Εφαρμογή Τεχνικών Εξόρυξης σε Πολυδιάστατα Αιματολογικά Δεδομένα

Σταύρος Σταυριανός

Α.Μ. 4337

Επιβλέπων: Βασίλειος Μεγαλοοικονόμου
Καθηγητής Τμήματος Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

Πάτρα,2015

ΠΕΡΙΛΗΨΗ

Η ανάλυση των αιματολογικών δεδομένων είναι μια αρκετά πολύπλοκη διαδικασία. Η κυτταρομετρία ροής, μια μέθοδος ανάλυσης αιματολογικών δεδομένων χρησιμοποιείται για την ταυτόχρονη μέτρηση και ανάλυση πολλαπλών φυσικών ή/και χημικών χαρακτηριστικών μικροσκοπικών σωματιδίων, συνήθως κυττάρων. Σημαντική τεχνολογική πρόοδος στο υλικό/πειραματικά όργανα και την ανάπτυξη φθοριζόντων ιχνηθετών και υποστρωμάτων, έχουν καταστήσει δυνατή την παραγωγή πολύ σύνθετων συνόλων δεδομένων (και μεγάλου αριθμού παραμέτρων) που απαιτούν την ανάπτυξη προηγμένων εργαλείων ανάλυσης. Αν και ο αριθμός των μεταβλητών που μετριούνται ταυτόχρονα μπορεί να αυξηθεί από τους διαφορετικούς δείκτες που χρησιμοποιούνται στην ανάλυση, από τις συνθήκες που επικρατούν κατά τη διεξαγωγή της μέτρησης (π.χ., χρόνος υποκίνησης, συγκέντρωση του ερεθίσματος) ή από τα χρονικά σημεία σε ένα in-vitro πείραμα ή κλινική δοκιμή τα δεδομένα αυτά δεν μπορούν να αξιοποιηθούν κατάλληλα από τους χρήστες με αποτέλεσμα την απώλεια σημαντικής πληροφορίας. Μέχρι σήμερα η ανάλυση βασίζεται σε επιλογή από τον χρήστη δυάδων παραμέτρων που απεικονίζονται δυσδιάστατα. Την ανάλυση της πρώτης δυάδας, ακολουθεί δεύτερη και ούτω καθεξής. Αυτή η διαδοχική διπαραμετρική ανάλυση είναι χρονοβώρα, απαιτεί μεγάλη εμπειρία και δεν αναδεικνύει όλες τις σχέσεις των δεδομένων.

Αρκετές προσπάθειες έχουν γίνει για να απλοποιηθεί η ανάλυση. Αυτές μπορούν να διαιρεθούν κατά προσέγγιση σε δύο κύριες κατηγορίες: εποπτευόμενες (supervised) και μη εποπτευόμενες (unsupervised). Οι περισσότερες από αυτές τις νέες προσεγγίσεις είναι κυρίως explorative και όχι ποσοτικές. Τα ιστόγραμμα και οι γραφικές παραστάσεις σημείων είναι πολύ απλοί και διαισθητικοί τρόποι για την ανάλυση δεδομένων κυτταρομετρίας ροής. Όσο περιλαμβάνουμε στην ανάλυση όλο και περισσότερες παραμέτρους, ο αριθμός των πιθανών συνδυασμών ($2n$, όπου το n είναι ο αριθμός παραμέτρων) αυξάνεται εκθετικά. Κατά συνέπεια, απαιτείται απλοποίηση των συνόλων δεδομένων. Αλγόριθμοι συσταδοποίησης έχουν χρησιμοποιηθεί για την εύρεση ομοιοτήτων και διαφορών μεταξύ των

δειγμάτων. Επίσης δεδομένου ότι τα δεδομένα κυτταρομετρίας ροής είναι υψηλής διαστατικότητας τεχνικές όπως η PCA έχουν εφαρμοστεί για μειώσουν τον αριθμό των διαστάσεων. Στη παρούσα εργασία θα γίνει μελέτη των τεχνικών που έχουν προταθεί στην βιβλιογραφία για την ανάλυση δεδομένων κυτταρομετρίας ροής και θα υλοποιηθούν κάποιες από αυτές.

Στην παρούσα διπλωματική χρησιμοποιήθηκαν πραγματικά δεδομένα και με εργαλεία από την εξόρυξη δεδομένων προσπάθησα να βγάλω χρήσιμα συμπεράσματα.

