A) = E(S) - \sum_{u \in Values(A)} \frac{|Su|}{S} . E(Su) \qquad (3.4.4.1.3)$$

Όπου E(S) είναι η εντροπία πληροφορίας του υπό εξέταση κόμβου, *A* είναι η ανεξάρτητη μεταβλητή, με τιμές Values(A), βάσει της οποίας επιχειρείται ο επόμενος διαχωρισμός, u είναι μία από τις δυνατές τιμές του A, Su είναι το πλήθος των εγγραφών με A=u και E(Su) η εντροπία πληροφορίας του υπό εξέταση κόμβου ως

προς την τιμή A=u. Ουσιαστικά, ο δεύτερος όρος είναι η εντροπία των παραδειγμάτων μετά το διαχωρισμό τους σύμφωνα με την τιμή του χαρακτηριστικού *A* και αποτελείται από το άθροισμα της εντροπίας για το κάθε σύνολο που προκύπτει μετά το διαχωρισμό.

Όπως αναφέραμε και προηγουμένως ο C4.5 είναι μια εκλέπτυνση του ID3. Η μέθοδος του C4.5 είναι βασισμένη στην διαδοχή Bernouli και έχει να κάνει με την εύρεση διαστήματος εμπιστοσύνης από τα δεδομένα εκπαίδευσης και ευρετική επιλογή ορίου για κλάδεμα. Η εκτίμηση σφάλματος του υποδέντρου αποτελεί σταθμισμένο άθροισμα των εκτιμήσεων σφάλματος όλων των φύλλων του. Η εκτίμηση σφάλματος κόμβου υπολογίζεται από την συνάρτηση:

$$e = \left(f + \frac{z^2}{2N} + z\sqrt{\frac{f}{N} - \frac{f^2}{N} + \frac{z^2}{4N}} \right) / \left(1 + \frac{z^2}{N} \right) \qquad (3.4.4.1.4)$$

όπου f είναι το σφάλμα στα δεδομένα εκπαίδευσης ,και N ο αριθμός των υποδειγμάτων που καλύπτονται από το φύλλο.

Τα βασικά πλεονεκτήματα του αλγορίθμου C4.5 έναντι του προκατόχου του ID3 είναι:

- Δυνατότητα καλύτερης διαχείρισης ελλιπών και συνεχών δεδομένων
- Μπορεί να γίνει κλάδεμα με δύο τεχνικές: αντικατάσταση υποδέντρου και ανύψωση υποδέντρου
- Βελτιωμένη συνάρτηση καταλληλότητας για αποφυγή της μεγάλης προσαρμογής στα δεδομένα του δείγματος εκμάθησης(overfitting)

3.5 Αξιολόγηση αλγορίθμων και εξαγωγή αποτελεσμάτων

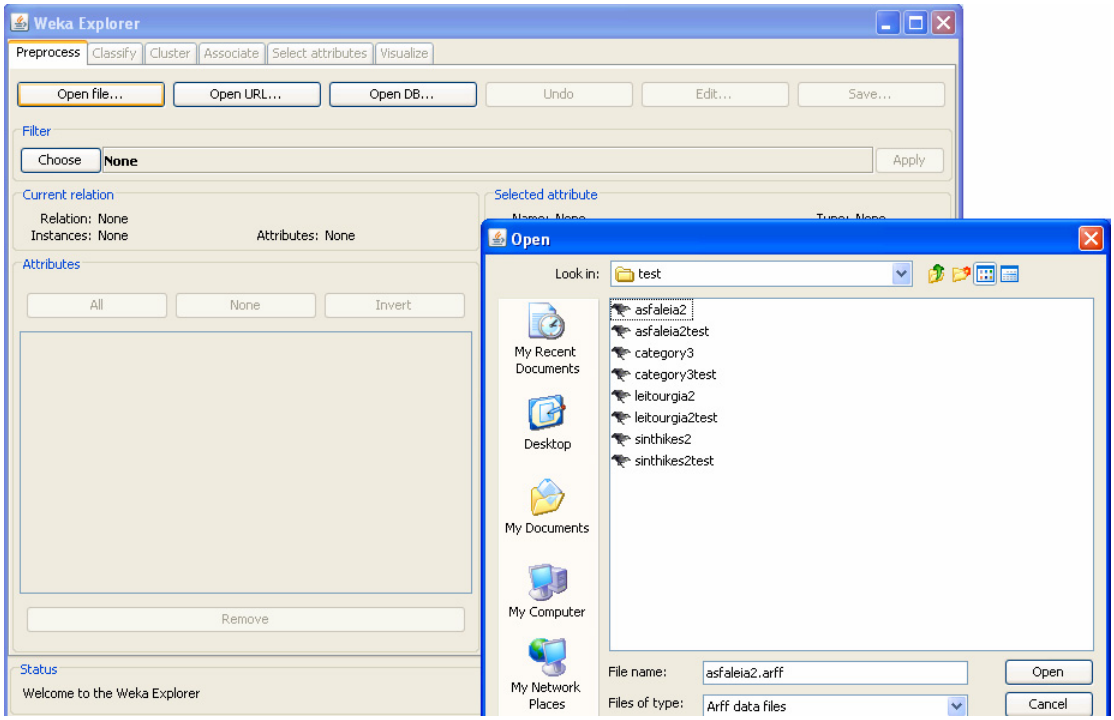
Στο σημείο αυτό θα εξετάσουμε διάφορους αλγόριθμους που διαθέτει το WEKA στην βιβλιοθήκη του, έτσι ώστε να επιλέξουμε αυτούς που θα μας δώσουν καλύτερα ποσοστά αναγνώρισης κλάσεων. Πολλοί από αυτούς απορρίπτονται εξ

αρχής καθώς αναγνωρίζουν συγκεκριμένες τιμές χαρακτηριστικών είτε αυτές είναι αριθμητικές(numeric) είτε ονομαστικές(nominal). Επίσης σημαντικό ρόλο παίζει και ο τύπος του στοιχείου που θέλουμε να προβλέψουμε, που στην περίπτωσή μας είναι nominal, άρα όλοι οι αλγόριθμοι που δέχονται μόνο αριθμητικές τιμές κλάσεων απορρίπτονται.

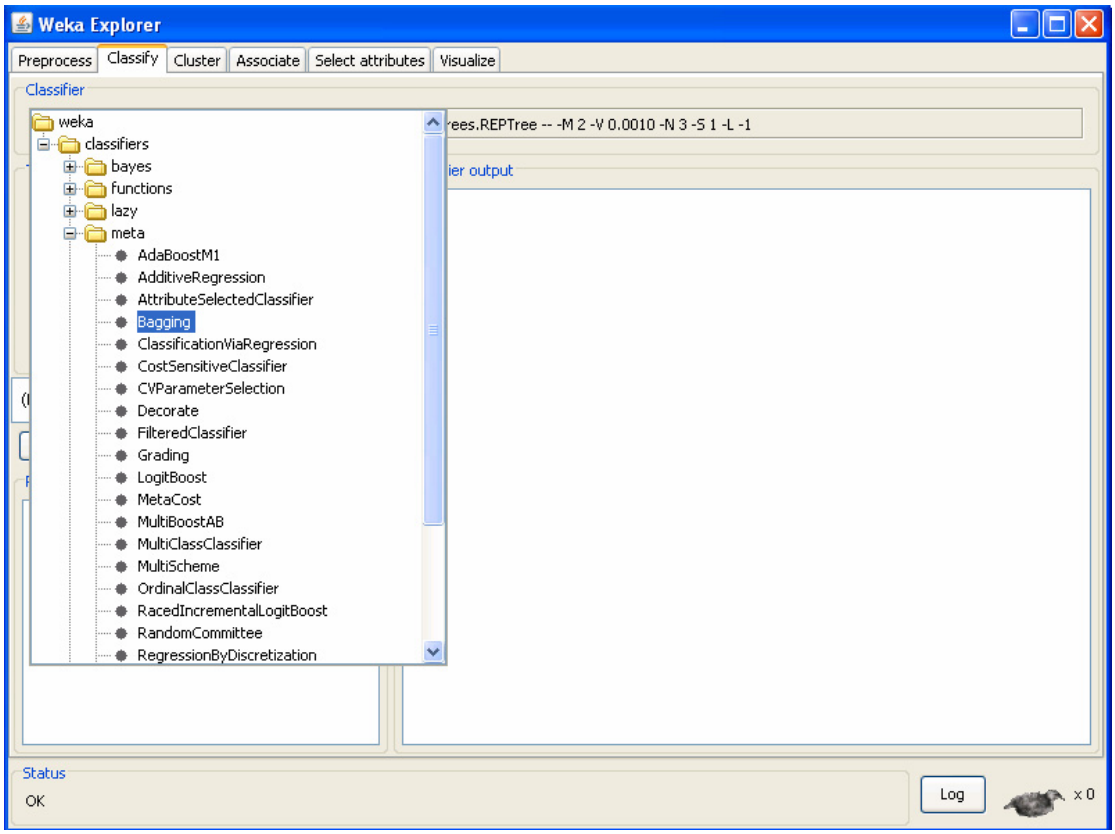
Όπως αναφέρθηκε και προηγουμένως τα δεδομένα μας τα χωρίσαμε σε training test και test set. Το training set καταλαμβάνει περίπου το 1/3 των δεδομένων και το test set το υπόλοιπο 2/3 έτσι ώστε να αποφύγουμε και το φαινόμενο overfitting. Ο αριθμός των στιγμιοτύπων εκπαίδευσης είναι αρκετά ικανοποιητικός ώστε να δημιουργηθούν οι κατάλληλοι κανόνες και σχήματα όπως και ο αριθμός των στιγμιοτύπων ελέγχου που θα μας δώσει ένα αντιπροσωπευτικό και αξιόπιστο ποσοστό αναγνώρισης κλάσεων.

Έχουν αναπτυχτεί πολλές και διάφορες προσεγγίσεις με διαφορετική συμπεριφορά και αποτελέσματα ανάλογα με το είδος των στοιχείων που θα εκπαιδευτούν και την συνάφεια των χαρακτηριστικών με την προς μοντελοποίηση κλάση. Έτσι θα χρησιμοποιήσουμε μεθόδους διαφορετικών οικογενειών όσων αφορά τον τρόπο εξαγωγής σχημάτων και κανόνων οι οποίοι και θα μας οδηγήσουν στην ταξινόμηση.

Για να αρχίσει η διαδικασία χρήσης του WEKA επιλέγουμε από το περιβάλλον διεπαφής τον explorer. Στο σημείο αυτό πηγαίνουμε στην επιλογή Preprocess όπου φορτώνουμε τα δεδομένα προς εκπαίδευση.



στην συνέχεια πηγαίνουμε στην επιλογή classify->choose επιλέγοντας κάθε φορά τον αλγόριθμο τον οποίο θα χρησιμοποιήσουμε



Στο σημείο αυτό έχουμε 4 επιλογές για το πώς θα αξιολογηθεί η ταξινόμηση:

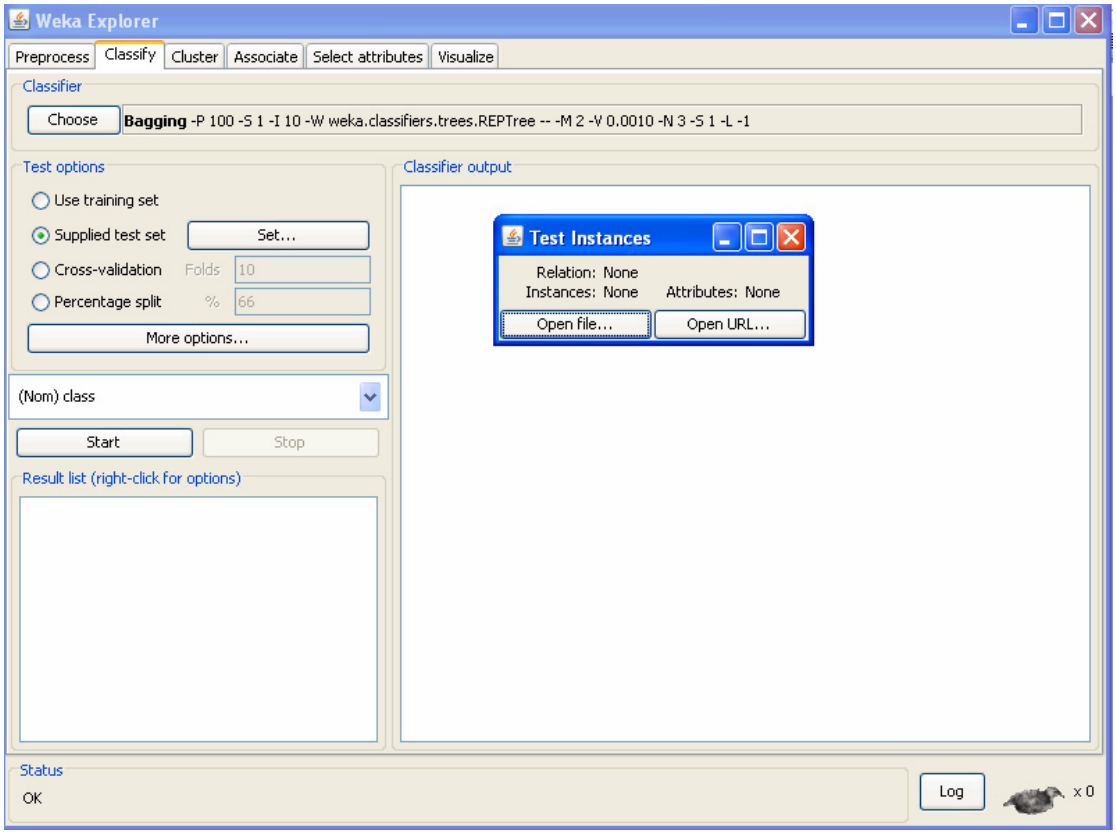
1)Use training set: Ο ταξινομητής αποτιμάται στο πόσο καλά μπορεί να προβλέψει την κλάση των στιγμιοτύπων που εκπαιδεύτηκε

2)Supplied test set: Ο ταξινομητής αποτιμάτε στο πόσο καλά προβλέπει την κλάση από το σετ των στιγμιοτύπων που φορτώθηκαν στο αρχείο

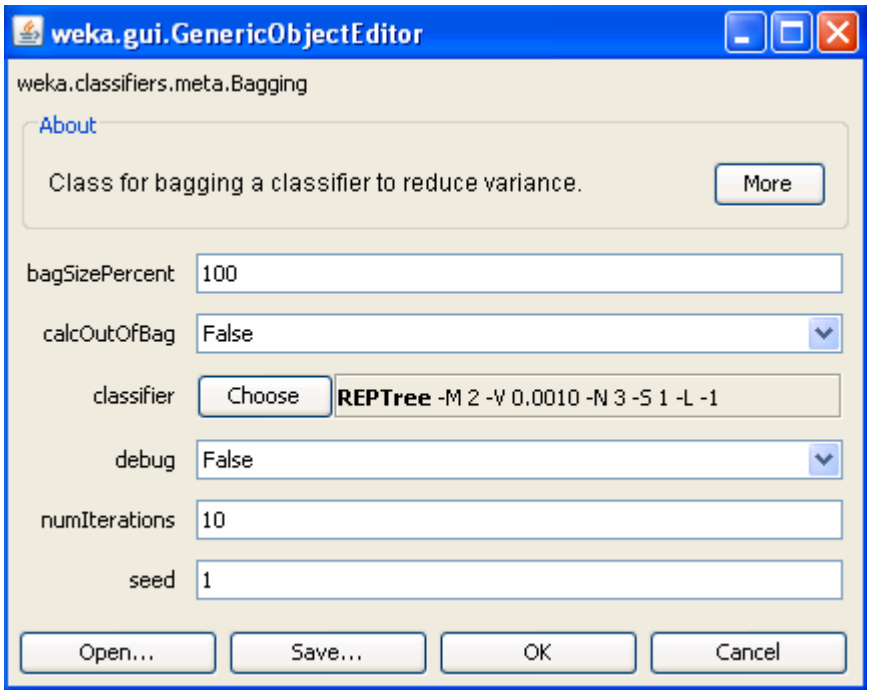
3)Cross-validation:Ο ταξινομητής αποτιμάται από Cross-validation, χρησιμοποιώντας τον αριθμό των folds που εισάγονται στο ανάλογο πεδίο

4)Percentage split: Ο ταξινομητής αποτιμάται στο πόσο καλά προβλέπει ένα certain percentage των δεδομένων που προσφέρονται για testing.Τα δεδομένα αυτά εξαρτώνται από την τιμή που εισάγεται στο πεδίο

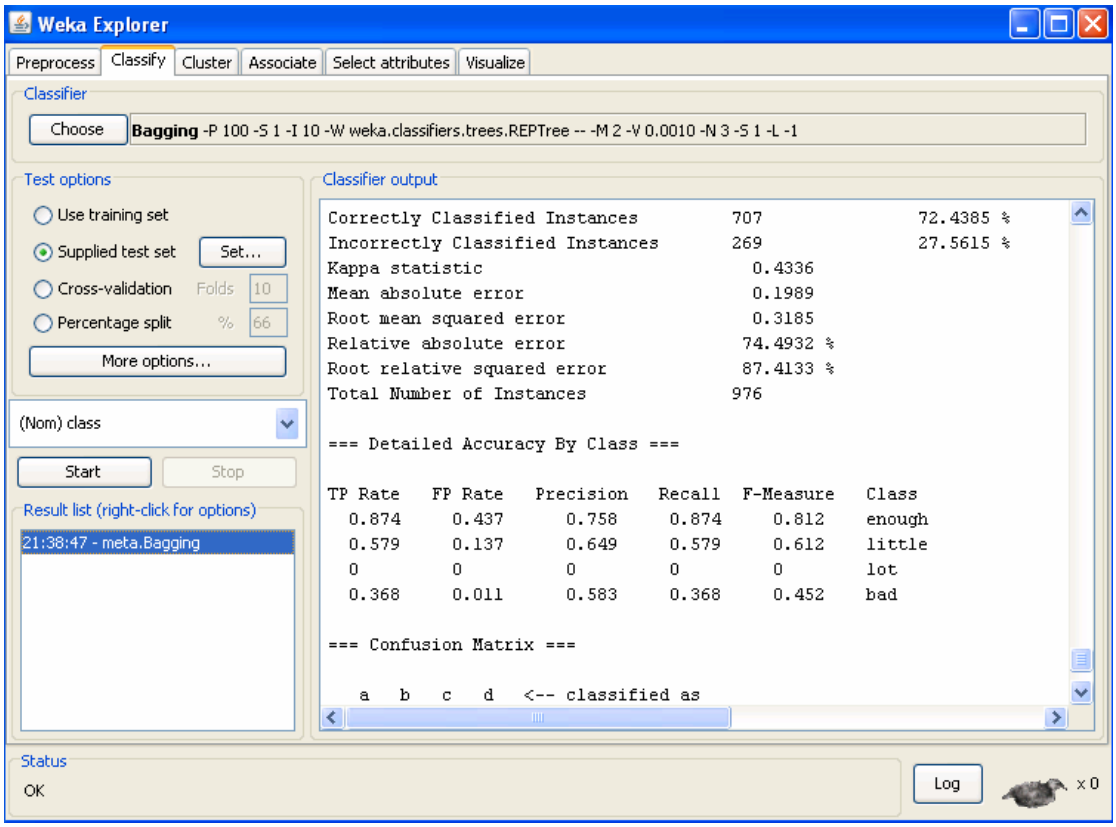
και επιλέγουμε την επιλογή Supplied test set από το test options για να φορτώσουμε και τα δεδομένα εκπαίδευσης.



και κάνοντας διπλό κλικ στην επιλεγμένη μέθοδο μας εμφανίζονται οι παράμετροι



πατώντας start ξεκινάει η διαδικασία του αλγορίθμου και στην συνέχεια εμφανίζονται τα αποτελέσματα.



Τα αποτελέσματα περιλαμβάνουν τις εξής πληροφορίες

1)Run information: Πληροφορίες σχετικά με τις επιλογές του learning scheme, relation name, instances, attributes, και το test mode που σχετίζονται με τη διαδικασία

2)Classifier model: Μια textual αναπαράσταση του classification μοντέλου που δημιουργήθηκε σε όλα τα training data

3)Τα αποτελέσματα

4)Summary: Λίστα στατιστικών για το πώς έγινε η πρόβλεψη του true class των instances κάτω από το επιλεγμένο test mode

5)Detailed Accuracy By Class: Μια πιο λεπτομερής αναφορά ανά κλάση για την ακρίβεια πρόβλεψης του ταξινομητή

6)Confusion Matrix: Δείχνει πόσα instances αντιστοιχίζονται σε κάθε κλάση. Τα elements δείχνουν τον αριθμό των set examples των οποίων η ακριβής κλάση είναι η γραμμή και των οποίων η προβλεπόμενη κλάση είναι η στήλη

Έπειτα από πειραματικούς υπολογισμούς οι αλγόριθμοι οι οποίοι μας έδωσαν καλύτερα ποσοστά σωστής ταξινόμησης είναι οι παρακάτω:

- Naive Bayes
- Ibk
- Kstar
- Bagging
- J48(C4.5)

Τα ποσοστά που μας έδωσαν οι συγκεκριμένοι αλγόριθμοι για τις 4 κατηγορίες δεδομένων είναι τα εξής:

ταξινόμηση για τα δεδομένα που αφορούν την λειτουργία στο μετρό

Algorithm	Correct classified inctances
Naive Bayes	72.8863 %
Ibk	72.6919 %
Kstar	74.5102 %
Bagging	74.2468 %
J48	72.0117 %

Παρατηρούμε ότι ο Kstar μας δίνει το καλύτερο αποτέλεσμα με ποσοστό 74.5102% ενώ ο J48 μας δίνει 72.0117.

ταξινόμηση για τα δεδομένα που αφορούν τις συνθήκες στο μετρό

Algorithm	Correct classified inctances
Naive Bayes	73.2818 %
Ibk	72.6434 %
Kstar	72.9508 %
Bagging	72.4385 %
J48	71.1006 %

Στην περίπτωση αυτή ο Naive Bayes μας δίνει το καλύτερο αποτέλεσμα με ποσοστό 73.2818% ενώ ο J48 μας δίνει 71.1006.

ταξινόμηση για τα δεδομένα που αφορούν την ικανοποίηση από το προσωπικό στο μετρό

Algorithm	Correct classified inctances
Naive Bayes	73.5868 %
Ibk	73.0359 %
Kstar	74.1028 %
Bagging	75.0727 %
J48	72.9389 %

Στην περίπτωση αυτή ο Bagging μας δίνει το καλύτερο αποτέλεσμα με ποσοστό 75.0727 % ενώ ο J48 μας δίνει 72.9389.

Γνωρίζοντας πια τα αποτελέσματα για τα τρία πρώτα σετ δεδομένων τα οποία έχουν κοινά μεταξύ τους αφού όλα αποτελούνται από τέσσερις κλάσεις έχουμε να κάνουμε ορισμένες παρατηρήσεις. Αρχικά πρέπει να τονίσουμε ότι τα χαρακτηριστικά τα οποία καθορίζουν τις κλάσεις δεν είναι ιδιαίτερα πολλά (3 και 4), με αποτέλεσμα ο διαχωρισμός των κλάσεων να είναι δύσκολος, οπότε τα ποσοστά εύστοχης ταξινόμησης είναι αρκετά ικανοποιητικά. Όσον αφορά τους αλγόριθμους, βλέπουμε ότι ο Naive Bayes και ο Ibk είναι σταθεροί και στις 3 περιπτώσεις αφού ο πρώτος κυμαίνεται από 72.8863% έως 73.5868% και ο δεύτερος από 72.6434% έως 73.0359% κάτι που μας επιβεβαιώνει για ακόμα μία φορά το πόσο αξιόπιστοι αλγόριθμοι είναι για ταξινόμηση. Επίσης παρατηρούμε ότι οι Kstar και Bagging μας δίνουν σχετικά κοντινά ποσοστά τα οποία στην πρώτη και την τρίτη περίπτωση είναι υψηλά και στην δεύτερη είναι φανερά χαμηλοτερά. Το γεγονός αυτό οφείλεται πιθανότατα στο ότι το δεύτερο σετ δεδομένων μας, περιέχει ένα χαρακτηριστικό περισσότερο από τα άλλα σετ κι έτσι δυσκολεύονται περισσότερο εξαιτίας της πολυπλοκότητας να διαχωρίσουν σωστά τις κλάσεις. Ο αλγόριθμος J48 ο οποίος και θα μας βοηθήσει στην συνέχεια της εργασίας παρουσιάζει όχι τόσο υψηλή απόδοση

σχετικά με τους άλλους αλγόριθμους καθώς κυμαίνεται περίπου από 71% έως 73%. Τέλος τα καλύτερα αποτελέσματα τα βρήκαμε για το τρίτο σετ δεδομένων κάτι που μας δηλώνει ότι τα χαρακτηριστικά των στιγμιοτύπων καθορίζουν καλύτερα τις κλάσεις από τα χαρακτηριστικά των άλλων σετ.

ταξινόμηση για τα δεδομένα που αφορούν την συνολική ικανοποίηση

Algorithm	Correct classified inctances
Naive Bayes	41.6748 %
Ibk	40.2142 %
Kstar	39.0458 %
Bagging	39.63 %
J48	40.3116 %

Στην τελευταία περίπτωση έχουμε να πειραματιστούμε με δεδομένα τα οποία ανήκουν σε 10 κλάσεις. Αυτό σημαίνει ότι εδώ τα ποσοστά μας θα είναι πολύ διαφορετικά σε σχέση με τα προηγούμενα σετ και προφανώς πολύ πιο χαμηλά. Στην περίπτωση αυτή ο Naive Bayes μας δίνει το καλύτερο αποτέλεσμα με ποσοστό 41.2142% ενώ ο J48 μας δίνει 40.3116.

ΚΕΦΑΛΑΙΟ 4: ΑΣΑΦΗΣ ΛΟΓΙΚΗ

4.1 Εισαγωγή και ιστορική αναδρομή

Τα τελευταία χρόνια ο αριθμός και η ποικιλία των εφαρμογών της ασαφής λογικής έχουν αυξηθεί σημαντικά. Η ασαφής λογική είναι ένα ευρύ επιστημονικό πεδίο το οποίο αφορά την αναπαράσταση αδιευκρίνιστης ή ανακριβούς γνώσης και δημιουργήθηκε από την ανάγκη για παράκαμψη της αυστηρής παραδοσιακής δυαδικής λογικής, στην οποία υπάρχουν μόνο οι καταστάσεις του αληθούς ή του ψευδούς. Επίσης μπορεί εύκολα να συνδυαστεί με άλλες σημαντικές μεθόδους όπως είναι οι γενετικοί αλγόριθμοι και τα νευρωνικά δίκτυα. Ασχολείται με την ασάφεια της γνώσης και όχι με την τυχαιότητά της. Χρησιμοποιεί την έννοια του βαθμού συμμετοχής/αλήθειας και όχι τον κάθετο διαχωρισμό αλήθειας-ψεύδους. Οι τιμές βαθμού αλήθειας ορίζονται από τις συναρτήσεις συμμετοχής και παίρνουν τιμές μεταξύ 0 και 1 ώστε να περιγράψουν κατά πόσο συμμετέχει κάποιο αντικείμενο σε ένα ασαφές σύνολο.

Τον όρο «ασαφή λογική» (fuzzy logic) εισήγαγε το 1962 με άρθρο του ο Zadeh, ο οποίος αναφέρθηκε στην αναγκαιότητα για τη δημιουργία μίας μαθηματικής θεωρίας που θα επεξεργάζεται ασαφείς-ανακριβείς έννοιες, οι οποίες δεν είναι δυνατό να μοντελοποιηθούν με τη θεωρία των πιθανοτήτων [Zadeh, (1962)]. Η μαθηματική θεμελίωση της ασαφούς λογικής επιτεύχθηκε με την διατύπωση της θεωρίας των ασαφών συνόλων από τον Zadeh μερικά χρόνια αργότερα (1965).

4.2 Περιγραφή της ασαφής λογικής

Για να καταλάβουμε για ποιον λόγο αναπτύχθηκε τόσο πολύ η ασαφής λογική πρέπει να καταλάβουμε αρχικά τι ακριβώς κάνει και ποιος ο σκοπός χρήσης της. Μπορούμε να πούμε ότι η ασαφής λογική έχει δύο διαφορετικές έννοιες. Με μια πρόχειρη επεξήγηση, η ασαφής λογική είναι ένα λογικό σύστημα το οποίο αποτελεί

μια προέκταση της λογικής πολλών τιμών. Ωστόσο με μία ευρύτερη ματιά μπορούμε να πούμε ότι αποτελεί ένα συνώνυμο της θεωρίας των ασαφών συνόλων, μια θεωρία η οποία συσχετίζει κλάσεις αντικειμένων με όρια γραφικών συναρτήσεων στις οποίες η ιδιότητες εξαρτώνται από το βαθμό συμμετοχής. Έτσι η ασαφής λογική διαφέρει πρακτικά και ουσιαστικά από άλλες παραδοσιακές μεθόδους λογικής πολλών τιμών.

Ένα βασικό συστατικό της ασαφής λογικής το οποίο παίζει σημαντικό ρόλο στις περισσότερες εφαρμογές είναι οι χρησιμοποίηση των κανόνων if-then ή αλλιώς fuzzy rules. Αν και υπάρχουν διάφορα συστήματα που βασίζονται σε κανόνες, αυτό που έλειπε ήταν ένας μηχανισμός χειρισμού ασαφών προγενέστερων στοιχείων (antecedent) με ασαφή επακόλουθα στοιχεία (antecedent). Αυτοί οι μηχανισμοί μας παρέχονται με τον υπολογισμό ασαφών κανόνων.

Η ασάφεια είναι μια έννοια που σχετίζεται με την ποσοτικοποίηση της πληροφορίας και οφείλεται σε αβεβαιότητα, δηλαδή έλλειψη ακριβούς πληροφορίας.

Οι κυριότερες πηγές αβεβαιότητας είναι:

- *Ανακριβή δεδομένα* (*imprecise data*): π.χ. από ένα όργανο περιορισμένης ακρίβειας.

- *Ελλιπή δεδομένα* (*incomplete data*): π.χ. σε ένα σύστημα ελέγχου, κάποιοι αισθητήρες τίθενται εκτός λειτουργίας και πρέπει να ληφθεί άμεσα μία απόφαση με τα δεδομένα από τους υπόλοιπους.

- Υποκειμενικότητα ή/και ελλείψεις στην περιγραφή της γνώσης: π.χ. η υιοθέτηση ευρεστικών μηχανισμών εισάγει πολλές φορές υποκειμενικότητα.

Το πιο γνωστό και απλό παράδειγμα που χρησιμοποιείται για την ασάφεια είναι το παράδειγμα του ύψους ενός ατόμου. Έτσι λοιπόν η φράση «Ο Χρήστος είναι ψηλός» ή «Ο Χρήστος είναι κοντός», μας δίνει την δυνατότητα να βγάλουμε κάποια συμπεράσματα και αποφάσεις για το ύψος του Χρήστου αν και δεν μπορούμε με ακρίβεια να το προσδιορίσουμε. Το πρόβλημα σε αυτές τις περιπτώσεις δεν οφείλεται

τόσο στις έννοιες που χρησιμοποιούνται, όσο στην αντίληψη που έχει ο καθένας για τέτοιους λεκτικούς προσδιορισμούς ποσοτικών μεγεθών. Κατά συνέπεια, η ασάφεια είναι ένα εγγενές χαρακτηριστικό της γλώσσας.

Στο παράδειγμά μας, αν θεωρήσουμε ότι ψηλό είναι οποιοδήποτε άτομο έχει ύψος πάνω από 1.80 μέτρα και κοντό οποιοδήποτε άτομο έχει ύψος κάτω από 1.70, δεν είναι απόλυτα σωστό να βγει το συμπέρασμα ότι κάποιος άνθρωπος με ύψος 1.79 μέτρα δεν είναι ψηλός ούτε ένα άτομο με 1.75 είναι ψηλός αλλά θα έχουμε μια πληροφορία σχετικά με το πόσο ψηλός η κοντός είναι. Το σύνολο το οποίο χαρακτηρίζει το ύψος αποτελεί ένα ασαφές σύνολο (fuzzy set) με την έννοια ότι δεν υπάρχουν αυστηρά κριτήρια που να καθορίζουν το κατά πόσο κάποιος είναι απόλυτα κοντός ή ψηλός. Η ασαφής λογική (fuzzy logic) και η θεωρία των ασαφών συνόλων (fuzzy set theory) παρέχουν ένα πλαίσιο χειρισμού της ασάφειας και ένα πλαίσιο συλλογιστικής βασισμένης στην ασάφεια.

4.3 Πλεονεκτήματα-μειονεκτήματα ασαφής λογικής

Πλεονεκτήματα

Η ασαφής λογική χρησιμοποιείται για πολλούς λόγους μερικοί από τους οποίους παρουσιάζονται παρακάτω:

- Είναι πολύ εύκολο να κατανοηθεί καθώς το μαθηματικό υπόβαθρο χρήσης της είναι αρκετά απλό και είναι μια διαισθητική προσέγγιση χωρίς πολύπλοκες διαδικασίες
- Είναι ανεκτική σε ελλιπή δεδομένα
- είναι εύκολη η υλοποίηση της ασαφούς λογικής χρησιμοποιώντας λογισμικό στους υπάρχοντες επεξεργαστές ή σε ειδικά σχεδιασμένο υλικό (hardware). Έτσι οι λύσεις που βασίζονται στην ασαφή λογική έχουν επικρατήσει σε ένα μεγάλο φάσμα εφαρμογών (όπως οι οικιακές εφαρμογές) σε σχέση με τις παραδοσιακές μεθόδους.

- Μπορεί να συνδυαστεί με άλλες τεχνικές ελέγχου. Τα ασαφή συστήματα όχι μόνο μπορούν να αντικαταστήσουν άλλες μεθόδους αλλά μπορούν να τις βελτιώσουν αλλά και να απλουστεύσουν την υλοποίησή τους.
- Η ασαφής λογική μπορεί να μετατρέπει πολύπλοκα προβλήματα σε απλούστερα χρησιμοποιώντας τη λογική της προσέγγισης. Ένα σύστημα περιγράφεται με τους κανόνες και τις συναρτήσεις συμμετοχής της ασαφούς λογικής χρησιμοποιώντας την γλώσσα των ανθρώπων και τις λεκτικές μεταβλητές. Έτσι κάποιος μπορεί εύκολα και αποτελεσματικά να χρησιμοποιήσει την γνώση του για να περιγράψει την συμπεριφορά του συστήματος.
- Μπορεί να μοντελοποιήσει μη γραμμικές συναρτήσεις μεγάλης πολυπλοκότητας. Μπορούμε να δημιουργήσουμε ασαφή συστήματα για να συνδυάσουμε δεδομένα εισόδου με δεδομένα εξόδου για οποιοδήποτε σετ δεδομένων.

Μειονεκτήματα

- Όσο η πολυπλοκότητα του συστήματος αυξάνεται, είναι δύσκολη η εύρεση του σωστού συνόλου κανόνων και συναρτήσεων συμμετοχής και τον αριθμό τους που απαιτούνται για την αποτελεσματική περιγραφή του συστήματος. Έτσι απαιτείται σημαντική έρευνα και προσπάθεια ώστε οι κανόνες και οι συναρτήσεις συμμετοχής να ρυθμιστούν για ένα καλό αποτέλεσμα.

- Στα πολύπλοκα συστήματα που χρειάζονται πολλοί κανόνες είναι πολύ δύσκολο να επιτευχθεί μία συσχέτιση μεταξύ τους και η ικανότητα προς συσχέτιση των κανόνων μειώνεται σημαντικά όταν το πλήθος τους αρχίζει να γίνει μεγάλο(περίπου 20 κανόνες).

- Η ασαφής λογική χρησιμοποιεί ευριστικούς αλγόριθμους για την αποασαφοποίηση και την εκτίμηση των κανόνων. Το μειονέκτημα των ευριστικών αλγορίθμων είναι ότι οι λύσεις που δίνουν δεν μπορούν να ανταποκριθούν αποτελεσματικά σε όλες τις πιθανές συνθήκες. Είναι

σημαντικό να υπάρχει η δυνατότητα της γενίκευσης ώστε να μπορεί το μοντέλο να ανταποκριθεί σε καταστάσεις και δεδομένα που δεν έχει ξαναδεί

4.4 Ασαφή σύνολα

Ένα κλασσικό ή ξερό(crisp) σύνολο, έστω Α, ορίζεται μέσω της χαρακτηριστικής συνάρτησης:

$$f_A(x): X \to \{0,1\} \qquad \text{όπου} \qquad f_A(x) = \begin{cases} 1, & \alpha v\ x \in A \\ 0, & \alpha v\ x \notin A \end{cases}$$

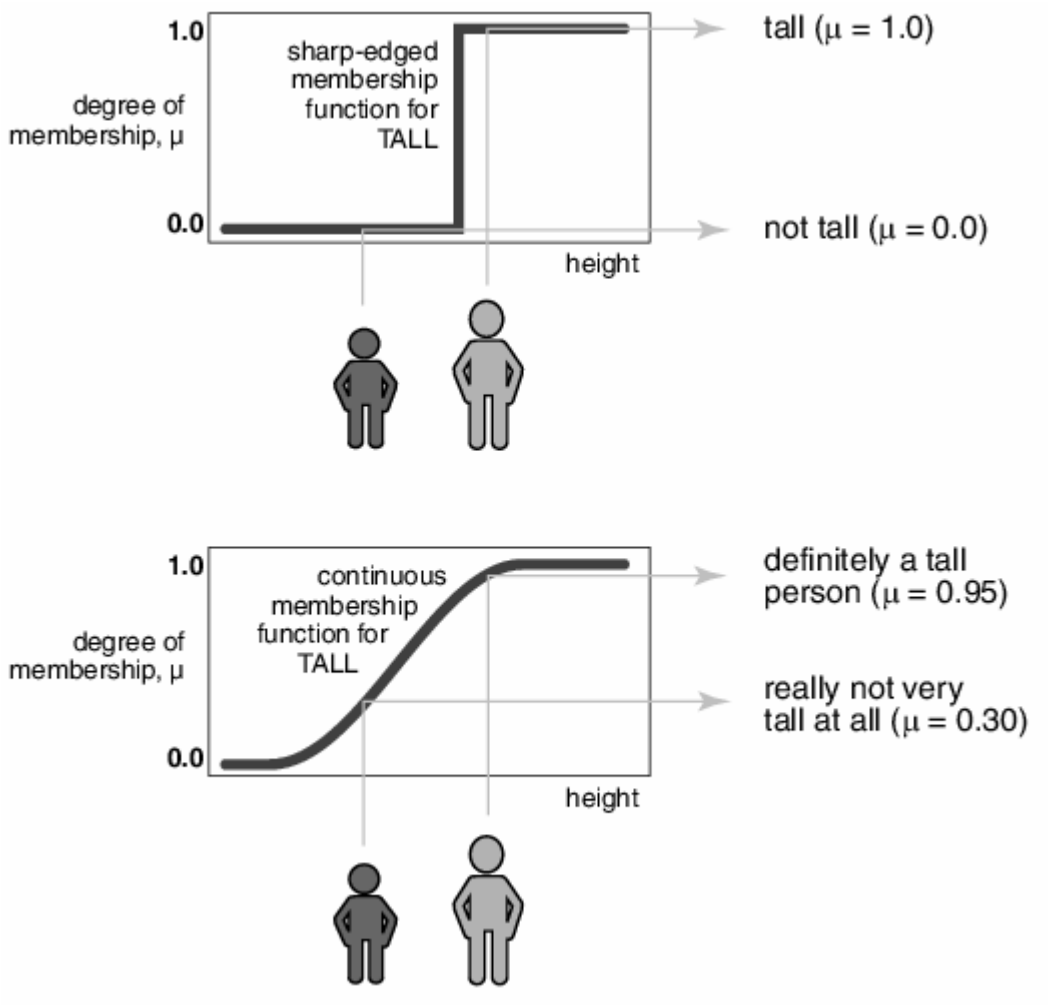
Όπου Χ είναι ο κόσμος αναφοράς (universe of discourse) ή σύνολο αναφοράς και x ένα στοιχείο του Χ.

Ένα ασαφές σύνολο (fuzzy set),έστω Α, ορίζεται μέσω της συνάρτησης συμμετοχής (membership function):

$$\mu_A(x): X \to [0,1] \qquad \text{όπου} \qquad \mu_A(x) = \begin{cases} 1 & \alpha v\ x\ ολικά\ στο\ A \\ 0 & \alpha v\ x\ καθόλου\ στο\ A \\ (0,1) & \alpha v\ x\ μερικώς\ στο\ A \end{cases}$$

όπου το $\mu_A(x)$ είναι ένας πραγματικός αριθμός ($0 <= \mu_A(x) <= 1$) που παριστάνει τον βαθμό στον οποίο το x είναι στοιχείο του Α και ονομάζεται βαθμός συμμετοχής (degree of membership) ή βαθμός αλήθειας (degree of truth) ή τιμή συμμετοχής (membership value).

Όπως αναφέραμε και πιο πριν, το ποιο γνωστό παράδειγμα ασαφούς λογικής είναι αυτό με το σύνολο των ψηλών και κοντών ανθρώπων. Στο παρακάτω σχήμα φαίνεται η διαφορά του κλασσικού συνόλου, που παίρνει τιμές 0 και 1, με το ασαφές σύνολο όπου παίρνει οποιεσδήποτε τιμές ανάμεσα στο 0 και 1. Στο πρώτο σχήμα η πληροφορία που παίρνουμε από τον βαθμό συμμετοχής μ της συνάρτησης είναι ότι το άτομο θα είναι απόλυτα 'κοντός' ή 'ψηλός ' κάτι που δείξαμε παραπάνω ότι δεν είναι απολύτως σωστό. Αντίθετα, στο δεύτερο σχήμα που αποτελείται από μια συνεχή συνάρτηση μας δίνεται μία ποιο σχετική και συγκεκριμένη πληροφορία που μας προσδιορίζει το κατά πόσο 'κοντό' ή 'ψηλό' είναι το συγκεκριμένο άτομο.



Εικόνα 4.1 παραδείγματα κλασσικού και ασαφούς συνόλου

4.5 Πράξεις σε ασαφή σύνολα

Ένα ασαφές σύνολο Α θεωρείται *κενό* εάν η συνάρτηση συμμετοχής του είναι μηδενική παντού, δηλαδή $A = \varnothing \Leftrightarrow \mu_A(x) = 0 \quad \forall x \varepsilon X$ (4.5.1)

Ισότητα

Λέμε πως τα δύο σύνολα Α και Β είναι ίσα και γράφουμε Α=Β αν και μόνο αν ισχύει

$$\mu_A(x) = \mu_B(x) , \forall x \varepsilon X \qquad (4.5.2)$$

Ένωση και τομή ασαφών συνόλων

Η κλασσική ένωση και τομή των συνόλων μπορεί να επεκταθεί στα ασαφή σύνολα ως εξής:

$\forall x \varepsilon X$,

Ένωση A,B: $A \cup B$, κατά πόσο ένα στοιχείο βρίσκεται σε ένα από τα δύο σύνολα

$$\mu_{A \cup B}(X) = \mu_A(x) \cup \mu_B(x) = \max \{ \mu_A(x) , \mu_B(x)\} \qquad (4.5.3)$$

Τομή A,B: $A \cap B$, κατά πόσο ένα στοιχείο βρίσκεται και στα δύο σύνολα

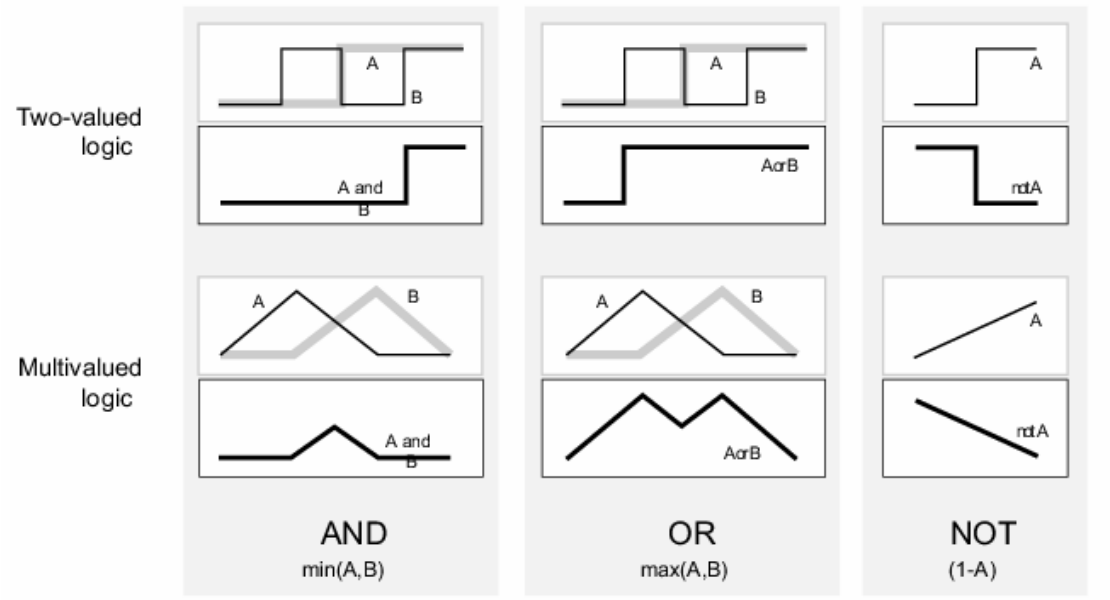
$$\mu_{A \cap B}(X) = \mu_A(x) \cap \mu_B(x) = \min \{ \mu_A(x) , \mu_B(x)\} \qquad (4.5.4)$$

Συμπλήρωμα ασαφούς συνόλου

Το συμπλήρωμα (*complement*) , $\mu_{\bar{A}}(x)$ του $\mu_A(x)$ ορίζεται με την ως εξής:

$\mu_{\bar{A}}(x)=1- \mu_A(x)$ (Κατά πόσο τα στοιχεία δεν ανήκουν στο A) (4.5.5)

Παρακάτω απεικονίζονται η τομή $A \cap B$, η ένωση $A \cup B$ και το συμπλήρωμα $\bar{A}$ για ένα κλασσικό και ένα ασαφές σύνολο



Εικόνα 4.2 πράξεις ανάμεσα σε σύνολα

Υποσύνολο ασαφούς συνόλου

Λέμε πως το ασαφές σύνολο Α είναι υποσύνολο του ασαφούς συνόλου Β και γράφουμε $A \subseteq B$ αν ισχύει,

$$\mu_A(x) \leq \mu_B(x), \quad \forall x \varepsilon X \qquad (4.5.6)$$

Διαφορά

Η πράξη της διαφοράς συμβολίζεται με A-B και ορίζεται από το

$$\mu_B(x) - \mu_A(x) = \min(\mu_B(x), \mu_{\bar{A}}(x)) = \mu_{\bar{A} \cap B}(X), \quad \forall x \varepsilon X \qquad (4.5.7)$$

4.6 Ιδιότητες πράξεων ασαφών συνόλων

Θεωρώντας τα ασαφή σύνολα A, B και C ισχύουν οι εξής ιδιότητες:

- **Αντιμεταθετικότητα:** $A \cup B = B \cup A$ και $A \cap B = A \cap B$ (4.6.1)
- **Προσεταιριστικότητα:** $A \cup (B \cup C) = (A \cup B) \cup C$ και $A \cap (B \cap C) = (A \cap B) \cap C$ (4.6.2)
- **Ανακλαστικότητα:** $A \cup A = A$ και $A \cap A = A$ (4.6.3)

- **Επιμεριστικότητα:** $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ (4.6.4.a)

  $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ (4.6.4.b)

- **Ανακλαστικότητα:** $A \cap \emptyset = \emptyset$ και $A \cup X = X$ (4.6.5.a)

  $A \cup \emptyset = A$ και $A \cap X = A$ (4.6.5.b)

- **Επαναφορά:** $\overline{\overline{A}} = A$ (4.6.6)

- **Μεταβατικότητα:** αν $A \subseteq B$ και $B \subseteq C$ τότε $A \subseteq C$ (4.6.7)

- **Απορροφητική ιδιότητα**: $A \cap (A \cup B) = A$ και $A \cup (A \cap B) = A$ (4.6.8)

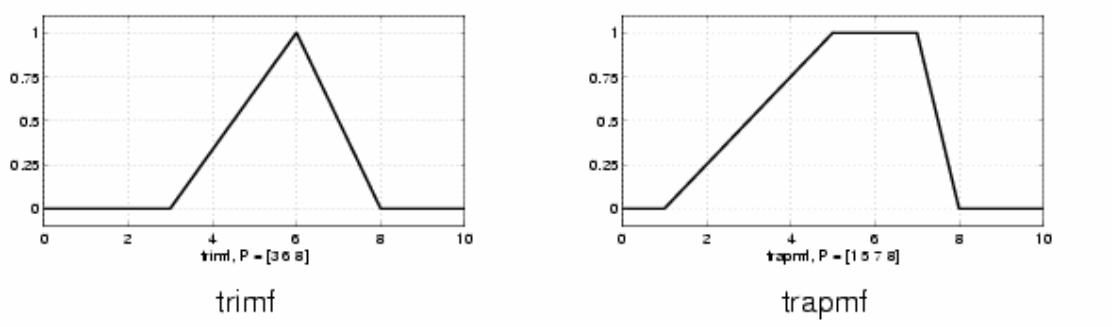
- **Νόμοι De Morgan:** $\overline{A \cap B} = \overline{A} \cup \overline{B}$ (4.6.9.a)

  $\overline{A \cup B} = \overline{A} \cup \overline{B}$ (4.6.9.b)

4.7 Συναρτήσεις συμμετοχής και οι μορφές τους

Συνάρτηση συμμετοχής (membership function) ονομάζεται η συνάρτηση της οποίας οι γραμμές αντιστοιχίζουν κάθε σημείο στον άξονα x με τον αντίστοιχο βαθμό συμμετοχής στον άξονα y με τιμές από 0 έως 1. Παρακάτω παρουσιάζονται οι πιο συχνές μορφές συναρτήσεων συμμετοχής οι οποίες παρέχονται και από το Toolbox της Matlab .

Οι πιο απλές συναρτήσεις είναι αυτές που αποτελούνται από ευθείες γραμμές και αυτές είναι οι τριγωνομετρική (trimf) και η τραπεζοειδής (trapmf). Η πρώτη καθορίζεται από 3 παραμέτρους και η δεύτερη από 4, οι οποίες καθορίζουν τις συντεταγμένες στον οριζόντιο άξονα των γωνιών των αντίστοιχων συναρτήσεων συμμετοχής.



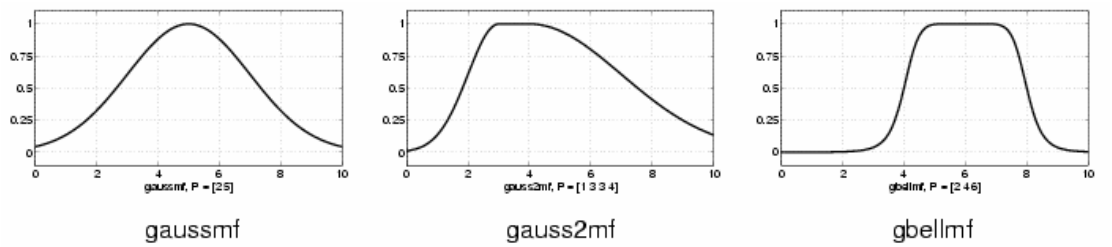
Χάρη στην απλή μορφή τους και την υπολογιστική αποτελεσματικότητα, τόσο οι τριγωνικές όσο και οι τραπεζοειδείς συναρτήσεις, έχουν χρησιμοποιηθεί ευρέως στον προσδιορισμό ασαφών συνόλων. Όμως από τη στιγμή που κατασκευάζονται από τμήματα ευθειών, δεν είναι ομαλές στα άκρα έτσι όπως αυτά προσδιορίζονται από τις παραμέτρους. Παρακάτω αναλύονται εναλλακτικοί τύποι συναρτήσεων συμμετοχής που χαρακτηρίζονται από την ομαλότητά τους..

Οι επόμενες τρεις συναρτήσεις είναι βασισμένες πάνω στην Γκαουσιανή κατανομή. Η Gaussian (gaussmf) συνάρτηση ορίζεται από την σχέση

$$f = e^{\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2} \qquad (4.7.1)$$

όπου το c δηλώνει την μέση τιμή και το σ το πλάτος της και θα είναι οι δύο παράμετροι που θα προσδιορίζουν την συνάρτηση. Οι άλλες δύο συναρτήσεις συμμετοχής είναι οι gauss2mf που είναι παρόμοια με την gaussmf και προσδιορίζεται από 4 παραμέτρους , και η συνάρτηση γενικευμένης καμπανοειδούς μορφής (gbellmf) που αποτελείται από 3 παραμέτρους και ορίζεται από την σχέση

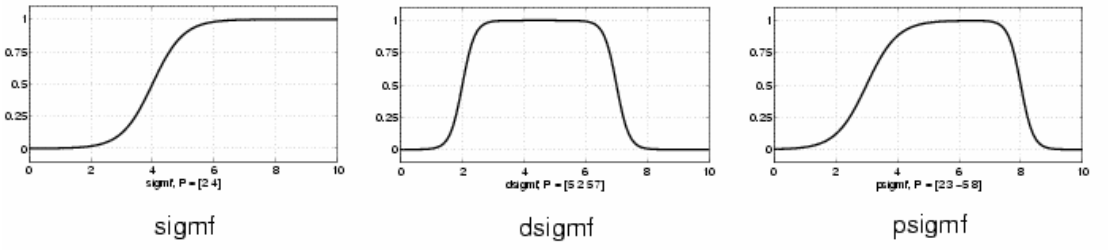
$$f = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}} \qquad (4.7.2)$$



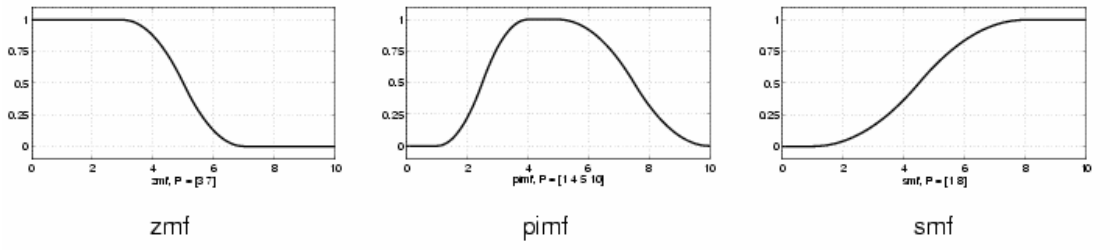
Η Gaussian και η καμπανοειδείς συναρτήσεις λόγω της ομαλότητας και του ακριβούς ορισμού τους τείνουν να χρησιμοποιούνται ολοένα και περισσότερο στον προσδιορισμό των ασαφών συνόλων. Όμως εξαιτίας αυτής της ιδιότητας είναι δύσκολο να προσδιορίσουν ασύμμετρες συναρτήσεις συμμετοχής οι οποίες είναι χρήσιμες για σημαντικές εφαρμογές. Έτσι χρησιμοποιούνται οι σιγμοειδείς (sigmoidal) συναρτήσεις συμμετοχής οι οποίες είναι συνήθως ασύμμετρες και κλειστές. Η σιγμοειδής συνάρτηση συμμετοχής ορίζεται από την σχέση:

$$f = \frac{1}{1 + e^{-a(x-c)}} \tag{4.7.3}$$

Το α καθορίζει την κλίση στο σημείο διασταύρωσης c, και ανάλογα με την τιμή που θα πάρει , η σιγμοειδής MF είναι ανοιχτή αριστερά ή ανοιχτή δεξιά. Επίσης με συνδυασμό σιγμοειδών συναρτήσεων μπορούμε να δημιουργήσουμε άλλες όμοιες. Έτσι έχουμε τις συναρτήσεις συμμετοχής sigmf , dsigmf, psigmf .



Τέλος έχουμε και τις πολυωνυμικές συναρτήσεις συμμετοχής. Αυτές είναι οι Zmf, Pmf, Smf και πήραν την ονομασία τους λόγω του σχήματος των γραφικών τους παραστάσεων. Η Zmf συνάρτηση ανοίγει από την αριστερή μεριά, η Pmf είναι 0 στα άκρα με μέγιστο περίπου στο μέσω και η Smf είναι ανοιχτή στην δεξιά μεριά.



4.8 Ασαφείς προτάσεις και κανόνες

Ασαφής πρόταση είναι αυτή που θέτει μια τιμή σε μια ασαφή μεταβλητή. Για παράδειγμα στην ασαφή πρόταση "Το ύψος του Χρήστου είναι μέτριο", το "ύψος" είναι η ασαφής μεταβλητή και "μέτριο" είναι ένα ασαφές σύνολο που είναι η τιμή της μεταβλητής.

Ασαφής κανόνας (*fuzzy rule*): είναι μία υπό συνθήκη έκφραση που συσχετίζει δύο ή περισσότερες ασαφείς προτάσεις. Το πιο απλό παράδειγμα ενός ασαφή κανόνα είναι το 'if x is A then y is B' όπου το A, B είναι οι ασαφείς μεταβλητές που έχουν ως τιμές τα ασαφή σύνολα x,y.

Η αναλυτική περιγραφή ενός ασαφούς κανόνα if-then είναι μια ασαφής σχέση R(x,y), που ονομάζεται σχέση συνεπαγωγής (implication relation), η οποία προκύπτει με κατάλληλο συνδυασμό του if και του then τμήματος του κανόνα, δηλαδή των συναρτήσεων συμμετοχής των ασαφών συνόλων A και B. Στη γενική της μορφή η συνάρτηση συνεπαγωγής ορίζεται ως:

$$\mathbf{R(x,y)\equiv\mu(x,y)=\varphi(\mu A(x),\mu B(y))} \qquad \mathbf{(4.8.1)}$$

Η συνάρτηση φ ονομάζεται τελεστής συνεπαγωγής (implication operator) και υποδεικνύει τον ακριβή τρόπο με τον οποίο πρέπει να συνδυαστούν οι συναρτήσεις.

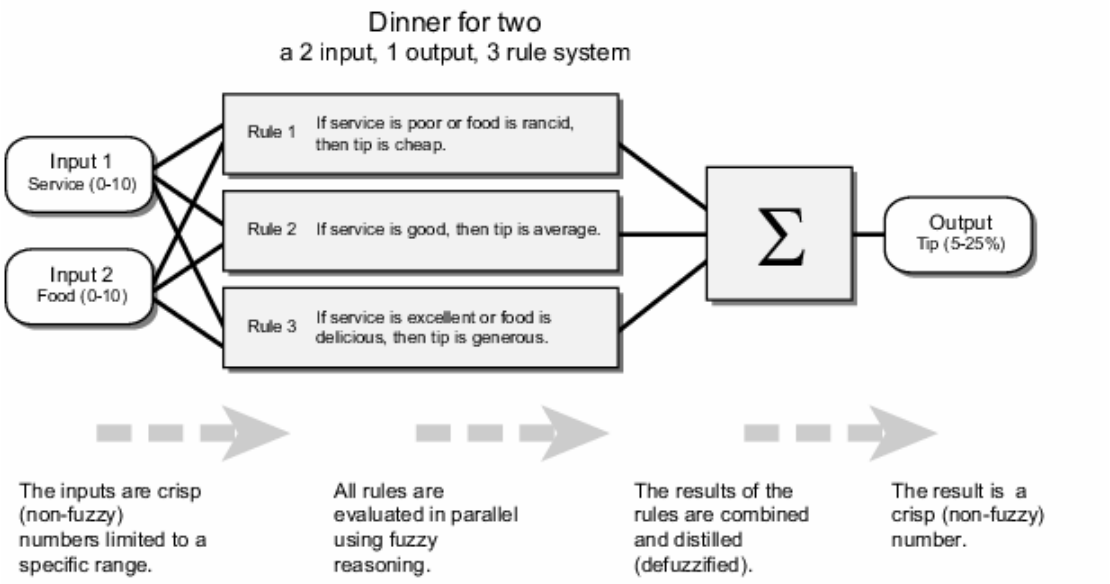
4.9 Λειτουργία ενός ασαφούς συστήματος

Τα ασαφή συστήματα (fuzzy interference systems) είναι τα συστήματα τα οποία αναλαμβάνουν τον σχεδιασμό της εξόδου από μια δοθέντα είσοδο. Η διαδικασία αυτή περιλαμβάνει όλα τα μέρη που περιγράψαμε προηγουμένως : ασαφή σύνολα, πράξεις ανάμεσα σε ασαφή σύνολα και ασαφείς κανόνες και προτάσεις. Υπάρχουν δύο διαφορετικοί τύποι ασαφών συστημάτων που μας παρέχει η Matlab, το Sygeno-type και το Mamdani –type με το οποίο και θα δουλέψουμε. Η μέθοδος Mamdani είναι μια από τις πιο διαδεδομένες και ανάμεσα στα πρώτα συστήματα ελέγχου που χρησιμοποίησαν την θεωρία των ασαφών συνόλων. Προτάθηκε το 1975

από τον Ebrahim Mamdani όπου προσπάθησε να ελέγξει έναν συνδυασμό λέβητα με ατμομηχανή συνθέτοντας ένα σετ από γλωσσικούς κανόνες ελέγχου.

Έστω ότι έχουμε τους παρακάτω τρεις κανόνες

1. If the service is poor or the food is rancid, then tip is cheap.
2. If the service is good, then tip is average.
3. If the service is excellent or the food is delicious, then tip is generous.



Εικόνα 4.3

Η διαδικασία της λειτουργίας ενός ασαφούς συστήματος με την μέθοδο Mamdani αποτελείται από τα εξής βήματα:

- **Fuzzify inputs** (ασαφοποίηση εισόδου): Μετατροπή των ξερών-crisp εισόδων σε ασαφείς. Το πρώτο βήμα της διαδικασίας είναι να λάβουμε τις εισόδους και να προσδιορίσουμε τον βαθμό συμμετοχής για κάθε ασαφές σύνολο μέσω της συνάρτησης συμμετοχής.

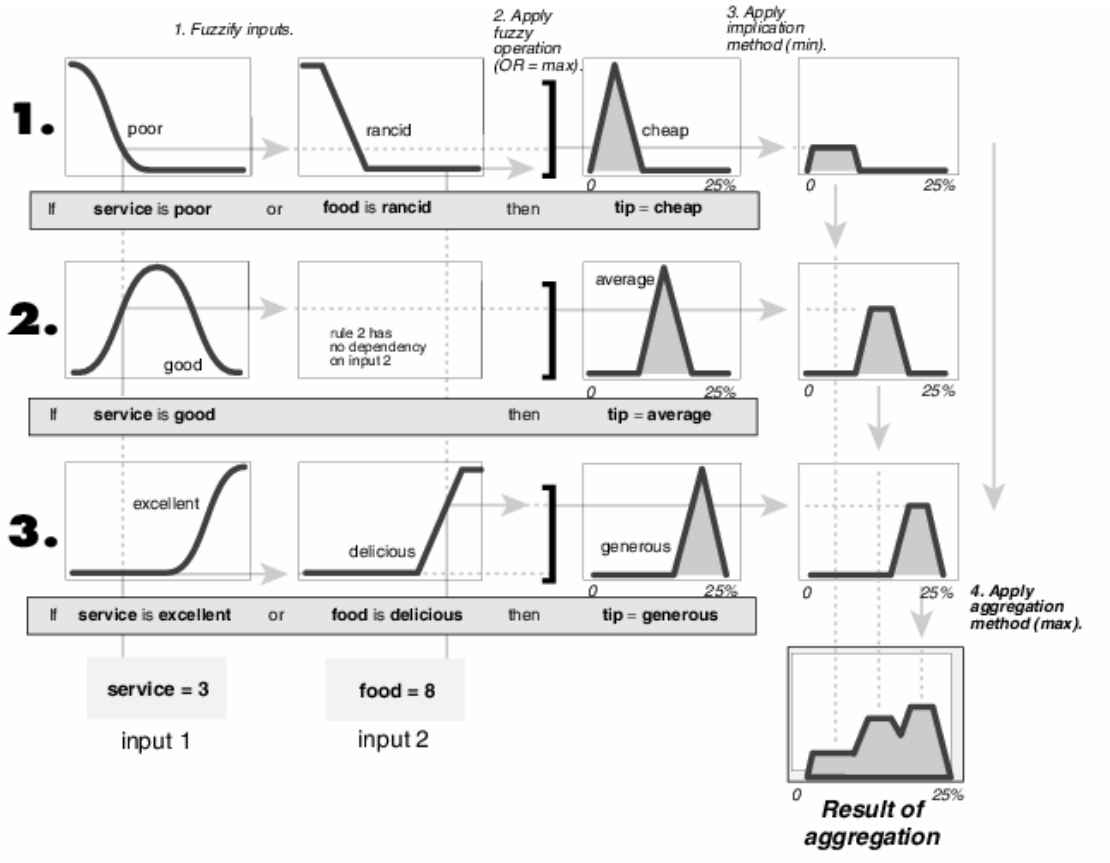
- **Apply Fuzzy operator** (χρήση τελεστών)**:** Μετά την ασαφοποίηση των κανόνων ξέρουμε τον βαθμό συμμετοχής για κάθε if-μέρος. Αν αυτό αποτελείται από 2 ή περισσότερα μέλη, τότε εφαρμόζουμε τον κατάλληλο

τελεστή ανάμεσά τους ο οποίος θα καθορίσει τον συνδυασμένο βαθμό συμμετοχής του if-μέρους που θα χρησιμοποιηθεί στην έξοδο

- **Apply implication Method** (εφαρμογή του συνδυασμένου βαθμού στην συνάρτηση συμμετοχής του συμπεράσματος): Η συνάρτηση συμμετοχής του συμπεράσματος ανασχηματίζεται χρησιμοποιώντας τον αριθμό (βαθμό συμμετοχής) που προέκυψε από το προηγούμενο βήμα. Έτσι παίρνοντας σαν είσοδο τον αριθμό αυτό προκύπτει στην έξοδο ένα καινούργιο ασαφές σύνολο

- **Aggregate All Outputs:** (συνάθροιση όλων των εξόδων). Η συνάθροιση είναι η διαδικασία κατά την οποία τα ασαφή σύνολα τα οποία αναπαριστούν τις εξόδους για κάθε κανόνα συνδυάζονται μεταξύ τους για να εκτιμήσουν το τελικό ασαφές σύνολο που θα χρησιμοποιηθεί για την διαδικασία της αποασαφοποίησης.

Παρακάτω φαίνονται τα βήματα τα οποία περιγράψαμε για το παράδειγμα

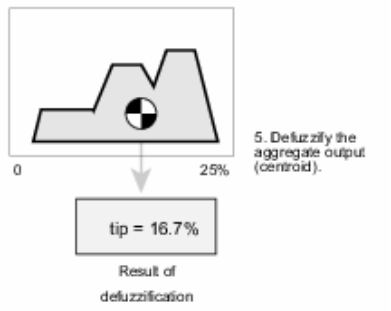


Εικόνα 4.4

- **Defuzzify: (**αποασαφοποίηση). Είναι το τελευταίο βήμα της διαδικασίας, όπου δέχεται σαν είσοδο το ασαφές σύνολο που προέκυψε από την συνάθροιση των εξόδων και το μετατρέπει σε ξερή-crisp τιμή. Η ποιο γνωστή μέθοδος αποασαφοποίησης είναι η centroid ,που βρίσκει την τεταγμένη του κέντρου βάρους της επιφάνειας μεταξύ της καμπύλης και των αξόνων, Αυτή υπολογίζεται από τον τύπο:

$$g = \frac{\int_a^b \mu_A(x) x dx}{\int_a^b \mu_A dx} \qquad (4.9.1)$$

Στο παρακάτω σχήμα φαίνεται το αποτέλεσμα της αποασαφοποίησης που είναι και το τελικό



Εικόνα 4.5

4.10 Υλοποίηση του προβλήματος με χρήση ασαφής λογικής

Μια από της πιο ευρέως χρησιμοποιούμενες εφαρμογές της ασαφής λογικής είναι για την ταξινόμηση (classification) δεδομένων. Στο κεφάλαιο αυτό θα περιγράψουμε την διαδικασία ταξινόμησης των ίδιων δεδομένων που ταξινομήσαμε χρησιμοποιώντας το εργαλείο Weka, με χρήση ασαφής λογικής, ώστε να συγκρίνουμε τα αποτελέσματα και με απώτερο σκοπό την βελτίωσή τους.