ΠΕΡΙΕΧΟΜΕΝΑ

1. ΚΥΤΤΑΡΟΜΕΤΡΙΑ ΡΟΗΣ

1.1 Τι είναι Κυτταρομετρία ροής;

Η Κυτταρομετρία ροής είναι μία τεχνική για τη μέτρηση και τον χαρακτηρισμό μικροσκοπικών σωματιδίων σε ρέον υγρό. Επιτρέπει την ταυτόχρονη ανάλυση πολλών παραμέτρων των φυσικών ή χημικών χαρακτηριστικών μεμονωμένων κυττάρων τα οποία ρέουν διαμέσου μιας συσκευής οπτικής ή/και ηλεκτρονιακής ανίχνευσης. Στις εφαρμογές της κυτταρομετρίας ροής μπορεί να περιλαμβάνεται και η ταξινόμηση των συστατικών βάσει φυσικών παραμέτρων.Η κυτταρομετρία ροής παρουσιάζει πολλά πλεονεκτήματα σε σχέση με άλλες τεχνικές όπως η μεγάλη ποικιλία υλικών που μπορουν να αναλυθούν ,υψηλή ταχύτητα ,υψηλή ευαισθησία και δυνατότητα πολυπαραμετρικής ανάλυσης. Τα τελευταία χρόνια με την ραγδαία αυξήση της τεχνολογίας και της επιστήμης της εξόρυξης γνώσης από δεδομένα ενδυνάμωσε τις δυνατότητες της κυτταρομετρίας ροής.

1.1.1 Ιστορικά

Κατά την διάρκεια του Β' Παγκοσμίου πολέμου (1947) ο Gucker εφτιάξε ένα όργανο το οποίο μπορούσε να ανιχνεύση βακτηρίδια και σπόρους στον αέρα. Η αρχή λειτουργίας αυτού του οργάνου αποτέλεσαι την ιδέα για την κατασκεύη από τον Coulter το 1956 του πρώτου αυτοματοποιημένου αναλυτή των έμμορφων συστατικών του αίματος.Την επόμενη δεκαετία (1960) και με την ανάπτυξη των ηλεκτρονικών υπολογιστών ο Luis Kamentsky σε συνεργασία με άλλους επιστήμονες παρουσίασε τον τον πρώτο ολοκληρωμένο κυτταρομετρητή ροής με δυνατότητα αναγνώρισης των κυτταρικών πληθυσμών και στατιστικής ανάλυσης των αποτελεσμάτων.Στην συνέχεια πολλοί επιστήμονες παρουσιάζουν βελτιώσεις και νέα χαρακτηριστικά σε δικές τους συσκεύες όπως ο Van Dilla την εισαγωγή του laser στην συσκεύη του.Έτσι την δεκαετία του 1970 ξεκινάει η εμπορική διάθεση συσκεύων κυτταρομετρίας ροής. Σήμερα με την πρόοδο της τεχνολογίας και τον ανταγωνισμό μεταξυ των εταιριών οι σημερινές συσκεύες με την πολλαπλών φακών laser ανθιχνεύουν μέχρι και 20 διαφορετικές παραμέτρους.

1.1.2 Αρχές της κυτταρομετρίας ροής

Μία δέσμη φωτός (συνήθως δέσμη λέιζερ) ενός μεμονωμένου μήκους κύματος κατευθύνεται διαμέσου μιας υδροδυναμικά συγκλίνουσας ροής υγρού. Ένας αριθμός ανιχνευτών περιβάλλουν το σημείο όπου η δέσμη του φωτός διαπερνάει τη ροή του υγρού: ένας σε ευθυγράμμιση με τη δέσμη φωτός, κάποιοι άλλοι κάθετοι σε αυτήν και ένας ή περισσότεροι ανιχνευτές φθορισμού. Κάθε σωματίδιο μεταξύ 0.2 και 150 μικρομέτρων αιωρούμενο στο υγρό που περνά διαμέσου της δέσμης σκεδάζει το φως προς κάποια κατεύθυνση και παράλληλα τα φθορίζοντα χημικά που βρίσκονται στο σωματίδιο ή επί της επιφάνειάς του μπορούν να διεγερθούν και να εκπέμψουν φως άλλου μήκους κύματος από αυτό της πηγής. Αυτός ο συνδυασμός σκεδασμένου και φθορίζοντος φωτός παραλαμβάνεται από τους ανιχνευτές και μετά από αναλύσεις είναι δυνατή η αποκόμιση πληροφοριών σχετικών με τη φυσική και χημική δομή κάθε μεμονωμένου σωματιδίου. Η εμπρόσθια σκέδαση ”FSC” (εκ του Forward Scattering) σχετίζεται με τον όγκο του κυττάρου και η πλάγια σκέδαση ”SSC” (εκ του Side Scattering) εξαρτάται από την εσωτερική πολυπλοκότητα του σωματιδίου (π.χ., σχήμα του πυρήνα, αριθμός κυτταροπλασματικών σωματιδίων ή αδρότητα κυτταρικής μεμβράνης). Κάποιες συσκευές κυτταρομετρίας ροής στην αγορά δεν περιλαμβάνουν τους ανιχνευτές φθορισμού και χρησιμοποιούν μόνο τη σκέδαση του φωτός για τις μετρήσεις. Άλλες, παράγουν απεικονίσεις του φθορισμού, της σκέδασης και της έντασης του φωτός για κάθε κύτταρο.