Κατά την ταξινόμηση με ασαφή λογική αντιστοιχούμε προσεγγιστικά κάθε κλάση με μία συνάρτηση συμμετοχής η οποία θα καθορίζει την έξοδο του συστήματος. Αντίστοιχα σαν είσοδο σε κάθε χαρακτηριστικό προσαρτούμε διάφορες συναρτήσεις συμμετοχής ανάλογα με το πόσες χρειάζονται για να το περιγράψουν

πλήρως, οι οποίες χρησιμοποιούνται ανάλογα με την τιμή του κάθε χαρακτηριστικού και τον τελεστή τον οποίο τις συνδυάζει. Έτσι για τον κάθε κανόνα θα έχουμε σαν έξοδο τον βαθμό συμμετοχής και την τροποποιημένη συνάρτηση συμμετοχής της κλάσης που προκύπτει για τον εκάστοτε κανόνα σύμφωνα με την διαδικασία η οποία περιγράφηκε στο προηγούμενο κεφάλαιο. Μετά από αυτό το στάδιο γίνεται η συνάθροιση των εξόδων των υποψηφίων κλάσεων ώστε να προκύψει η τελική γραφική παράσταση του ασαφούς συνόλου, από την αποασαφοποίηση της οποίας θα προκύψει η κλάση στην οποία θα ανήκουν τα χαρακτηριστικά.

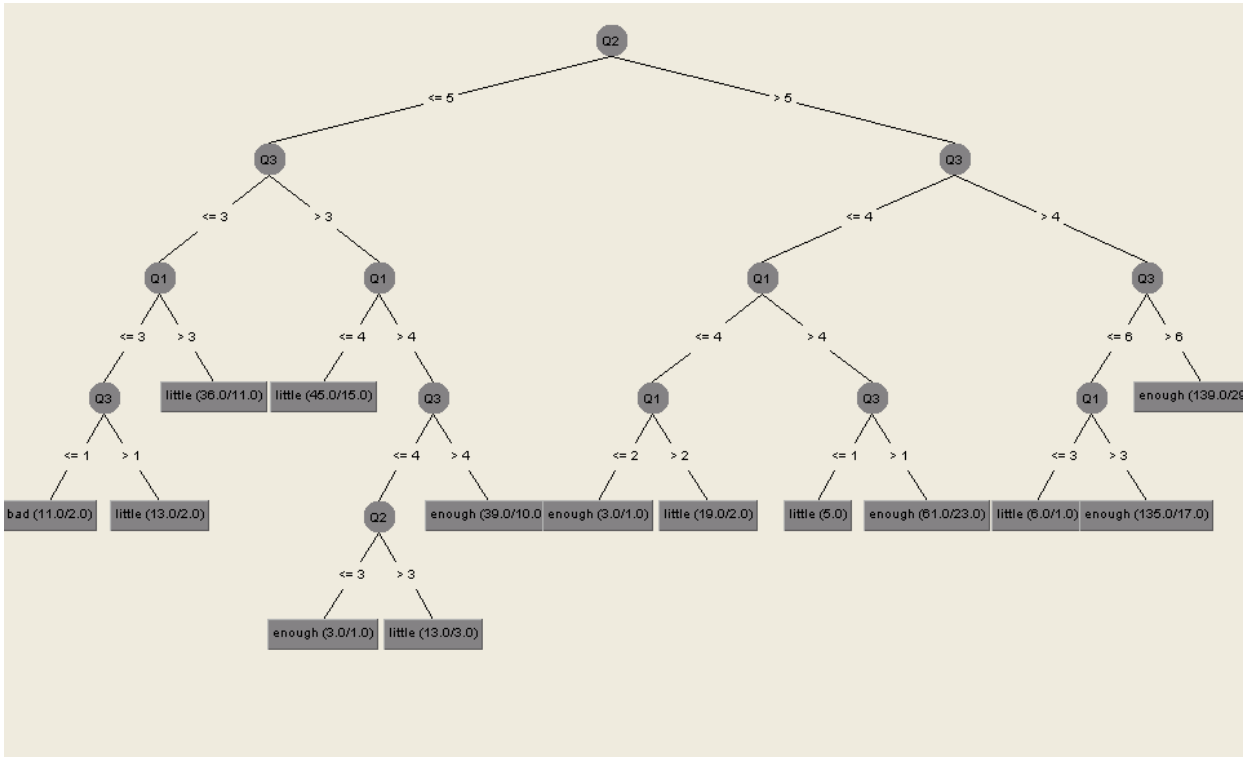
Τα δεδομένα θα είναι διαχωρισμένα σε training και test set ακριβώς όπως και στην διαδικασία ταξινόμησης τους με το εργαλείο weka. Χρησιμοποιώντας το σετ εκπαίδευσης θα πρέπει να ανακαλύψουμε τους κανόνες οι οποίοι προσδιορίζουν την κλάση για στιγμιότυπα εκπαίδευσης και μετά να τους εφαρμόσουμε στα στιγμιότυπα του σετ ελέγχου για να διαπιστώσουμε την αποδοτικότητα του συστήματος στο να προβλέπει τις κλάσεις τους. Το αρχικό πρόβλημα σε αυτό το σημείο ήταν ο προσδιορισμός των κανόνων καθώς ο αριθμός των χαρακτηριστικών για κάθε κλάση ήταν μικρός και αρχικά απαιτούσε την ύπαρξη αρκετών κανόνων. Όπως ξέρουμε ήδη, από έναν αριθμό κανόνων (20) και μετά δεν αποδίδει καλά το σύστημα, οπότε σκοπός είναι να αυξήσουμε την αποδοτικότητα με όσο το δυνατόν λιγότερους κανόνες.

Η λύση στο πρόβλημα αυτό ήρθε χρησιμοποιώντας τους κανόνες οι οποίοι προέκυψαν όταν εφαρμόσαμε στα δεδομένα τον αλγόριθμο δέντρου C4.5(j48) στο εργαλείο weka. Εφαρμόζοντας τον αλγόριθμο αυτό στα δεδομένα εκπαίδευσης προκύπτει ένα δέντρο που αναπαριστά κανόνες οι οποίοι ξεκινώντας από την κορυφή και καταλήγοντας σταδιακά προς τα φύλλα του καθορίζουν την κλάση των χαρακτηριστικών. Η διαδρομή εξαρτάται από τους κόμβους του δέντρου όπου ελέγχουν τις τιμές που παίρνουν τα χαρακτηριστικά και ανάλογα με αυτές υποδεικνύουν την συνέχεια της διαδρομής μέχρι να φτάσουν στα φύλλα. Εκεί παρουσιάζεται η επικρατούμενη κλάση για τις συγκεκριμένες τιμές χαρακτηριστικών μαζί με την πιθανότητά της.

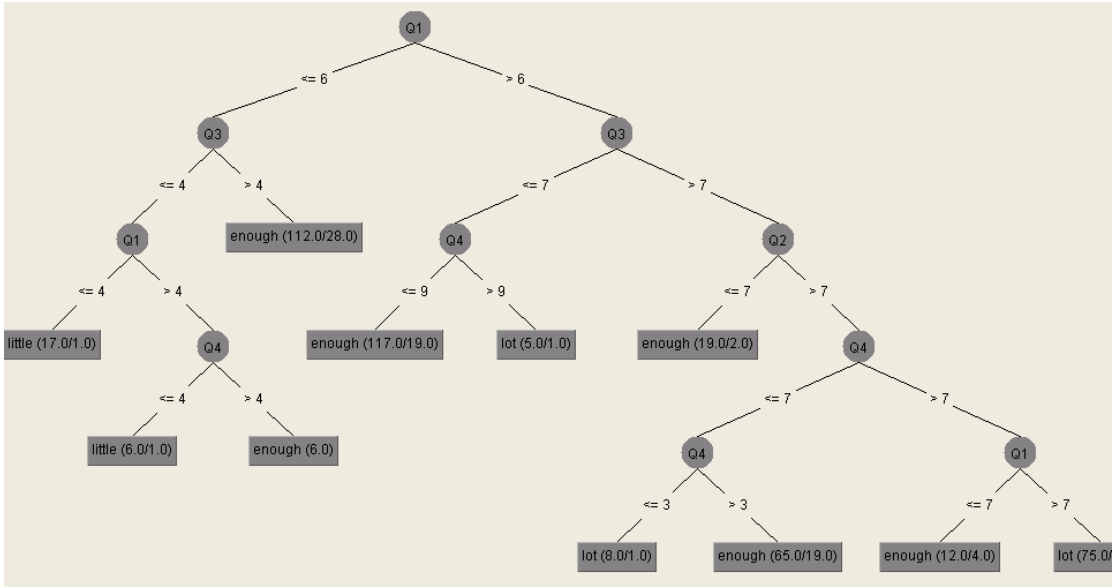
Για παράδειγμα, εάν έχουμε σε ένα φύλλο του δέντρου την πληροφορία enough (61.0/23.0) , αυτό μας δίνει την γνώση ότι αν εφαρμόσουμε τον κανόνα που διαγράφεται στην πορεία του δέντρου τότε η κλάση η οποία θα προκύψει θα είναι η enough με πιθανότητα 61/23. Δηλαδή για κάθε 84 στιγμιότυπα που θα εξεταστούν με τον συγκεκριμένο κανόνα τα 61 θα κατηγοριοποιηθούν στην κλάση enough που προφανώς θα είναι και η σωστή και τα υπόλοιπα 23 θα κατηγοριοποιηθούν σε λανθασμένες κλάσεις.

Εμείς αυτό που θα πράξουμε είναι με δοσμένους τους κανόνες αυτούς να μπορέσουμε να βελτιώσουμε τις αποδόσεις ταξινόμησης που προέκυψαν με το εργαλείο weka. Αυτό θα γίνει χρησιμοποιώντας τις κατάλληλες συναρτήσεις συμμετοχής για τα χαρακτηριστικά και τροποποιώντας τους κανόνες δέντρου ώστε να ελαχιστοποιήσουμε το σφάλμα ταξινόμησης. Η τροποποίηση των κανόνων λαμβάνει χώρα όταν ένας κανόνας δεν μας δίνει ένα ικανοποιητικό αποτέλεσμα. Δηλαδή όταν η επικρατέστερη κλάση προκύπτει με χαμηλό ποσοστό πιθανότητας, τότε ‘σπάμε’ τον κανόνα αυτό σε 2 ή περισσότερους κανόνες έτσι ώστε να μπορούμε να την προσδιορίσουμε καλύτερα αλλά και να μπορέσουμε να προβλέψουμε σε όσο μεγαλύτερο βαθμό μπορούμε την κλάση των στιγμιοτύπων που στον προηγούμενο κανόνα ταξινομήθηκαν εσφαλμένα. Έτσι οι κανόνες αυτοί θα μας δώσουν μεγαλύτερη πληροφορία και συνεπώς καλύτερη δυνατότητα ταξινόμησης. Τέλος αν διαπιστωθεί ότι κάποιος κανόνας δίνει πολύ άσχημα ποσοστά ή ότι η χρήση του δεν είναι αρκετά σημαντική ,τότε τον αγνοούμε και δεν τον χρησιμοποιούμε καθώς θα αποτελούσε επιβάρυνση για το σύστημά μας .Παρακάτω παρουσιάζονται τα δέντρα κανόνων για τα τέσσερα σετ δεδομένων

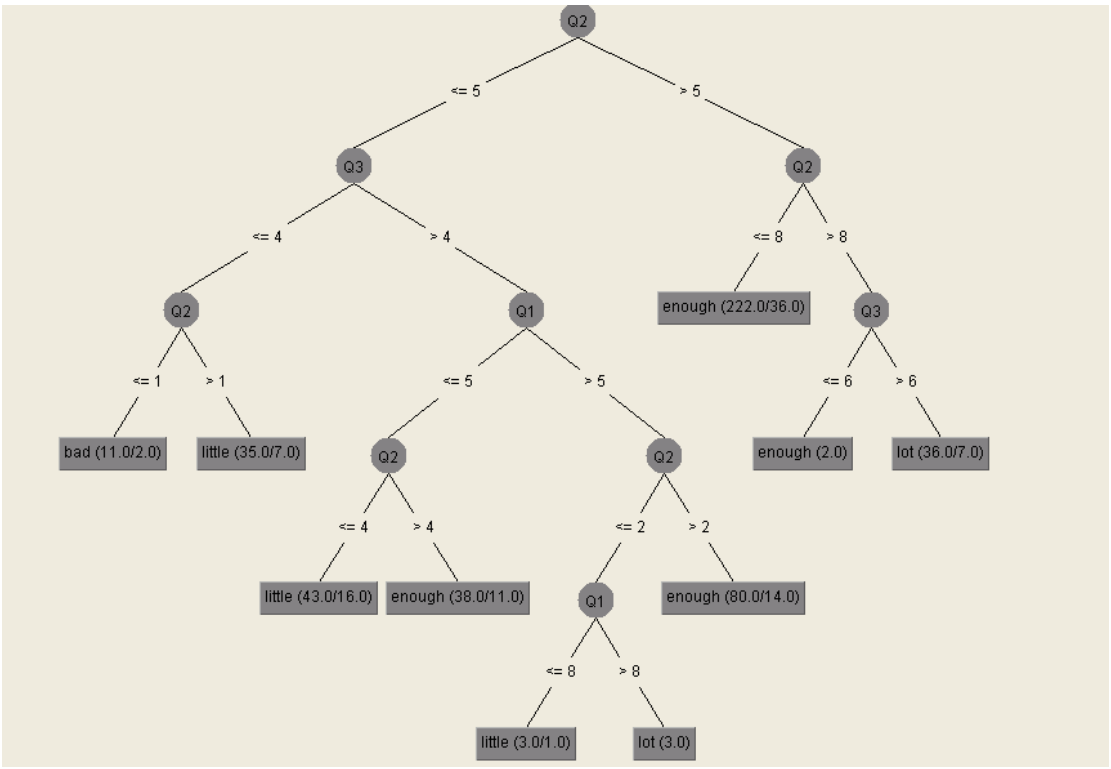
Για τα δεδομένα που αφορούν τις συνθήκες στο μετρό



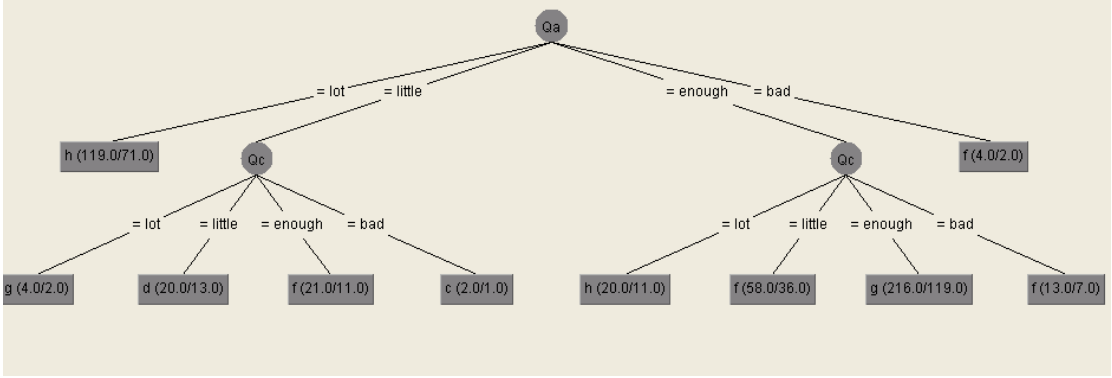
Για τα δεδομένα που αφορούν την λειτουργία στα δρομολόγια των τραίνων



Για τα δεδομένα που αφορούν την εξυπηρέτηση από το προσωπικό



Για τα δεδομένα που αφορούν την συνολική ικανοποίηση του πολίτη



4.11 Προσομοίωση με το Fuzzy Logic Toolbox

Για να υλοποιήσουμε το σύστημά μας χρησιμοποιήσαμε το Fuzzy Logic Toolbox το οποίο μας παρέχει η Matlab και είναι πολύ χρήσιμο για παρόμοιες εφαρμογές καθώς μας δίνει την δυνατότητα να χρησιμοποιήσουμε if-then κανόνες για να περιγράψουμε την συμπεριφορά ενός συστήματος. Το Fuzzy Logic Toolbox είναι ένα σύνολο συσχετισμένων συναρτήσεων υλοποιημένες σε Matlab και μας δίνει την δυνατότητα δημιουργίας και έκδοσης ασαφών συστημάτων(Fuzzy Inference Systems) χρησιμοποιώντας σαν διεπαφή το περιβάλλον της Matlab.Αν και είναι δυνατόν να χρησιμοποιήσουμε το toolbox από την γραμμή εντολών της matlab, είναι πολύ πιο εύκολο και οικονομικό να υλοποιήσουμε το σύστημα γραφικά. Το toolbox μας παρέχει πέντε εργαλεία για την δημιουργία, την έκδοση και την παρακολούθηση του ασαφούς συστήματος. Αυτά είναι τα παρακάτω:

- **Fuzzy Inference System (FIS) Editor:** χειρίζεται τα πιο υψηλού επιπέδου θέματα που αφορούν το σύστημα όπως πόσες μεταβλητές εισόδου και εξόδους θα χρησιμοποιήσουμε καθώς και το όνομά τους. Το εργαλείο δεν έχει όριο για τον αριθμό των εισόδων, ωστόσο αυτό είναι υποκειμενικό καθώς εξαρτάται και από την διαθέσιμη μνήμη του υπολογιστή μας. Αν ο αριθμός των εισόδων ή των συναρτήσεων συμμετοχής είναι αρκετά μεγάλος τότε θα είναι δύσκολο να αναλύσουμε το σύστημα .

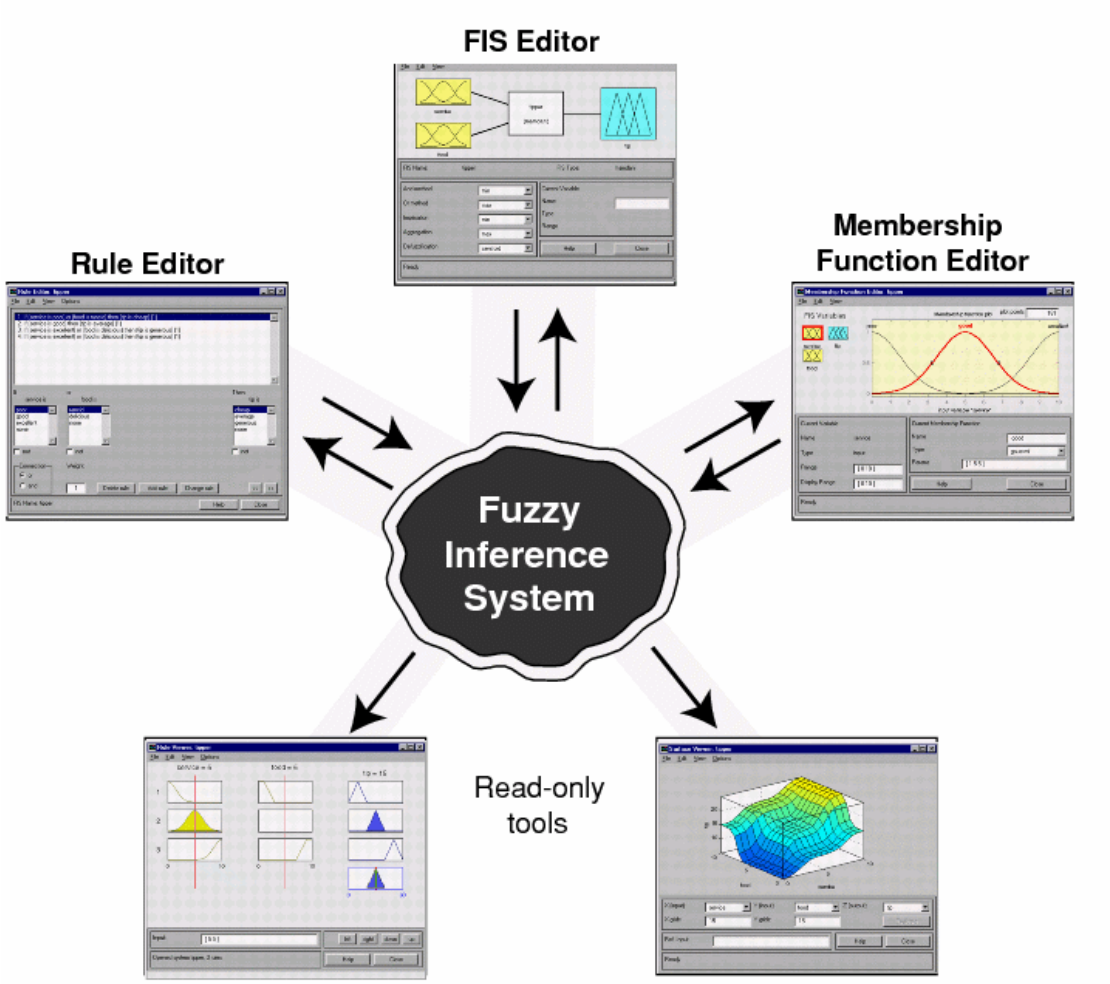
- **Membership Function Editor:** χρησιμοποιείται για να καθορίσουμε τα σχήματα των συναρτήσεων συμμετοχής κάθε εισόδου και εξόδου.

- **Rule Editor:** χρησιμοποιείται για να τοποθετήσουμε τους κανόνες που καθορίζουν την συμπεριφορά του συστήματος.

- **Rule Viewer:** χρησιμοποιείται για την επίβλεψη και παρακολούθηση του συστήματος χωρίς να μπορούμε να παρεμβάλουμε σε αυτό. Ουσιαστικά μας δείχνει το τελικό στάδιο της διαδικασίας (αποτέλεσμα) και διαγνωστικά μπορούμε να δούμε ποιοι κανόνες είναι ενεργοί ή κατά πόσο η κάθε συνάρτηση συμμετοχής επηρεάζει ή όχι το τελικό αποτέλεσμα.

- **Surface Viewer:** όπως και το προηγούμενο χρησιμοποιείται μόνο για να βλέπουμε την συμπεριφορά του συστήματος χωρίς να μπορούμε να παρεμβάλουμε σε αυτό. Χρησιμοποιείται για την επίδειξη της εξάρτησης μιας εξόδου από μία ή περισσότερες εισόδους και για τον λόγο αυτό δημιουργεί και σχεδιάζει μια έξοδο του συστήματος επιφανειακά

Στο παρακάτω σχήμα φαίνονται τα πέντε αυτά εργαλεία και ο τρόπος διασύνδεσης μεταξύ τους



Εικόνα 5.5 Τα εργαλεία του Fuzzy logic Toolbox

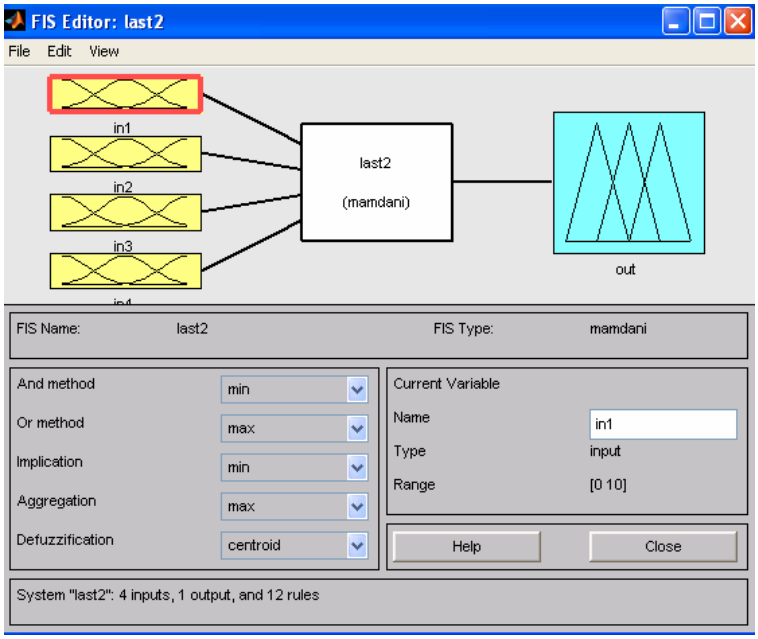
Τα πέντε αυτά εργαλεία αλληλεπιδρούν μεταξύ τους και μπορούν να ανταλλάξουν πληροφορία σε κάθε περίπτωση. Για ένα σύστημα μπορεί να είναι

ανοιχτό από ένα μέχρι και όλα τα εργαλεία.. Όταν είναι ανοιχτά παραπάνω από ένα και γίνει μία αλλαγή σε ένα από αυτά τότε το περιβάλλον διεπαφής αντιλαμβάνεται την παρουσία και των άλλων και τα ενημερώνει άμεσα. Για παράδειγμα αν έχουμε ανοιχτό το Membership Function Editor και το Rule Editor, τότε αν αλλάξουμε τα ονόματα στις συναρτήσεις συμμετοχής τότε αυτόματα θα γίνουν και οι κατάλληλες αλλαγές στο Rule Editor.Τέλος θα πρέπει να τονίσουμε ότι τα τρία πρώτα εργαλεία μπορούμε και να τα διαβάσουμε αλλά και να επέμβουμε στην τροποποίηση των δεδομένων του ασαφούς συστήματος αλλά τα τελευταία δύο είναι αυστηρά μόνο για παρακολούθηση.

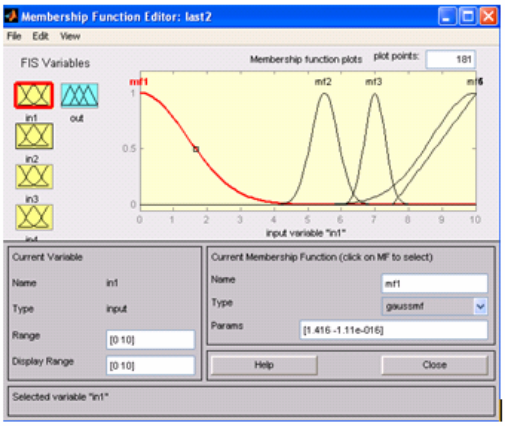
4.12 Παρουσίαση αποτελεσμάτων

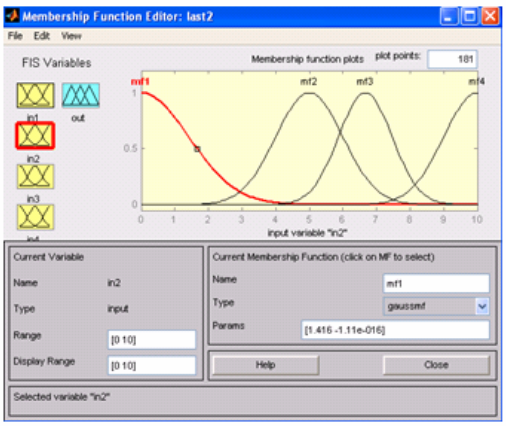
Στην συνέχεια θα παρουσιάσουμε την διαδικασία ταξινόμησης για τα διαφορετικά σετ δεδομένων τα οποία διαθέτουμε καθώς και τα αποτελέσματα που προέκυψαν. Θα αρχίσουμε πρώτα από το σετ δεδομένων που μας δίνει πληροφορίες για την λειτουργία στο μετρό και παράλληλα θα γίνει πλήρη ανάλυση της υλοποίησης του συστήματος απο το Fuzzy Logic Toolbox.

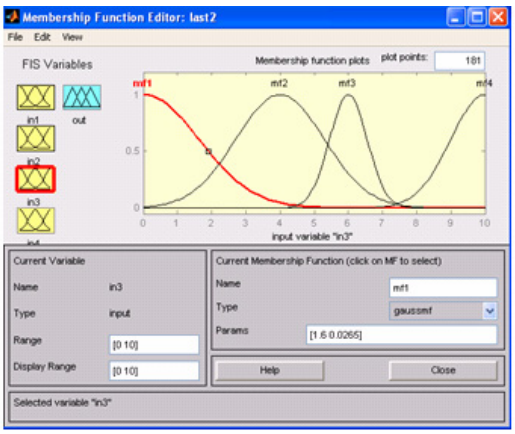
Τα χαρακτηριστικά(features) που προσδιορίζουν την κλάση των στιγμιοτύπων είναι τέσσερα, και οι κλάσεις κατηγοριοποίησης είναι επίσης τέσσερις. Έτσι δημιουργούμε 4 γραφικές παραστάσεις για τα χαρακτηριστικά και 1 για την έξοδο με τα αντίστοιχα ονόματά τους με την βοήθεια του **Fuzzy Inference System (FIS) Editor** .

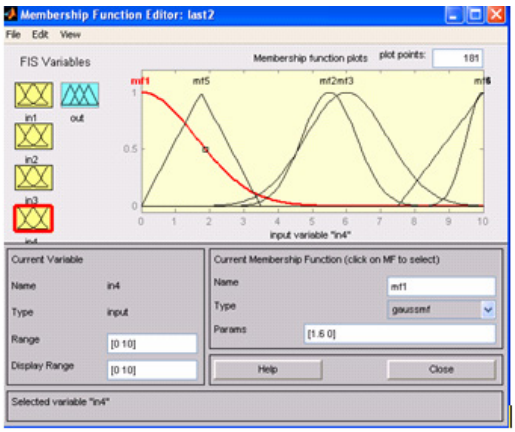


Κάθε γραφική παράσταση θα συμπεριλαμβάνει διάφορες συναρτήσεις συμμετοχής ανάλογα με τις ιδιότητες των χαρακτηριστικών και οι επικρατέστερες είναι οι Γκαούσιαν για τα χαρακτηριστικά , και οι τραπεζοειδείς και τριγωνομετρικές για τις κλάσεις. Ο αριθμός των συναρτήσεων συμμετοχής των χαρακτηριστικών και οι κανόνες ουσιαστικά εξαρτώνται το ένα από το άλλο. Για το λόγο αυτό εκμεταλεύοντας τους κανόνες που είδαμε προηγουμένως, μπορούμε να δούμε ποιες είναι οι ‘κρίσιμες‘ τιμές των χαρακτηριστικών, δηλαδή οι διαχωριστικές τιμές που καθορίζουν τους κανόνες και κατά επέκταση τις κλάσεις τους. Συνήθως οι τιμές αυτές καθορίζουν τα άκρα των συναρτήσεων συμμετοχής, κι έτσι μπορούμε να προσδιορίσουμε όσο πιο καλά μπορούμε σε ποιες τιμές ανήκει κάθε συνάρτηση. Τις συναρτήσεις τις σχεδιάζουμε με το **Membership Function Editor**

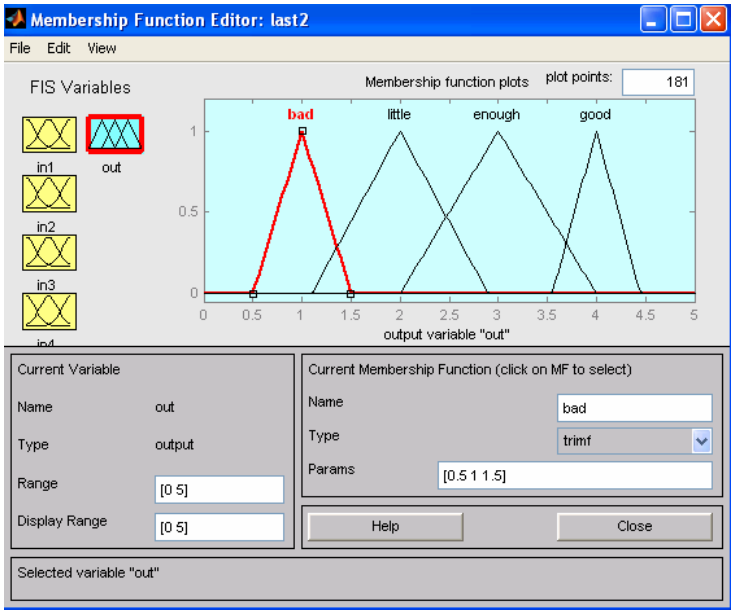








Η γραφική παράσταση της εξόδου από την οποία προκύπτει σε ποια κλάση θα ανήκουν τα χαρακτηριστικά, αποτελείται από τέσσερις συναρτήσεις συμμετοχής όσες είναι και οι κλάσεις. Το κέντρο των συναρτήσεων αυτών βρίσκεται περίπου στον αριθμό τον οποίο έχουμε αντιστοιχίσει για κάθε κλάση.



Στον παρακάτω πίνακα φαίνεται η αντιστοίχηση του ονόματος της συνάρτησης συμμετοχής με την τιμή της κλάσης.

Class	Number
Bad	1
Little	2
Enough	3
Good	4

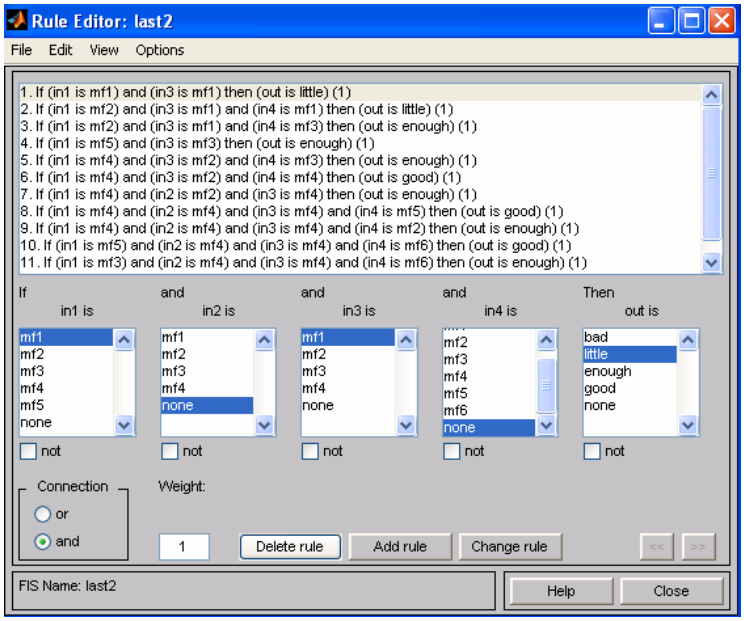
Επόμενο βήμα μας είναι η δημιουργία των κανόνων με το **Rule Editor.** Το εργαλείο αυτό μας δίνει την δυνατότητα με πολύ απλό τρόπο να εφαρμόσουμε τους κανόνες οι οποίοι χαρακτηρίζουν το σύστημα. Αυτό γίνεται συνδυάζοντας τα ονόματα των συναρτήσεων των εισόδων μεταξύ τους μέσω των τελεστών **AND**, **OR** , **NOT**, ο συνδυασμός των οποίων παράγει μία έξοδο που θα αντιστοιχεί στην προβλεπόμενη κλάση. Το **Rule Editor** μας δίνει τις εξής δυνατότητες:

- Δημιουργία κανόνα επιλέγοντας ένα στοιχείο για κάθε είσοδο και έξοδο και τον αντίστοιχο τελεστή. Αν ένα χαρακτηριστικό δεν λαμβάνει μέρος στον κανόνα τότε μπορούμε να χρησιμοποιήσουμε την επιλογή none.

- Διαγραφή ενός κανόνα με την επιλογή Delete Rule.

- Δυνατότητα διόρθωσης και αυτόματης αλλαγής του κανόνα με την επιλογή Change Rule.

- Δυνατότητα προσθήκης βάρους σε κάθε κανόνα από 0 έως 1 ανάλογα με την σημαντικότητά του. Αν δεν αντιστοιχήσουμε έναν κανόνα με το βάρος του τότε παίρνει default τιμή ίση με 1.



Όπως αναφέρθηκε προηγουμένως τους περισσότερους κανόνες τους λάβαμε από τον αλγόριθμο δέντρου απόφασης J48. θα περιγράψουμε την δημιουργία των δύο πρώτων κανόνων για το συγκεκριμένο σετ για να κάνουμε πιο σαφή τον τρόπο με τον οποίο τους υλοποιήσαμε στο **Rule Editor.** Ξεκινώντας από την αριστερή πλευρά του δέντρου έχουμε για το πρώτο χαρακτηριστικό Q1<=4, για το τρίτο Q3<=4 και στο κλαδί του την κλάση little με πιθανότητα little(17/1). Έτσι λοιπόν έχουμε αντιστοιχίσει αυτές τις 2 τιμές χαρακτηριστικών όπως είδαμε και στα σχήματα των **Membership Function Editor** τις συναρτήσεις συμμετοχής mf1 στις εισόδους in1 και in3 και σαν έξοδο την little. Οπότε ο κανόνας που προκύπτει θα είναι ο
if (in1 is mf1)and(in3 is mf1) then (out is little).

Παρόμοια για τον δεύτερο κανόνα έχουμε ότι Q1<=4, Q3<=4, Q4<=4 που αντιστοιχούν στην κλάση little επίσης. Το Q4<=4 αντιστοιχεί στην συνάρτηση συμμετοχής mf1 για το τέταρτο χαρακτηριστικό οπότε εδώ ο κανόνας μας θα είναι ο ***if (in1 is mf1)and(in3 is mf1)and(in4 is mf1) then (out is little)***.

Για το συγκεκριμένο σετ οι κανόνες θα είναι 12. Οι υπόλοιποι 10 είναι:

If (in1 is mf2)and(in3 is mf1)and(in4 is mf3) then (out is enough)
if (in1 is mf5)and(in3 is mf3) then (out is enough)
if (in1 is mf4)and(in3 is mf2)and(in4 is mf3) then (out is enough)
if (in1 is mf4)and(in3 is mf2)and(in4 is mf4) then (out is good)
if (in1 is mf4)and(in2 is mf2)and(in3 is mf4) then (out is enough)
if (in1 is mf4)and(in2 is mf4)and(in3 is mf4)and(in4 is mf5) then (out is good)
if (in1 is mf4)and(in2 is mf4)and(in3 is mf4)and(in4 is mf3) then (out is enough)
if (in1 is mf5)and(in2 is mf4)and(in3 is mf4)and(in4 is mf6) then (out is good)
if (in1 is mf3)and(in2 is mf4)and(in3 is mf4)and(in4 is mf6) then (out is enough)
if (in1 is m4)and(in2 is mf4)and(in3 is mf4) and(in4 is mf6) then (out is good)

Αξίζει να σημειωθεί ότι σχεδόν σε όλους τους κανόνες έχουμε προσαρτήσει βάρος ίσο με 1 αφού λόγω του αριθμού των κανόνων και των συναρτήσεων συμμετοχής ήταν δύσκολο να υπολογίσουμε ικανοποιητικά βάρη. Αργότερα θα δούμε ότι με χρήση γενετικών αλγορίθμων θα μπορέσουμε να τα υπολογίσουμε.

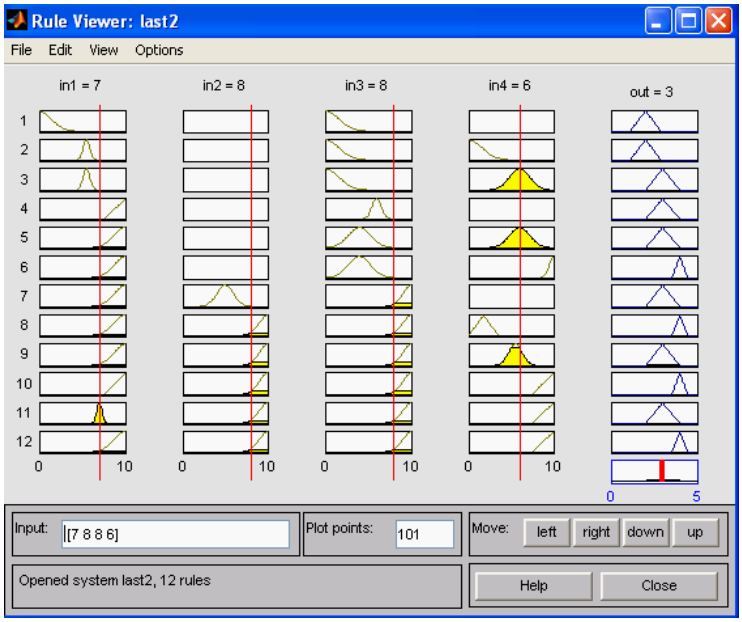
Στην συνέχεια μέσω του **Rule Viewer** μπορούμε να εξακριβώσουμε κατά πόσο οι κανόνες που δημιουργήσαμε είναι ικανοί ώστε να μας προσφέρουν μια ταξινόμηση με ικανοποιητικά αποτελέσματα τοποθετώντας τιμές χαρακτηριστικών στην είσοδο και σαν έξοδο να παίρνουμε την αναμενόμενη κλάση. Ουσιαστικά τώρα έχουμε την δυνατότητα να αναλύσουμε και να δούμε όλη την πορεία του ασαφούς συστήματος που περιγράψαμε προηγουμένως. Έτσι μπορούμε να διακρίνουμε τα εξής στοιχεία :

- Αριστερά από κάθε γραμμή βρίσκεται ο αριθμός του κανόνα που φτιάξαμε στο **Ryle Editor.**

- Οι τρεις πρώτες στήλες κάθε γραμμής με τον κίτρινο σχεδιασμό αναπαριστούν τις συναρτήσεις συμμετοχής που αντιστοιχούν στο if-μέρος (είσοδος) του κανόνα.

- Η τελευταία στήλη των γραμμών με τον μπλε σχεδιασμό αναπαριστά την συνάρτηση συμμετοχής του then-μέρους(έξοδος) του κανόνα.

- Τέλος η τελευταία γραμμή περιέχει μόνο μία γραφική παράσταση η οποία αντιστοιχεί στην συνάθροιση εξόδων όλων των προηγούμενων κανόνων και με την κόκκινη κάθετη γραμμή παρουσιάζεται το αποτέλεσμα της αποασαφοποίησης.

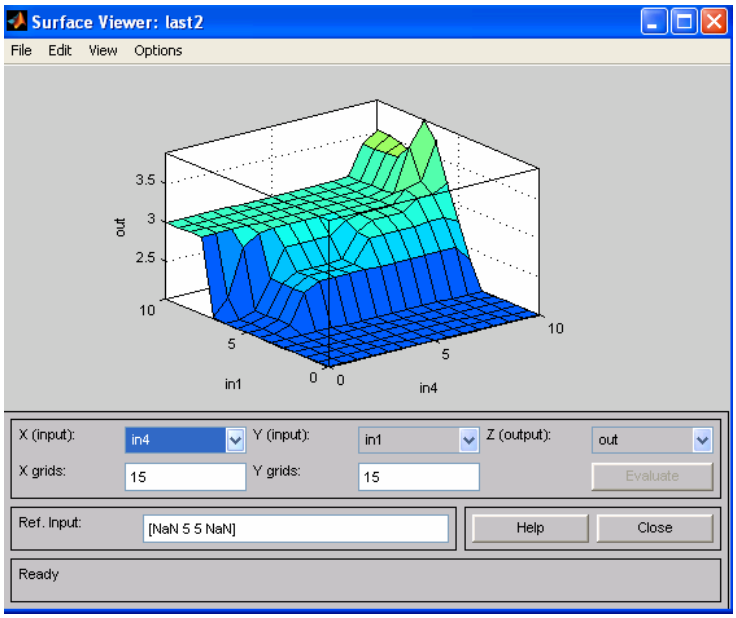


Το αποτέλεσμα της αποασαφοποίησης μας δείχνει την κλάση στην οποία ανήκουν τα συγκεκριμένα χαρακτηριστικά. Η εκάστοτε κλάση ισοδυναμεί με τον πλησιέστερο αριθμό (*1,2,3,4*), που αντιστοιχεί σε αυτές, ως προς το αποτέλεσμα της αποασαφοποίησης. Για παράδειγμα εάν το αποτέλεσμα θα είναι *2,6* τότε τα χαρακτηριστικά θα ανήκουν στην κλάση *3*. Το ποσοστό σωστής ταξινόμησης υπολογίζεται με την υλοποίηση συνάρτησης σε κώδικα Matlab. Αρχικά φορτώνουμε τα δεδομένα ελέγχου και το σύστημα που μόλις υλοποιήσαμε. Στην συνέχεια επιλέγουμε μόνο τα χαρακτηριστικά από κάθε στιγμιότυπο και με loop τα περνάμε στο Rule Viewer, όπου υπολογίζουμε τις κλάσεις. Από εκεί και μετά για κάθε

στιγμιότυπο του σετ ελέγχου συγκρίνουμε την προκύπτουσα με την πραγματική κλάση για όλα τα στιγμιότυπα και προκύπτει το τελικό ποσοστό επιτυχίας.

Στην περίπτωση αυτή το ποσοστό επιτυχούς ταξινόμησης υπολογίστηκε 74.89%. Βλέπουμε ότι υπάρχει μια σημαντική αύξηση 2.89% σε σχέση με τον J48 ενώ έχει ξεπεράσει και το καλύτερο ποσοστό που είχαμε υπολογίσει στο WEKA κατά 0.38%, μια αύξηση η οποία δεν είναι αρκετά μεγάλη αλλά μας δείχνει ότι ο στόχος μας επιτεύχθηκε και ξεπεράσαμε τα ποσοστά που μας έδωσαν όλοι οι αλγόριθμοι στο εργαλείο WEKA.