1.1.3 Συσκευές κυτταρομετρίας ροής

Οι σύγχρονες συσκευές κυτταρομετρίας ροής έχουν τη δυνατότητα ανάλυσης μερικών χιλιάδων σωματιδίων το δευτερόλεπτο, σε ”πραγματικό χρόνο”, και ορισμένες έχουν επιπλέον τη δυνατότητα ενεργού διαχωρισμού και απομόνωσης σωματιδίων με συγκεκριμένες ιδιότητες. Μια συσκευή κυτταρομετρίας ροής μοιάζει με ένα μικροσκόπιο με τη διαφορά ότι αντί να παράγει την εικόνα ενός κυττάρου, η κυτταρομετρία ροής παρέχει δεδομένα για τα χαρακτηριστικά ενός μεγάλου αριθμού κυττάρων σε λίγο χρόνο. Για την ανάλυση στερεών ιστών θα πρέπει πρώτα να παρασκευαστεί αναιώρημα απομονωμένων (όχι συσσωματικών) κυττάρων.

Μια συσκευή κυτταρομετρίας ροής αποτελείται από 5 κύρια συστατικά μέρη:

- μία ροή υγρού περιβλήμματος (sheath fluid) η οποία μεταφέρει και διευθετεί τα κύτταρα έτσι ώστε να περνούν ένα-ένα από τη δέσμη του φωτός λέιζερ

- ένα οπτικό σύστημα που περιλαμβάνει μονοχρωματικές πηγές φωτός που είναι γνωστές σε όλους μας με την ονομασία laser (μία ή και περισσότερες), καθώς και ένα πολύπλοκο σύστημα φακών, φωτοανιχνευτών και φωτοπολλαπλασιαστών.

- έναν ανιχνευτή και ένα σύστημα τροποποίησης σήματος από αναλογικό σε ψηφιακό (Analogue-to-Digital Conversion - ADC) - το οποίο μετατρέπει σήματα πρόσθιας (FSC) και πλάγιας (SSC) σκέδασης καθώς και σήματα φθορισμού από το φως σε ηλεκτρικά σήματα τα οποία μπορούν να επεξεργαστούν από τον Η/Υ

- ένα σύστημα ενίσχυσης - δυνατότητα γραμμικής ή λογαριθμικής απεικόνισης

- έναν Η/Υ για την μετατροπή των φωτεινών σημάτων σε πληροφορίες και για την ανάλυση των δεδομένων.

Τα σύγχρονα όργανα κυτταρομετρίας ροής διαθέτουν πολλαπλά λέιζερ και ανιχνευτές φθορισμού . Η αύξηση του αριθμού των λέιζερ και των ανιχνευτών φθορισμού επιτρέπει την πολλαπλή σήμανση με χρήση αντισωμάτων και συνεπώς δίνει τη δυνατότητα για πιο ακριβή ταυτοποίηση των πληθυσμών που αποτελούν το στόχο βάσει του φαινοτύπου των τελευταίων. Ορισμένα όργανα μπορούν ακόμα και να φωτογραφήσουν μεμονωμένα κύτταρα επιτρέποντας τον προσδιορισμό της πηγής του σήματος φθορισμού εντός ή επί της επιφανείας των κυττάρων.