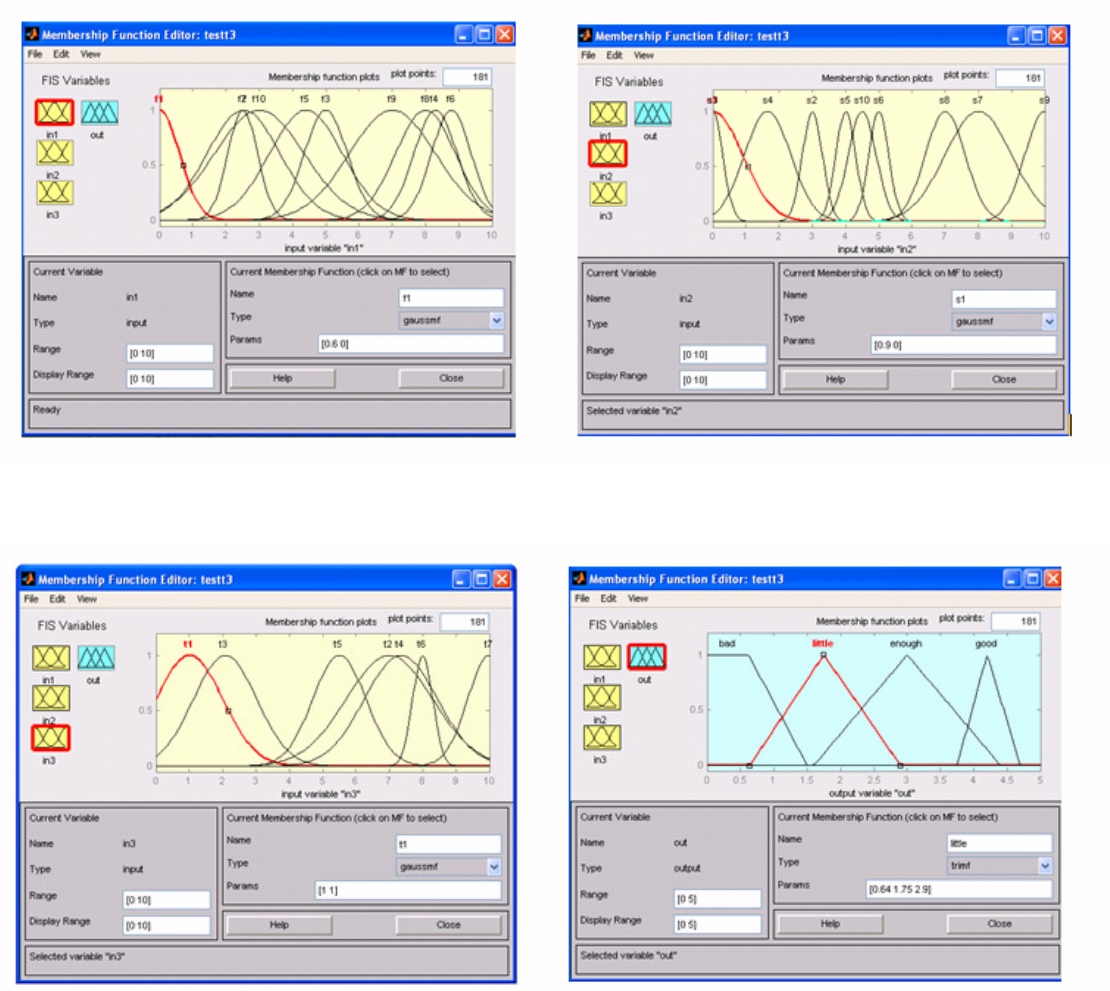
Τέλος με το **Surface Viewer** βλέπουμε με τρισδιάστατη απεικόνιση την εξάρτηση των εξόδων από τις εισόδους. Στον άξονα X και Y αναπαριστούμε δύο εκ των τεσσάρων εισόδων του συστήματος και στον Z απεικονίζεται η έξοδος για κάθε τιμή τους .Έτσι στο παρακάτω σχήμα διακρίνουμε πως όσο μεγαλώνει η τιμή των δύο χαρακτηριστικών τόσο μεγαλώνει και η τιμή του βαθμού της τάξης κάτι που ήταν αναμενόμενο.



Στην συνέχεια θα παρουσιάσουμε για τα υπόλοιπα σετ δεδομένων τις γραφικές παραστάσεις των συναρτήσεων συμμετοχής των εισόδων και εξόδων.

Ασαφές σύστημα για τα δεδομένα που αφορούν τις συνθήκες στο μετρό

Εδώ έχουμε τρία χαρακτηριστικά, οπότε και τρεις γραφικές παραστάσεις εισόδου και μία για την έξοδο. Οι γραφικές παραστάσεις των συναρτήσεων συμμετοχής εισόδων και εξόδων παρουσιάζονται παρακάτω:



Οι κανόνες μας είναι 20 και είναι οι εξής:

if (in1 is f1)and(in2 is s4) then (out is bad)

if (in1 is f2)and(in2 is s1)and(in3 is t1) then (out is bad)

if (in1 is f2)and(in2 is s2)and(in3 is t1) then (out is little)

if (in1 is f9)and(in2 is s3)and(in3 is t1) then (out is enough)

if (in1 is f9)and(in2 is s1)and(in3 is t1) then (out is little)

if (in1 is f3)and(in2 is s2)and(in3 is t1) then (out is enough)

if (in1 is f4)and(in2 is s2)and(in3 is t1) then (out is bad)

if (in1 is f5)and(in2 is s4)and(in3 is t2) then (out is little)

if (in1 is f6)and(in2 is s1)and(in3 is t2) then (out is enough)

if (in1 is f6)and(in2 is s2)and(in3 is t2) then (out is little)

if (in2 is s10)and(in3 is t1) then (out is little)

if (in1 is f7)and(in2 is s5)and(in3 is t4) then (out is little)

if (in1 is f8)and(in2 is s5)and(in3 is t4) then (out is enough)

if (in2 is s6)and(in3 is t4) then (out is enough)

if (in1 is f10)and(in2 is s7)and(in3 is t1) then (out is little)

if (in1 is f4)and(in2 is s7)and(in3 is t1) then (out is enough)

if (in2 is s7)and(in3 is t5) then (out is enough)

if (in2 is s7)and(in3 is t6) then (out is enough)

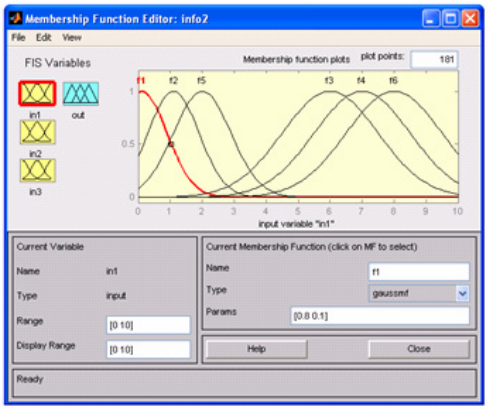
if (in2 is s8)and(in3 is t7) then (out is enough)

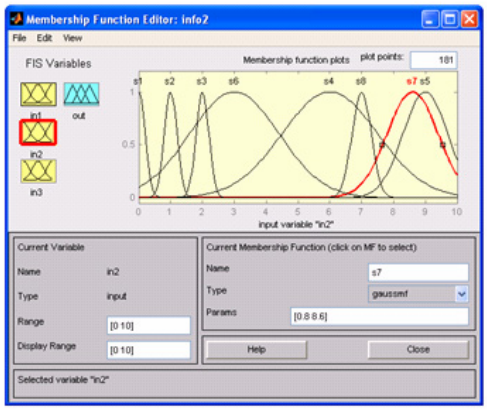
if (in2 is s9)and(in3 is t7) then (out is good)

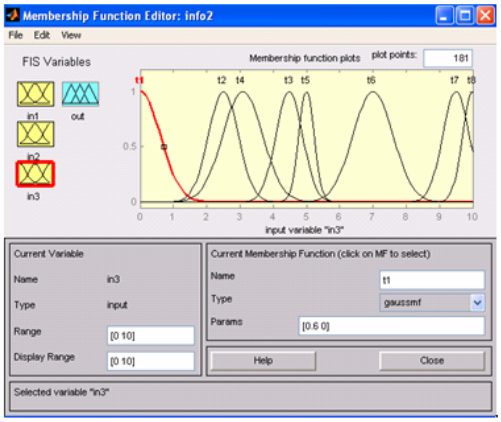
Στην περίπτωση αυτή το ποσοστό επιτυχούς ταξινόμησης υπολογίστηκε 75.59%. Βλέπουμε ότι υπάρχει μια σημαντική αύξηξη 4.49% σε σχέση με τον J48 ενώ έχει ξεπεράσει και το καλύτερο ποσοστό που είχαμε υπολογίσει στο WEKA κατά 2.308%. Εδώ παρατηρούμε ότι τα ασαφές σύστημα λειτούργησε με μεγαλύτερη επιτυχία από το προηγούμενο και η βελτίωση που έγινε ήταν πολύ σημαντική σε σχέση με τους αλγόριθμους του WEKA.

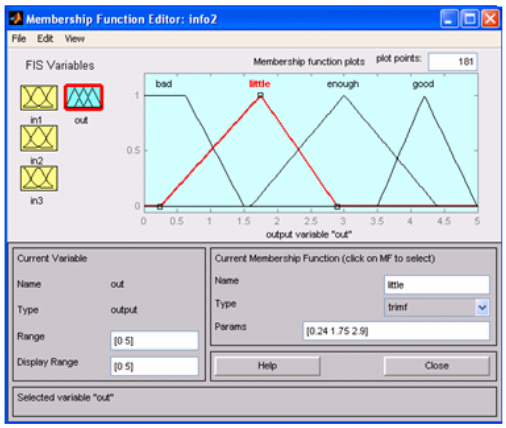
Ασαφές σύστημα για τα δεδομένα που αφορούν την ικανοποίηση από το προσωπικό

Όπως και προηγουμένως έχουμε τρία χαρακτηριστικά, οπότε τρεις γραφικές παραστάσεις εισόδου και μία για την έξοδο. Αυτές είναι οι παρακάτω:









Οι κανόνες μας εδώ είναι 15 και είναι οι παρακάτω:

if (in3 is t1) then (out is bad)

if (in1 is f2)and(in2 is s2)and(in3 is t4) then (out is bad)

if (in1 is f4)and(in2 is s1)and(in3 is t4) then (out is little)

if (in1 is f4)and(in2 is s2)and(in3 is t4) then (out is bad)

if (in1 is f1)and(in2 is s3)and(in3 is t2) then (out is bad)

if (in1 is f1)and(in2 is s4)and(in3 is t2) then (out is little)

if (in1 is f1)and(in2 is s4)and(in3 is t3) then (out is little)

if (in1 is f3)and(in2 is s4)and(in3 is t4) then (out is little)

if (in1 is f3)and(in2 is s4)and(in3 is t5) then (out is enough)

if (in3 is t6) then (out is enough)

if (in2 is s6)and(in3 is t8) then (out is enough)

if (in2 is s6)and(in3 is t8) then (out is good)

if (in2 is s8)and(in3 is t8) then (out is enough)

if (in1 is f5)and(in2 is s5)and(in3 is t7) then (out is enough)

if (in1 is f6)and(in2 is s5)and(in3 is t7) then (out is good)

Στην περίπτωση αυτή το ποσοστό επιτυχούς ταξινόμησης υπολογίστηκε 76.65%. Όπως και προηγουμένως παρατηρούμε ότι έχουμε σημαντική βελτίωση αφού έχουμε αύξηση 3.71% σε σχέση με τον J48 ενώ έχει ξεπεράσει και το καλύτερο ποσοστό που είχαμε υπολογίσει στο WEKA κατά 1.62%.

Ασαφές σύστημα για τα δεδομένα που αφορούν τον συνολικό βαθμό ικανοποίησης

Ομοίως με πριν έχουμε τρία χαρακτηριστικά, οπότε τρεις γραφικές παραστάσεις εισόδου και μία για την έξοδο. Αυτές είναι οι παρακάτω:

Οι κανόνες μας είναι 21 και είναι οι παρακάτω:

if (in1 is little)and(in3 is good) then (class is mf5)

if (in1 is little)and(in3 is little) then (class is mf5)

if (in1 is little)and(in3 is enough) then (class is mf6)

if (in1 is little)and(in3 is bad) then (class is mf3)

if (in1 is bad) then (class is mf6)

if (in1 is enough)and(in3 is little) then (class is mf6)

if (in1 is enough)and(in2 is little)and(in3 is enough) then (class is mf6)

if (in1 is enough)and(in2 is enough)and(in3 is enough) then (class is mf7)

if (in1 is good)and(in2 is good)and(in3 is enough) then (class is mf8)

if (in1 is good)and(in2 is good)and(in3 is good) then (class is mf9)

if (in1 is good)and(in2 is enough)and(in3 is good) then (class is mf8)

if (in1 is enough)and(in2 is good)and(in3 is good) then (class is mf8)

if (in1 is good)and(in2 is enough)and(in3 is enough) then (class is mf8)

if (in1 is enough)and(in2 is enough)and(in3 is good) then (class is mf8)

if (in1 is enough)and(in2 is good)and(in3 is enough) then (class is mf8)

if (in1 is enough)and(in2 is little)and(in3 is bad) then (class is mf5)

if (in1 is bad)and(in2 is little)and(in3 is enough) then (class is mf5)

Στην περίπτωση αυτή το ποσοστό επιτυχούς ταξινόμησης υπολογίστηκε 43.39%. Και στην περίπτωση αυτή έχουμε αύξηξη 3.078 % σε σχέση με τον J48 ενώ έχει ξεπεράσει και το καλύτερο ποσοστό που είχαμε υπολογίσει στο WEKA κατά 1.715%.

Η βελτίωση των ποσοστών επιτυχούς ταξινόμησης σε όλες τις κατηγορίες δεδομένων ήταν σημαντική κάτι που μας επιβεβαιώνει ότι η ασαφής λογική είναι μία πολύ χρήσιμη μέθοδος για την ταξινόμηση δεδομένων και δίκαια την επιλέξαμε. Κύρια αιτία της βελτίωσης αυτής είναι σαφώς η σωστή δημιουργία των κανόνων καθώς και η απεικόνιση των συναρτήσεων συμμετοχής που είναι άμεσα συνδεδεμένες με του κανόνες.

ΚΕΦΑΛΑΙΟ 5 : ΓΕΝΕΤΙΚΟΙ ΑΛΓΟΡΙΘΜΟΙ

5.1 Εισαγωγή και ιστορική αναδρομή

Οι γενετικοί αλγόριθμοι είναι συστήματα επίλυσης προβλημάτων που βασίζονται στις αρχές τις φυσικής εξέλιξης που έχουν αναπτυχθεί τα τελευταία τριάντα χρόνια.

Η λειτουργία τους έχει ως εξής: Διατηρούμε ένα πληθυσμό πιθανών κωδικοποιημένων λύσεων του προβλήματος που θέλουμε να επιλύσουμε και εφαρμόζουμε πάνω σε αυτόν διάφορες διαδικασίες εμπνευσμένες από την βιολογική εξέλιξη. Έτσι, περνώντας από γενιά σε γενιά, τα συστήματα αυτά δημιουργούν συνέχεια νέους πληθυσμούς πιθανών λύσεων εξελίσωντας τους προηγούμενους πληθυσμούς με κατάλληλες μεθόδους που θα επεξηγηθούν στην συνέχεια, και στο τέλος κρατάμε τον τελευταίο πληθυσμό ο οποίος και θα είναι η λύση του προβλήματός μας.

Η πρώτη εμφάνιση των γενετικών αλγορίθμων χρονολογείται στις αρχές του 1950, όταν επιστήμονες από τον χώρο της βιολογίας αποφάσισαν να χρησιμοποιήσουν υπολογιστές στην προσπάθειά τους να προσομοιώσουν πολύπλοκα βιολογικά συστήματα. Όμως η ανάπτυξή τους που τους οδήγησε στην μορφή η οποία είναι ευρέως διαδεδομένη έως και στις μέρες μας πραγματοποιήθηκε στις αρχές του 1970 από τον john Holland και τους συνεργάτες του στο πανεπιστήμιο του Michigan.

Η βασική ιδέα των γενετικών αλγορίθμων (Γ.Α.) είναι η μίμηση των μηχανισμών της βιολογικής εξέλιξης που απαντώνται στην φύση. Σαν επακόλουθο χρησιμοποιείται ορολογία η οποία είναι δανεισμένη από τον χώρο της Φυσικής Γενετικής. Έτσι αναφερόμαστε σε άτομα ή γενότυπους μέσα σε ένα πληθυσμό. Κάθε άτομο ή γενότυπος αποτελείται από χρωμοσώματα. Στους Γ.Α αναφερόμαστε συνήθως σε άτομα με ένα χρωμόσωμα. Τα χρωμοσώματα αποτελούνται από γονίδια που είναι διατεταγμένα σε γραμμική ακολουθία. Κάθε γονίδιο επηρεάζει την

κληρονομικότητα ενός η περισσότερων χαρακτηριστικών. Τα γονίδια τα οποία επηρεάζουν συγκεκριμένα χαρακτηριστικά γνωρίσματα του ατόμου ονομάζονται loci και κάθε χαρακτηριστικό γνώρισμα του ατόμου έχει την δυνατότητα να εμφανιστεί με διάφορες μορφές ανάλογα με την κατάσταση στην οποία βρίσκεται το αντίστοιχο γονίδιο που την επηρεάζει. Οι διαφορετικές αυτές καταστάσεις που μπορεί να πάρει το γονίδιο ονομάζονται alleles.

Κάθε γενότυπος αναπαριστά μια πιθανή λύση στο πρόβλημά μας. Το αποκωδικοποιημένο περιεχόμενο ενός συγκεκριμένου χρωμοσώματος καλείται φαινότυπος. Μια διαδικασία εξέλιξης που εφαρμόζεται πάνω σε ένα πληθυσμό αντιστοιχεί σε ένα εκτενές ψάξιμο στον χώρο των πιθανών λύσεων, και με την εξερεύνηση όλου του διαστήματος και την διατήρηση των πιθανών καλύτερων λύσεων θα έχουμε τα επιθυμητά αποτελέσματα.

Οι Γ.Α. διατηρούν ένα πληθυσμό πιθανών λύσεων, του προβλήματος που θέλουμε να επιλύσουμε, πάνω στον οποίο δουλεύουν, αντίθετα από άλλες μεθόδους αναζήτησης που επικεντρώνονται σε ένα μόνο σημείο του διαστήματος αναζήτησης. Έτσι πραγματοποιείται ταυτόχρονη αναζήτηση σε πολλές κατευθύνσεις και υποστηρίζει καταγραφή και ανταλλαγή πληροφοριών μεταξύ αυτών των κατευθύνσεων. Ο πληθυσμός υφίσταται μια προσωμειωμένη γενετική εξέλιξη, οπότε σε κάθε γενιά οι σχετικά ¨καλές¨ λύσεις αναπαράγονται ενώ οι σχετικά ¨κακές¨ λύσεις απομακρύνονται ώστε στο τέλος να βρούμε την βέλτιστη λύση. Ο διαχωρισμός και η αποτίμηση των διαφόρων λύσεων γίνεται με την βοήθεια μιας αντικειμενικής συνάρτησης η οποία παίζει τον ρόλο του περιβάλλοντος μέσα στο οποίο εξελίσσεται ο πληθυσμός

5.2 Δομή και λειτουργία ενός γενετικού αλγόριθμου

Κατά την διάρκεια της γενιάς t ο Γ.Α. διατηρεί ένα πληθυσμό από n πιθανές λύσεις. Κάθε άτομο αποτιμάται από αυτές και δίνει ένα μέτρο της καταλληλότητας και ορθότητάς του. Αφού ολοκληρωθεί η αποτίμηση όλων των μελών του πληθυσμού, δημιουργείται ένας νέος πληθυσμός (γενιά t+1) που προκύπτει από την

επιλογή των πιο κατάλληλων στοιχείων του πληθυσμού της προηγούμενης γενιάς. Μερικά μέλη από τον καινούργιο αυτό πληθυσμό υφίστανται αλλαγές με την βοήθεια των γενετικών διαδικασιών της διασταύρωσης και της μετάλλαξης σχηματίζοντας νέες πιθανές λύσεις. Η διασταύρωση συνδυάζει τα στοιχεία των χρωμοσωμάτων των δύο γονέων για να δημιουργήσει δύο νέους απογόνους ανταλλάσσοντας κομμάτια από γονείς. Για παράδειγμα αν οι δυο γονείς αναπαρίστανται με χρωμοσώματα 5 γονιδίων (a1,b1,c1,d1,e1) και (a2,b2,c2,d2,e2) τότε οι απόγονοι οι οποίοι θα προκύψουν με σημείο διασταύρωσης το σημείο 2 θα είναι οι (a1,b1,c2,d2,e2) (a2,b2,c1,d1,e1). Η διαδικασία της μετάλλαξης αλλάζει αυθαίρετα ένα ή και περισσότερα γονίδια ενός συγκεκριμένου χρωμοσώματος. Πραγματοποιείται με τυχαία αλλαγή γονιδίων με πυκνότητα ίση με τον ρυθμό μετάλλαξης.

Συνοψίζοντας ,ένας Γ.Α. για ένα πρόβλημα πρέπει να αποτελείται από τα παρακάτω συστατικά:

1) *μια γενετική αναπαράσταση των πιθανών λύσεων του προβλήματος*

2) *ένα τρόπο δημιουργίας ενός αρχικού πληθυσμού από πιθανές λύσεις*

3) *μια αντικειμενική συνάρτηση αξιολόγησης των μελών του πληθυσμού ,που παίζει τον ρόλο του περιβάλλοντος*

4) *γενετικούς τελεστές για την δημιουργία νέων μελών*

5) *τιμές για τις διάφορες παραμέτρους που χρησιμοποιεί ο Γ.Α.*

5.3 Βασικός Γενετικός Αλγόριθμος

Ο ΓΑ θεωρεί κάθε υποψήφια λύση κάποιου προβλήματος ως μία οντότητα που ονομάζεται χρωμόσωμα και η οποία μπορεί να κωδικοποιηθεί με ένα σύνολο παραμέτρων. Οι παράμετροι θεωρούνται ότι είναι τα γονίδια ενός χρωμοσώματος και μπορούν να δομηθούν ως ακολουθίες δυαδικών ψηφίων ή πραγματικών αριθμών. Κάθε υποψήφια λύση αποτιμάται από μία συνάρτηση, την συνάρτηση προσαρμογής,

και δίνει μία τιμή, την τιμή προσαρμογής π_i η οποία αντανακλά το πόσο «καλό» είναι το χρωμόσωμα για το πρόβλημα.

Αρχικά επιλέγεται ένα σύνολο από n χρωμοσώματα v_i, $i=1,2,\ldots,n$ που αποτελούν τον αρχικό πληθυσμό χρωμοσωμάτων και που το μέγεθος του κυμαίνεται από πρόβλημα σε πρόβλημα. Η επιλογή του αρχικού πληθυσμού είναι τις περισσότερες φορές τυχαία εκτός αν υπάρχει κάποια γνώση για τα δεδομένα όποτε ο αρχικός πληθυσμός προσαρμόζεται σε αυτή την γνώση. Μετά από αυτό αρχίζει ένας κύκλος εξελικτικής λειτουργίας που επαναλαμβάνεται τόσες φορές όσες θα οριστούν από το πρόγραμμα. Τα βασικά βήματα ενός γενετικού αλγορίθμου είναι τα παρακάτω:

1. Αρχικοποίηση. Δημιουργία με τυχαίο τρόπο ενός αρχικού πληθυσμού δυνατών λύσεων

2. Αξιολόγηση της κάθε λύσης μέσω της αντικειμενική μας συνάρτησης

3. Επιλογή ενός νέου πληθυσμού με βάση την απόδοση κάθε μέλους του προηγούμενου πληθυσμού

4. Εφαρμογή στον πληθυσμό που προκύπτει μετά τη διαδικασία της επιλογής των γενετικών τελεστών της διασταύρωσης(crossover) και της μετάλλαξης (mutation)

5. Με την ολοκλήρωση του προηγούμενου βήματος θα έχει δημιουργηθεί η επόμενη γενιά οπότε και επιστρέφουμε ξανά πίσω στο βήμα 2

6. Μετά από κάποιο αριθμό γενεών και αφού ο αλγόριθμος συγκλίνει και δεν επιδέχεται περαιτέρω βελτίωση ,τότε ο γενετικός αλγόριθμος τερματίζεται

5.3.1 Αρχικοποίηση

Κατά την διαδικασία της αρχικοποίησης δημιουργούμε έναν αρχικό πληθυσμό από δυνατές λύσεις. Αυτό γίνεται παράγοντας size τυχαίους αριθμούς όπου size είναι

το μέγεθος του πληθυσμού το οποίο και θα επεξεργαστεί ο γενετικός αλγόριθμος και οι αριθμοί ουσιαστικά αποτελούν τα χρωμοσώματα που περιέχουν τις λύσεις. Τέλος το μέγεθος του πληθυσμού παραμένει σταθερό σε όλη την διαδικασία του Γ.Α.

5.3.2 Επιλογή

Η λειτουργία της επιλογής επιλέγει τους γονείς για την επόμενη γενιά βασισμένη στην απόδοση κάθε μέλους του πληθυσμού στην συνάρτηση στόχου. Όσο καλύτερο είναι κάθε μέλος του πληθυσμού τόσο μεγαλύτερη είναι η πιθανότητα να επιλεγεί και να περάσει στην επόμενη γενιά. Με αυτή την λογική τα μέλη με την μεγαλύτερη τιμή προσαρμογής αναμένεται να επιλεχθούν πάνω από μία φορές, αυτά με ενδιάμεση τιμή μένουν στα ίδια επίπεδα και αυτά με την χειρότερη τιμή δεν επιλέγονται σε επόμενο κύκλο (πεθαίνουν).

5.3.3 Διασταύρωση και Μετάλλαξη

Το επόμενο βήμα του γενετικού αλγορίθμου είναι η διασταύρωση και η μετάλλαξη. Κατά την διαδικασία αυτή ο Γ.Α. χρησιμοποιεί τα άτομα της τρέχουσας γενιάς για να δημιουργήσει τα παιδιά-άτομα τα οποία θα ενεργήσουν στην επόμενη γενιά. Έτσι λοιπόν εκτός από τα άτομα με την μεγαλύτερη τιμή προσαρμογής (elite children) σε μία τρέχουσα γενιά ο αλγόριθμος δημιουργεί:

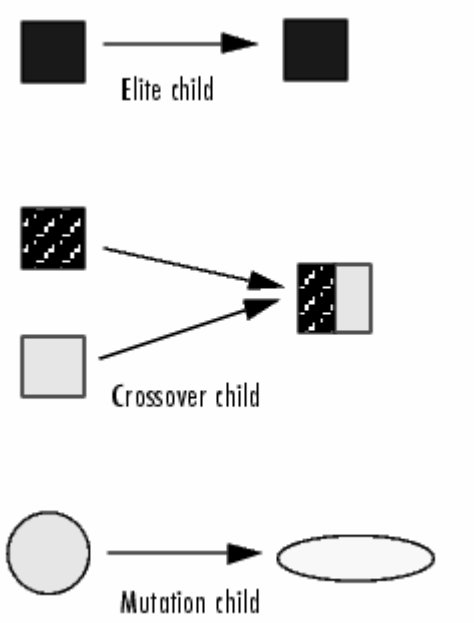
- Διασταυρωμένα παιδιά-άτομα (crossover children) επιλέγοντας συγκεκριμένα γονίδια ή και δανύσματα γονιδίων από ζευγάρια τρέχων ατόμων που συνδυάζονται μεταξύ τους για την δημιουργία των ατόμων αυτών.

- Παιδιά μετάλλαξης (mutation children) με την εφαρμογή τυχαίων αλλαγών σε ένα μεμονωμένο άτομο της τρέχουσας γενιάς.

Και οι δύο αυτές οι διαδικασίες είναι πολύ σημαντικές και ουσιαστικές για ένα γενετικό αλγόριθμο. Η διασταύρωση επιτρέπει στον Γ.Α. να εξαγάγει τα καλύτερα γονίδια από διαφορετικά άτομα και να τα επανασυνσδυάσει στα ενδεχομένως ανώτερα παιδιά. Η μετάλλαξη προσθέτει στην ποικιλομορφία ενός

πληθυσμού και με αυτόν τον τρόπο αυξάνει την πιθανότητα ότι ο αλγόριθμος θα παράγει τα άτομα με τις μεγαλύτερες τιμές προσαρμογής. Χωρίς μετάλλαξη, ο αλγόριθμος θα μπορούσε να παράγει μόνο τα άτομα των οποίων τα γονίδια ήταν ένα υποσύνολο των συνδυασμένων γονιδίων στον αρχικό πληθυσμό. Μετά τις διαδικασίες της διασταύρωσης και της μετάλλαξης τα άτομα αποτιμούνται πάλι.

Παρακάτω παρουσιάζεται ένα διάγραμμα για το πώς δημιουργούνται αυτοί οι τύποι των παιδιών-ατόμων:



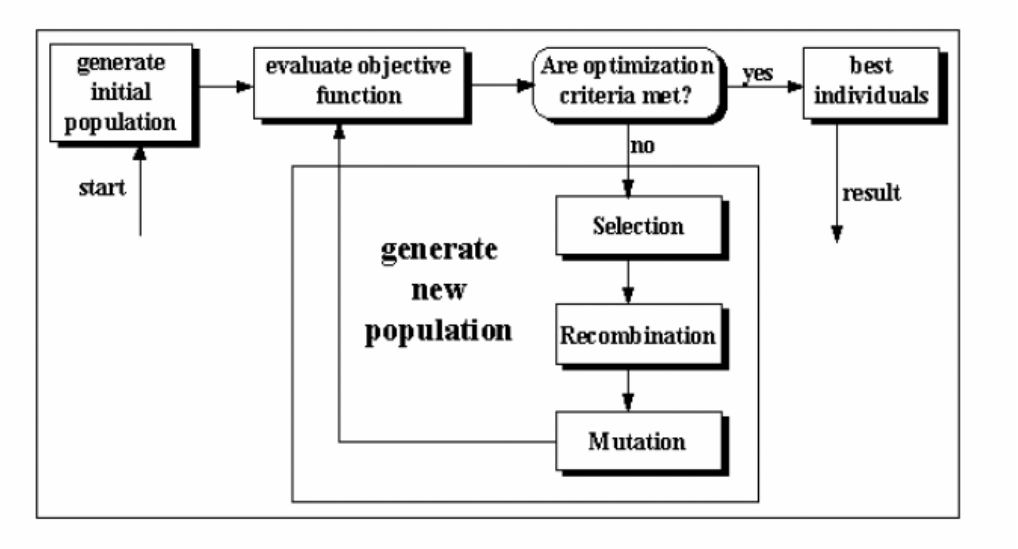
Εικόνα 5.1 Οι τρεις τύποι των παιδιών-ατόμων που μπορούν να προκύψουν

Στην συνέχεια εφαρμόζεται η διαδικασία ελιτισμού η οποία συγκρίνει τα k καλύτερα χρωμοσώματα της προηγούμενης γενιάς με τα k χειρότερα χρωμοσώματα της τρέχουσας γενιάς και κρατάει τα k καλύτερα και από τις δύο γενιές. Μετά και από αυτή την διαδικασία ο εξελικτικός κύκλος έχει ολοκληρωθεί και το σύνολο των χρωμοσωμάτων που δημιουργήθηκαν αποτελούν την καινούργια γενιά. Ο κύκλος αυτός επαναλαμβάνεται μέχρι να εκπληρωθεί κάποιο κριτήριο.

Κάθε φορά, η καινούργια γενιά μπορεί να μην έχει πάντα χρωμόσωμα με βελτιωμένη τιμή προσαρμογής αλλά η συνολική τιμή προσαρμογής είναι μεγαλύτερη. Με την πάροδο των γενεών και την λογική ότι κρατούνται πάντα τα πιο «ισχυρά» χρωμοσώματα θα φτάσουμε σε μία γενιά που θα περιέχει πολύ ισχυρά χρωμοσώματα

αν και ο αλγόριθμος δεν βρίσκει αναγκαστικά τα βέλτιστα. Παρόλα αυτά είναι ένας αλγόριθμος που συγκλίνει γρήγορα και είναι ιδιαίτερα αποδοτικός σε περιπτώσεις που ο χώρος αναζήτησης είναι τελείως άγνωστος.

Παρακάτω παρουσιάζεται ένα σχεδιάγραμμα που μας δείχνει εικονικά την διαδικασία που ακολουθεί ένας γενετικός αλγόριθμος, από την αρχικοποίηση μέχρι να τερματιστεί, καθώς και ο ψευδοκώδικας του αλγορίθμου των βασικών βημάτων υλοποίησής του.



Εικόνα 5.2 Βασικά βήματα γενετικού αλγορίθμου

```
Begin
1.  t=0
2.  initialize population P(t)
3.  compute fitness P(t)
4.  t = t+1
5.  if termination criterion achieved go to step 10
6.  select P(t) from P(t-1)
7.  crossover P(t)
8.  mutate P(t)
9.  go to step 3
10.  Output best and stop
End
```

Εικόνα 5.3 Ψευδοκώδικας του γενετικού αλγορίθμου

5.4 Βελτίωση του συστήματος ασαφής λογικής χρησιμοποιώντας γενετικούς αλγόριθμους

Στο προηγούμενο κεφάλαιο υλοποιήσαμε ένα σύστημα ασαφής λογικής που χρησιμοποιήθηκε για την ταξινόμηση των δεδομένων μας. Οι κανόνες που χαρακτηρίζουν την συμπεριφορά του αναπαράχθηκαν μέσω του αλγορίθμου J.48 που διαθέτει το εργαλείο WEKA. Τα αποτελέσματα σωστής ταξινόμησης μέσω ασαφής λογικής για τις 4 κατηγορίες δεδομένων που διαθέτουμε ήταν σαφώς καλύτερα σε σύγκριση και με τον J.48 αλλά και με τις άλλες μεθόδους που χρησιμοποιήσαμε μέσω του WEKA. Σκοπός μας τώρα είναι χρησιμοποιώντας τους ίδιους κανόνες να μπορέσουμε να βελτιώσουμε ακόμα πιο πολύ την απόδοση του συστήματος με χρήση γενετικών αλγορίθμων.

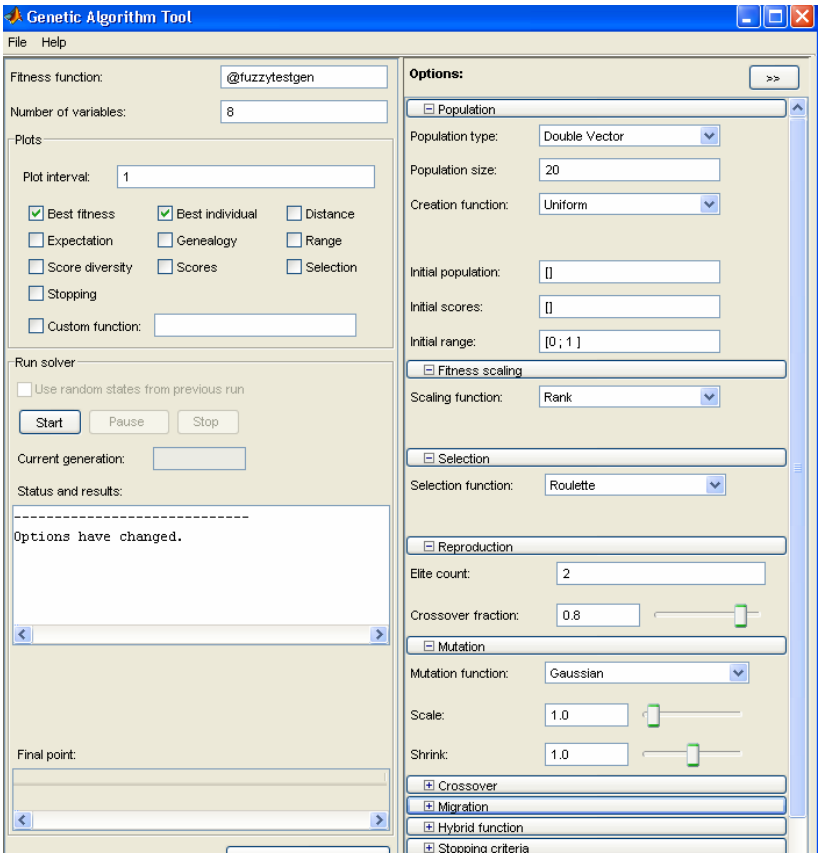
Το ζήτημά μας τώρα είναι ότι από την στιγμή που έχουμε ήδη κανόνες με ποιον τρόπο θα μπορέσουμε να βελτιώσουμε το σύστημά μας. Οι συναρτήσεις συμμετοχής είναι πολύ σημαντικές για την ακρίβεια του συστήματος αλλά και την εφαρμογή των κανόνων. Ωστόσο, αυτές προσδιορίζονται χειροκίνητα και εμπειρικά με βάση τους κανόνες και είναι αρκετά δύσκολο να προσδιοριστεί τέλεια ένα σετ συναρτήσεων ειδικά όταν ο όγκος των δεδομένων είναι τόσο μεγάλος. Έτσι ένας στόχος είναι η βελτίωση των συναρτήσεων συμμετοχής που σημαίνει ότι κάθε συνάρτηση συμμετοχής των χαρακτηριστικών εισόδων θα μετακινηθεί ή θα αλλάξει το εύρος της τόσο ώστε να έχουμε καλύτερη απόδοση. Το δεύτερο στοιχείο που μπορούμε να βελτιώσουμε είναι να προσδιορίσουμε καλύτερα τα βάρη των κανόνων ώστε κάποιοι κανόνες να υπερισχύουν έναντι άλλων λιγότερο σημαντικών. Αυτή η βελτίωση θα είναι και η αφετηρία μας, οπότε το πρώτο που θα επιδιώξουμε είναι η εύρεση των κατάλληλων βαρών για τους κανόνες και με βάση αυτούς θα βελτιστοποιήσουμε και τις συναρτήσεις συμμετοχής.

5.5 Το εργαλείο Genetic Algorithm Toolbox

Για να εφαρμόσουμε τον γενετικό αλγόριθμο χρησιμοποιήσαμε το Genetic Algorithm Toolbox που μας παρέχει η Matlab, το οποίο είναι μια συλλογή συναρτήσεων που επιφέρουν την βελτιστοποίηση προβλημάτων χρησιμοποιώντας

σαν διεπαφή το περιβάλλον της Matlab. Έτσι μας παρέχει συναρτήσεις για την αρχικοποίηση, επιλογή, διασταύρωση και μετάλλαξη καθώς και παραμέτρους με τις οποίες θα πειραματιστούμε έτσι ώστε να έχουμε όσο το δυνατό καλύτερο αποτέλεσμα όπως ο αριθμός των χρωμοσωμάτων και των γενεών και παράμετροι που καθορίζουν τις συναρτήσεις.

Το Genetic Algorithm Toolbox ανοίγει με την εντολή 'gatool' και η μορφή του παρουσιάζεται στην παρακάτω εικόνα



Εικόνα 5.4 το GA Toolbox

Τα κυριότερα χαρακτηριστικά που πρέπει να γνωρίζουμε για το Genetic Algorithm Toolbox είναι τα εξής:

Στην κατηγορία **Fitness Function** εισάγουμε το όνομα της αντικειμενικής συνάρτησης που θέλουμε να ελαχιστοποιήσουμε την τιμή της και στο **Number of Variables** τις μεταβλητές από τις οποίες εξαρτάται η συνάρτηση αυτή. Μπορούμε επίσης να δούμε διάφορα στοιχεία γραφικά , που μας βοηθούν να καταλάβουμε την λειτουργία του γενετικού, με ποιο σημαντικές τις επιλογές **Best Fitness** και **Best Individual** που μας αποτυπώνουν την καλύτερη τιμή προσαρμογής σε σύγκριση με

την μέση τιμή της σε κάθε γενιά , και τα καλύτερα άτομα στα οποία οφείλεται η τιμή αυτή αντίστοιχα. Στην συνέχεια πρέπει να καθορίσουμε τις τιμές ορισμένων σημαντικών παραμέτρων όπως είναι το μέγεθος του πληθυσμού που δηλώνεται στην κατηγορία **Population size**, την συνάρτηση επιλογής στο **Selection function**, τα άτομα με την καλύτερη απόδοση που θα επιζήσουν στην επόμενη γενιά στην κατηγορία **Elite count**, και τα άτομα τα οποία θα υποστούν την διαδικασία διασταύρωσης και μετάλλαξης στην κατηγορία **Crossover fraction**. Για παράδειγμα εάν έχουμε ότι **Population size=20, Elite count=2, Crossover fraction=0.8,** τότε ο τύπος των ατόμων-παιδιών που θα δημιουργηθούν στην επόμενη γενιά θα είναι:

- 2 elite παιδιά-άτομα.
- μας απομένουν 18 άτομα οπότε υπολογίζονται 18*0.8= 14.4 =14 παιδιά-άτομα που θα υποστούν διασταύρωση .
- και τα 4 άτομα που θα απομείνουν θα υποστούν την διαδικασία της μετάλλαξης.

Για την διαδικασία της επιλογής ενός νέου πληθυσμού χρησιμοποιούμε την διαδικασία της ρουλέτας. Η επιλογή γίνεται με βάση την απόδοση κάθε μέλους του πληθυσμού και όσο καλύτερο είναι κάθε μέλος του πληθυσμού τόσο μεγαλύτερη είναι η πιθανότητα να επιλεγεί και να περάσει στην επόμενη γενιά .Τα διάφορα μέλη του πληθυσμού τοποθετούνται στην ρουλέτα με βάση την απόδοσή τους ,οπότε όσο μεγαλύτερη απόδοση έχουν τόσο μεγαλύτερες σχισμές κατέχουν στην ρουλέτα. Έτσι για την κατασκευή της ρουλέτας ακολουθούμε την παρακάτω διαδικασία:

- Κατά την επιλογή, όλα τα χρωμοσώματα αποτιμώνται και δημιουργείται για αυτά μία τιμή προσαρμογής π_i=eval(v_i). Το άθροισμα όλων των τιμών προσαρμογής όλων των αρχικών χρωμοσωμάτων δίνει την ολική προσαρμογή

$$F_{συνολικό} = \sum_{i=1}^{n} \pi_i \qquad (5.5.1)$$

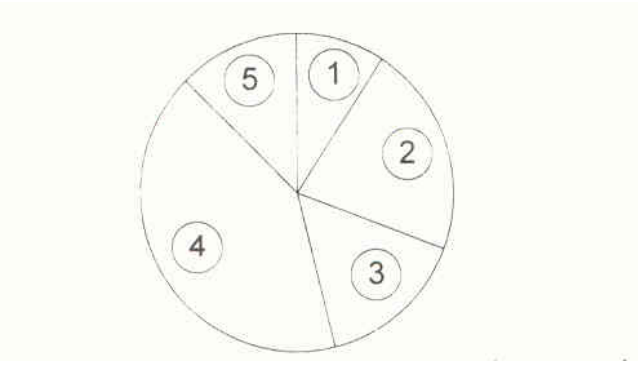
- Η πιθανότητα επιλογής για κάθε χρωμόσωμα δίνεται από τον τύπο

$$p_i = \frac{\pi_i}{F_{συνολικό}} \qquad (5.5.2)$$

- Η συσωρευτική πιθανότητα q_i για κάθε χρωμόσωμα δίνεται από τον τύπο

$$q_i = \sum_{j=1}^{i} p_j \qquad (5.5.3)$$

Στην συνέχεια δημιουργείται ένας τυχαίος αριθμός r στο διάστημα [0,1]. Αν $r < q_1$ τότε επιλέγεται το χρωμόσωμα v_1 ενώ αν $q_{i-1} < r < q_i$ τότε επιλέγεται το χρωμόσωμα v_i. Η επιλογή χρωμοσωμάτων επαναλαμβάνεται για n φορές μέχρι δηλαδή να επιλεγούν n χρωμοσώματα από τον αρχικό πληθυσμό. Η λογική με την οποία γίνεται η επιλογή ακολουθεί τον κανόνα «τροχού ρουλέτας». Δηλαδή ανάλογα με την πιθανότητα επιλογής που έχει το κάθε χρωμόσωμα μπορεί να παρασταθεί σε μία διάταξη με την μορφή πίτας όπως φαίνεται στο σχήμα 3.1.



Εικόνα 5.5: Επιλογή χρωμοσωμάτων με τον μηχανισμό της ρουλέτας

Τέλος στην κατηγορία **Stopping criteria** έχουμε την δυνατότητα να καθορίσουμε πότε θα τερματιστεί ο αλγόριθμος επιλέγοντας αριθμό γενεών ή χρόνο λειτουργίας του.

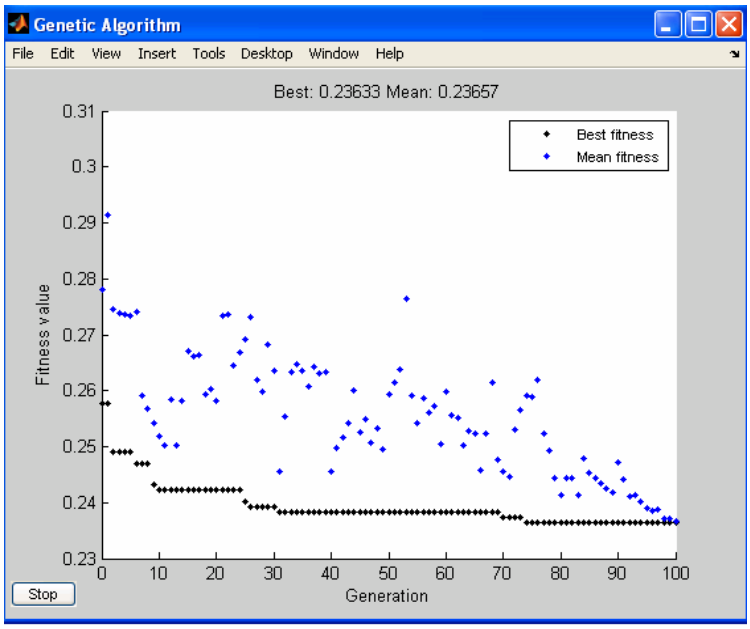
5.6 Παρουσίαση διαδικασίας και αποτελεσμάτων

Όπως αναφέραμε και προηγουμένως το κριτήριο για την επίδοση ενός γενετικού αλγορίθμου αποτελεί η αντικειμενική συνάρτηση - ¨στόχος¨. Η συνάρτησή μας θα περιέχει τον κώδικα τον οποίο χρησιμοποιήσαμε για να υπολογίσουμε το ποσοστό ταξινόμησης με ασαφή λογική. Το αποτέλεσμα της συνάρτησης αυτής μας δίνει το ποσοστό λανθασμένης ταξινόμησης το οποίο και θέλουμε να ελαχιστοποιήσουμε. Ανάλογα με το ποιες από τις παραμέτρους θέλουμε να

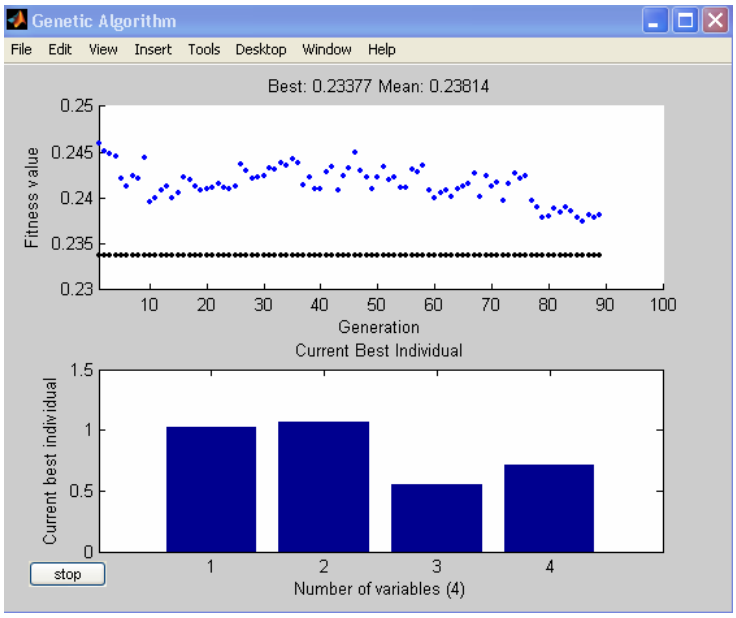
βελτιστοποιήσουμε κάθε φορά θα κάνουμε παρεμβολή στην συνάρτηση έτσι ώστε να καθορίσουμε τα χρωμοσώματα του γενετικού αλγορίθμου. Αυτά με την σειρά τους θα διαμορφώνονται γενιά με γενιά μέσω της γνωστής διαδικασίας, και στο τέλος κρατάμε αυτά τα οποία μας έδωσαν το μικρότερο ποσοστό λανθασμένης ταξινόμησης.

Βελτιστοποίηση ταξινόμησης για τα δεδομένα που αφορούν την λειτουργία στο μετρό

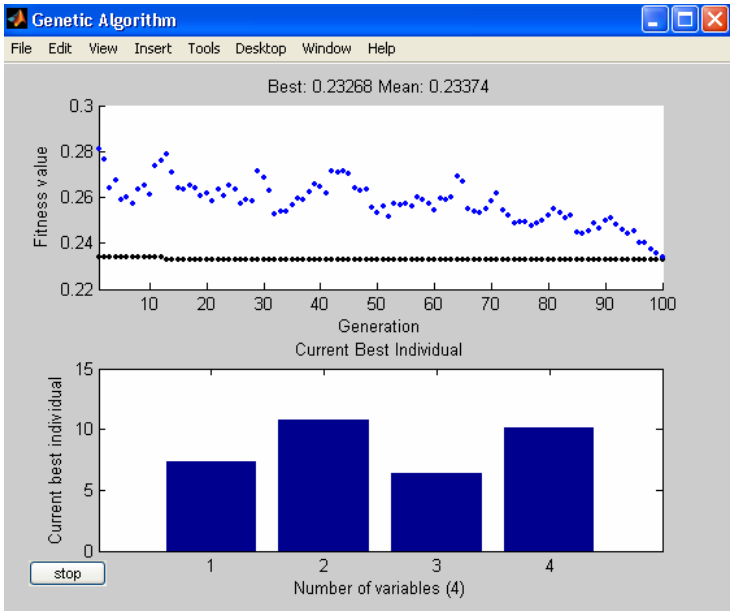
Θα γίνει αναλυτική επεξήγηση της διαδικασίας βελτιστοποίησης για το πρώτο σετ δεδομένων την οποία ακολουθούν και τα υπόλοιπα σετ. Πρωταρχικός στόχος είναι η βελτίωση των βαρών των κανόνων που διέπουν το σύστημα. Αυτό σημαίνει ότι τα χρωμοσώματα του Γ.Α θα αποτελούνται από 12 τιμές όσοι και οι κανόνες. Οι γενιές που επιλέξαμε να γίνει η διαδικασία είναι 100 ενώ ο αριθμός των χρωμοσωμάτων για κάθε γενιά θα είναι 40. Σε κάθε γενιά τα ‘καλά’ χρωμοσώματα επιζούν ώστε να συνεχίσουν και να γίνει η διαδικασία της μετάλλαξης και διασταύρωσης και τα ‘κακά’ χρωμοσώματα απορρίπτονται. Στο τέλος της τελευταίας γενιάς θα επιλεγεί το χρωμόσωμα που μας δίνει την μικρότερη δυνατή τιμή για το σφάλμα ταξινόμησης. Αυτό θα περιέχει τα καινούργια 12 βάρη των κανόνων τα οποία τα προσαρμόζουμε στο σύστημα έχοντας πια μικρότερο σφάλμα ταξινόμησης. Στο παρακάτω σχήμα φαίνεται η καλύτερη και η μέση τιμή της αντικειμενικής συνάρτησης σταδιακά για κάθε γενιά από την αρχή μέχρι το τέλος. Η βελτίωση της απόδοσης είναι αρκετά μεγάλη όπως φαίνεται και στο παρακάτω σχήμα αφού από 74.5% το ποσοστό εύστοχης ταξινόμησης εκτοξεύθηκε στο 76.367 % δηλαδή μια αύξηση 1.867%.



Επόμενος στόχος μας είναι η βελτιστοποίηση των συναρτήσεων συμμετοχής των χαρακτηριστικών. Τα χαρακτηριστικά εισόδου για το συγκεκριμένο σετ δεδομένων είναι 4, οπότε αυτό που θα κάνουμε είναι για κάθε χαρακτηριστικό ξεχωριστά να βρούμε πρώτα τις νέες τιμές των παραμέτρων που καθορίζουν την διακύμανση και μετά αυτές που καθορίζουν την μέση τιμή των Γκαουσιανών συναρτήσεων. Κάθε φορά που θα επιτυγχάνεται μια βελτίωση, τότε αποθηκεύουμε στο σύστημα τις τιμές αυτές ώστε στην συνέχεια της διαδικασίας να γίνει περαιτέρω βελτίωση με βάση τις τιμές αυτές. Αν σε ένα βήμα της διαδικασίας διαπιστώσουμε ότι το σύστημα δεν επιδέχεται βελτίωση, κάτι που είναι πολύ πιθανό, τότε απλά προχωράμε στο επόμενο βήμα. Παρακάτω φαίνεται η νέα βελτίωση του συστήματος καθώς και οι καινούργιες τιμές των παραμέτρων που καθορίζουν την διακύμανση των συναρτήσεων συμμετοχής για το πρώτο χαρακτηριστικό Latency.



Στην συνέχεια δοκιμάζουμε να βελτιώσουμε τις τιμές των παραμέτρων που ορίζουν την μέση τιμή των Γκαουσιανών συναρτήσεων συμμετοχής του χαρακτηριστικού Latency. Η νέα βελτίωση και οι τιμές των παραμέτρων φαίνονται στο παρακάτω σχήμα



Με τα ίδια βήματα συνεχίζουμε την βελτίωση των συναρτήσεων συμμετοχής των άλλων 3 χαρακτηριστικών. Από τα παραπάνω σχήματα διαπιστώνουμε πως οι βελτιώσεις που επιδέχεται το σύστημα πια θα είναι αρκετά μικρές, καθώς βήμα με βήμα με την προσθήκη των γενετικών αλγορίθμων το ποσοστό ορθής ταξινόμησης

φτάνει στα όριά του με αποτέλεσμα να μην μπορούμε να το βελτιώσουμε άλλο. Οι βελτιώσεις που έγιναν σταδιακά παρουσιάζονται παρακάτω:

76,367%	Τροποποίηση των βαρών των κανόνων
76,623%	Τροποποίηση της διακύμανσης των συναρτήσεων συμμετοχής του πρώτου χαρακτηριστικού
76,732%	Τροποποίηση της μέσης τιμής των συναρτήσεων συμμετοχής του πρώτου χαρακτηριστικού
77,16%	Τροποποίηση της μέσης τιμής των συναρτήσεων συμμετοχής του δεύτερου χαρακτηριστικού
77.46%	Τροποποίηση της διακύμανσης των συναρτήσεων συμμετοχής του τρίτου χαρακτηριστικού
77.57%	Τροποποίηση της διακύμανσης των συναρτήσεων συμμετοχής του τέταρτου χαρακτηριστικού

Στο σημείο αυτό αξίζει να αναφέρουμε για ποιο λόγο δεν επιχειρήσαμε να βελτιώσουμε και τις συναρτήσεις συμμετοχής των εξόδων. Οι συναρτήσεις συμμετοχής των εξόδων στην πλειοψηφία τους αποτελούνται από τριγωνοειδείς συναρτήσεις που καθορίζονται από 3 παραμέτρους (δύο για τα άκρα και μια για την γωνία). Σαν επακόλουθο η μία παράμετρος θα πρέπει να έχει αυστηρά μεγαλύτερη τιμή από την άλλη κάτι το οποίο δεν μπορεί να προσδιοριστεί από τον γενετικό αλγόριθμο καθώς οι τιμές που παίρνουν τα χρωμοσώματα είναι τυχαίες και δεν μπορούσε να τις δεχτεί το σύστημα. Για το λόγο αυτό δεν μπορέσαμε να βελτιστοποιήσουμε τις συναρτήσεις εξόδου.

Βελτιστοποίηση ταξινόμησης για τα δεδομένα που αφορούν τις συνθήκες στο μετρό

76,86%	Τροποποίηση των βαρών των κανόνων
77.15%	Τροποποίηση της διακύμανσης των συναρτήσεων συμμετοχής του δεύτερου χαρακτηριστικού

Βελτιστοποίηση ταξινόμησης για τα δεδομένα που αφορούν την εξυπηρέτηση από το προσωπικό στο μετρό

77.93%	Τροποποίηση των βαρών των κανόνων
78.13%	Τροποποίηση της μέσης τιμής των συναρτήσεων συμμετοχής του πρώτου χαρακτηριστικού
78.22%	Τροποποίηση της διακύμανσης των συναρτήσεων συμμετοχής του δεύτερου χαρακτηριστικού
78.42%	Τροποποίηση της διακύμανσης των συναρτήσεων συμμετοχής του τρίτου χαρακτηριστικού

Βελτιστοποίηση ταξινόμησης για τα δεδομένα που αφορούν την ολική ικανοποίηση.

44.66	Τροποποίηση των βαρών των κανόνων
44.87	Τροποποίηση της διακύμανσης των συναρτήσεων συμμετοχής του πρώτου χαρακτηριστικού
45.24	Τροποποίηση της μέσης τιμής των συναρτήσεων συμμετοχής του πρώτου χαρακτηριστικού
45.68	Τροποποίηση της μέσης τιμής των συναρτήσεων συμμετοχής του τρίτου χαρακτηριστικού

5.7 Συμπεράσματα

Στην εργασία αυτή εξετάστηκαν μια σειρά από αλγόριθμους ταξινόμησης που εντάσσονται στον ευρύτερο χώρο της μηχανικής μάθησης. Η δομή της εργασίας αποτελείται από τρεις σημαντικές ενότητες : την εξέταση αλγόριθμων που διαθέτει το λογισμικό μάθησης WEKA, την υλοποίηση συστήματος ασαφής λογικής και την βελτίωση του συστήματος αυτού με την χρήση γενετικών αλγορίθμων με σκοπό την σύγκριση και την βελτίωση των αποτελεσμάτων που αφορούν την σωστή ταξινόμηση.

Σαν αφετηρία χρησιμοποιήσαμε τα αποτελέσματα τα οποία εξήχθησαν έπειτα από τη εφαρμογή αλγορίθμων τους οποίους διαθέτει το λογισμικό μάθησης WEKA. Εκεί έπειτα από δοκιμές επιλέξαμε τους 5 καταλληλότερους αλγόριθμους ταξινόμησης δεδομένων από διάφορες οικογένειες αλγορίθμων τους οποίους εφαρμόσαμε στα σετ δεδομένων, και ανάλογα με τα δεδομένα διαπιστώσαμε ποιος είναι ο κατάλληλος κάθε φορά.

Επόμενο βήμα της εργασίας ήταν η υλοποίηση ενός συστήματος ασαφής λογικής, το οποίο χρησιμοποιήσαμε σαν ταξινομητή και τον εφαρμόσαμε στα δεδομένα. Για την δημιουργία του εκμεταλλευτήκαμε την παραγωγή κανόνων από τον αλγόριθμο J48 που μας παρέχει το WEKA τους οποίους βελτιώσαμε ελαφρώς. Τα αποτελέσματα τα οποία λάβαμε ήταν αρκετά ικανοποιητικά καθώς σε όλες τις κατηγορίες δεδομένων παρατηρήσαμε σημαντικές βελτιώσεις τόσο ως προς τον αλγόριθμο J48 όσο και ως προς τους άλλους αλγόριθμους που εξετάσαμε.

Στο τελευταίο κομμάτι που ήταν και η μεγαλύτερη πρόκληση ,επιχειρήσαμε την περαιτέρω βελτίωση του συστήματος που υλοποιήσαμε, με την χρήση γενετικών αλγορίθμων που αποτελεί μία μέθοδο βελτιστοποίησης πολλών προβλημάτων. Εντέλει είχαμε σαν αποτέλεσμα την δημιουργία ενός αρκετά ικανού συστήματος το οποίο ταξινομεί με μεγάλη επιτυχία τις κλάσεις που θέλαμε να προβλέψουμε.

Τα συνολικά αποτελέσματα που προέκυψαν από όλες τις μεθόδους παρατίθενται στον παρακάτω πίνακα όπου για κάθε μέθοδο παρουσιάζονται τα αντίστοιχα ποσοστά:

ΜΕΘΟΔΟΣ	ΛΕΙΤΟΥΡΓΙΑ	ΣΥΝΘΗΚΕΣ	ΠΡΟΣΩΠΙΚΟ	ΟΛ.ΙΚΑΝΟΠΟΙΗΣΗ
J48	72.0017 %	71.1006 %	72.9389 %	40.3116 %
Best WEKA	74.5102 %	73.2818 %	75.0727 %	41.6748 %
M.O.WEKA	73.27 %	72.483 %	73.75 %	40.175 %
Fuzzy logic	74.89 %	75.59 %	76.65 %	43.39 %
G.A.	77.57 %	77.15 %	78.42 %	45.68 %

Μπορούμε να διαπιστώσουμε ότι τα τελικά ποσοστά είναι αρκετά ικανοποιητικά καθώς η επιτυχής ταξινόμηση με χρήση γενετικών αλγορίθμων είναι αυξημένη κατά 3-4 ποσοστιαίες μονάδες από τον καλύτερο αλγόριθμο που διαθέτει το WEKA κάτι που σημαίνει ότι η βελτίωση που κάναμε είναι πολύ σημαντική. Το γεγονός αυτό μας αποδεικνύει για ακόμα μια φορά πόσο χρήσιμη είναι η εφαρμογή των γενετικών αλγορίθμων για την βελτιστοποίηση ενός προβλήματος.

Επίσης παρατηρούμε ότι με την εφαρμογή του ασαφούς συστήματος είχαμε μια αύξηση της τάξεως του 2% από τον καλύτερο αλγόριθμο που διαθέτει το WEKA εκτός από την κατηγορία που αφορά την λειτουργία των τραίνων όπου είχαμε αύξηση μόνο 0.38%. Η μη ικανοποιητική όμως αυτή βελτίωση ισοσταθμίστηκε με την αύξηση κατά 2.7% περίπου με την χρήση Γ.Α. που ήταν μεγαλύτερη από τις άλλες ομάδες δεδομένων. Τέλος όσον αφορά τον αλγόριθμο J.48 στον οποίο βασίστηκαν οι κανόνες του συστήματος βλέπουμε ότι υπήρξε μια πολύ σημαντική βελτίωση της τάξης του 5-6% η οποία αποκτά μεγάλο βάρος ιδιαίτερα στα πρώτα τρία σετ δεδομένων που σαν αφετηρία είχαμε ένα υψηλό ποσοστό περίπου 71-73% .

ΒΙΒΛΙΟΓΡΑΦΙΑ

Ελληνική βιβλιογραφία:

1. Σάκης Γ. , *Αυτόματη Κατάταξη Μηνυμάτων Ηλεκτρονικού Ταχυδρομείου σε Κατηγορίες*. Πτυχιακή Εργασία ,Τμήμα Πληροφορικής ,Πανεπιστήμιο Αθηνών 2001.

2. Καλαπανίδας Θ.Η. , *Περιβαλλοντική Πρόβλεψη με Μεθόδους Μηχανικής Μάθησης*. Διδακτορική Διατριβή ,Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών,Πάτρα 2003.

3. Δουλφής Ν., *Ανάπτυξη και αξιολόγηση μοντέλων πρόβλεψης της κατανάλωσης φυσικού αερίου για οικιακούς και μικρούς βιομηχανικούς καταναλωτές.*Διπλωματική Εργασία, Πολυτεχνείο Κρήτης Τμήμα Μηχανικών Παργωγής και Διοίκησης ,Χανια 2008.

4. Αλεξανδράτος Τ., *Συκριτική πειραματική ανάλυση μεθόδων επιλογής χαρακτηριστικών και αλγορίθμων ταξινόμησης με χρήση μεθόδων μηχανικής μάθησης.* Διπλωματική Εργασία, Πολυτεχνείο Κρήτης Τμήμα Μηχανικών Παργωγής και Διοίκησης ,Χανια 2008.

5. Ευστράτιος Φ.Γεωργόπουλος, Σπυρίδων Δ.Λυκοθανάσης, *Εισαγωγή στους Γενετικούς Αλγορίθμους,* Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής , ΠΑΤΡΑ 1999

6. Θεοδωρίδης Γ., Πελέκης Ν., *Αποθήκες Δεδομένων και Εξόρυξη γνώσης,* Τμήμα Πηροφορικής ,Πανεπιστήμιο Πειραιώς.

7. Ντούτση Ε., *Ανακάλυψη Γνώσης από Δεδομένα και Εξόρυξη Γνώσης στο εργαλείο WEKA.* Ομάδα Διαχείρισης Δεδομένων, Τμήμα Πληροφορικής ,Πανεπιστήμιο Πειραιώς,2008.

8. Μακρής Χ., Χατζηλυγερούδης Ι., *Εξόρυξη Δεδομένων και Αλγόριθμοι Μάθησης.*Τμήμα Μηχανικών Η/Υ και Πληροφορκής ,Πανεπιστήμιο Πατρών.

9. Βλαχάβας Ι.,Κεφαλάς Π., Βασιλειάδης Ν., Κόκκορας Φ., Σακελλαρίου Η., *Ανακάλυψη Γνώσης σε Βάσεις Δεδομένων.*

10. Σωτήρης Β.Κωτσιαντής ,*Ομάδες ταξινομητών για την αύξηση της ακρίβειας των μεθόδων μηχανικής μάθησης και εξόρυξης γνώσης.*Διδακτορική διατριβή,Σχολή Θετικών επιστημών, Τμήμα Μαθηματικών,Πάτρα 2005.

11. Φωτόπουλος Σ., *Επεξεργασία σημάτων εικόνας και Ασαφής Λογική.*Τμήμα Φυσικής,Πανεπιστήμιο Πατρών ,2005.

12. Χατζηναγνώστου Σ., Γιαννακόπουλος θ., *Ασαφής Λογική και Ασαφή Συστήματα.*Τμήμα Πληροφορικής,Θεσσαλονίκη 2004.

13. Θωμαίδης Ν., *Ασαφής Λογική και λήψη αποφάσεων υπο καθεστώς αβεβαιότητας.* Τμήμα Μηχανικών Παραγωγής και Διοίκησης ,Πολυτεχνείο Κρήτης ,2001.

14. Βλαχάβας Ι.,Κεφαλάς Π., Βασιλειάδης Ν., Κόκκορας Φ., Σακελλαρίου Η., *Γενετικοί Αλγόριθμοι.* Τεχνητή Νοημοσύνη - Β΄Έκδοση.

15. Αγγελής Ε., *Εξελικτικοί αλγόριθμοι, εφαρμογές σε προβλήματα στατιστικού σχεδιασμού.*Τμήμα Πληροφορικής, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης.

16. Σταύρος Ι.Περαντώνης, *υπολογιστική Ευφυία και εφαρμογές.* Ινστιτούτο Πληροφορικής και Τηλεπικοινωνιών ΕΚΕΦΕ 'ΔΗΜΟΚΡΙΤΟΣ'.

Διεθνής βιβλιογραφία:

1. Ian H. Witten, Eibe Frank, 'DATA MINING', *Practical Machine Learning Tools and Techniques.* SECOND EDITION ,2005.

2. Zdravko Markov, *An introduction to the WEKA Data Mining Systems.* Central Connecticut State University.

3. Sugato Basu, Prem Melville, *Machine Learning Group.* Department of Computer Sciences, University of Texas at Austin.

4. Jiawei Han ,Micheline Kamber, *'DATA MINING', Concepts and Techniques.* SECOND EDITION.

5. Tan, P.N., Steinbach, M. and Kumar, *Introduction to Data Mining.* Addison Wesley ,2005.

6. Quinlan J.R., *Induction of decision trees.* Machine learning, 1986.

7. Quinlan J.R., *C4.5: Programs for Machine Learning, 1998.*

8. Kononenko I. and Bratko I., *Information Based Evaluation Criterion for Classifiers Performance,* Machine Learning, 1991.

9. Ting K. M., *Common Issues in Instances Based and Naïve Bayesian Classifiers.* University of Sydney, NSW 2006.

10. Liu H. and Motoda h., *Feature Selection for knowledge Discovery and Data Mining.* Boston, Kluwer Academic, 1998
11. Dudani, S.A., *The Distance-Weighted k-Nearest Neighboor Rule.* IEEE Transactions on Systems ,Man and Cybernetics, Vol .SMC-6,1976.

12. Martin Hellmann, *Fuzzy logic Introduction,* March 2001.

13. S.N.Sivanandam, S.Sumathi, S.N.Deepa.*Introduction to fuzzy logic using MATLAB.*

14. Isabelle Bloch, Alfredo Petrosino, Andrea G.B. Tettmanzi, *Fuzzy Logic and Application p.312* . 6th Internasional Workshop,WILF 2005.

15. Elie Sanchez, Takanori Shibata, Lofti A.Zadeh, *Genetic Algorithms and Fuzzy Logic Systems.*

16. Mohd Saberi Mohamad, Safaai Deris, F*eature selection method using genetic algorithm for the classification of small and high dimension data.* University Teknologi Malaysia.

17. Lipo Wang ,Yaochu Jin, *Fuzzy Systems and Knowledge Discovery.* Second International Conference ,FSKD 2005.

18. Aldenferder S.Mark, Blashfield K. Roger, *Cluster Analysis, Quantitative Applications in the Social Sciences,* SAGE Publications, Inc, 1984