2. ΤΕΧΝΙΚΕΣ ΕΞΟΡΥΞΗΣ ΔΕΔΟΜΕΝΩΝ

2.1 Εισαγωγή

Η εξόρυξη γνώσης από μεγάλες αποθήκες δεδομένων έχει εξελιχθεί σε ένα από τα βασικότερα ερευνητικά ζητήματα στον τομέα των βάσεων δεδομένων,των μηχανών γνώσης, της στατιστικής, καθώς επίσης και ως μία σημαντική ευκαιρία για καινοτομία στις επιχειρήσεις.Οι δικτυακές εφαρμογές που διαχειρίζονται μεγάλες αποθήκες δεδομένων έχουν αρχίσει να κάνουν χρήση διαφόρων τεχνικών εξόρυξης δεδομένων, με σκοπό τη βελτίωση της ποιότητας των παρεχόμενων υπηρεσιών μέσω μελέτης της συμπεριφοράς των πελατών και της εξαγωγής χρήσιμων συμπερασμάτων απο αυτήν.

Η τελευταία δεκαετία έχει επιφέρει μια αλματώδη αύξηση στην παραγωγή και συλλογή δεδομένων.Η πρόοδος στην τεχνολογία των βάσεων δεδομένων μας παρέχει νέες τεχνικές για την αποδοτική και αποτελεσματική συλλογή, αποθύκευση και διαχείρηση των δεδομένων. Κάθε χρόνο παράγονται τεράστιοι όγκοι δεδομένων από εταιρείες και πανεπιστήμια τα οποία αποθηκέυονται σε μεγάλες βάσεις δεδομένων. Επίσης η δυνατότητα ανάλυσης και ερμηνείας των συνόλων δεδομένων, και η εξαγωγή χρήσιμης γνώσης από αυτά έχει ξεπεράσει κάθε όριο και η ανάγκη για μία νέα γενιά εργαλείων και τεχνικών για ευφυή ανάλυση βάσεων δεδομένων έχει δημιουργηθεί. Αυτή η ανάγκη έχει προσελκύσει την προσοχή των ερευνητών από διάφορες περιοχές (τεχνηκή νοημοσήνη,στατιστική,αποθήκες δεδομένων,διαδραστική ανάλυση και επεξεργασία,έμπειρα συστήματα και οπτικοποίηση δεδομένων) και ένας νέος ερευνητικός τομέας δημιουργείται, γνωστός ως εξόρυξη δεδομένων και γνώσης (Data and Knowledge Mining).

2.2 Εξόρυξη γνώσης και δεδομένων

Η ανακάλυψη γνώσης από βάσεις δεδομένων (Knowledge Discoevry in Databases -KDD) αναφέρεται στη διεργασία εξόρυξης γνώσης απο τις μεγάλες αποθήκες δεδομένων.Ο όρος εξόρυξη δεδομένω χρησιμοποιείται ως συνώνυμο της ανακάλυ-

ψης γνώσης από βάσεις δεδομένων,καθώς επίσης και για αναφορά στις πραγματικές τεχνικές που χρησιμοποιούνται για την ασνάλυση και την εξαγωγή της απο διάφορα σύνολα δεδομένων.Πόλλοι ερευνητές θεωρούν τον όρπ εξόρυξη δεδομένων μη αντιπροσωπευτικό της διαδικασίας που αντιπρσωπεύει, υποστιρίζοντας ότι ο όρος εξόρυξη γνώσης θασ ήταν μια πιο κατάλληλη περιγραφή.Ο όρος εξόρυξη δεδομένων είναι αυτός που έχει επικρατήσει και χαρακτηρίζει τη διαδικασία της εύρεσης δομών γνώσης οι οποίες περιγράφουν με ακρίβεια μεγάλα σύνολα πρωτογενών δεδομένων. Οι δομές αυτές αναδεικνύουν γνώση (συσχετίσεις ή κανόνες) ποθ είναι κρυμμένες μέσα στα δεδομένα και δεν μπορούν να εξαχθούν από τον άνθρωπο-χρήστη της βάσης δεδομένωνμε ”γυμνό” μάτι.

2.3 Αλγόριθμοι κατηγοροποίησης

2.3.1 Εισαγωγή

Το πρόβλημα της κατηγοριοποίησης έχει μελετηθεί εκτενώς στη στατιστική, στην αναγνώριση προτύπων και στη μηχανική μάθηση στα πλαίσια του προβλήματος της ανάκτησης ή εξαγωγής γνώσης από σύνολα δεδομένων. Χαρακτηρίζεται ως μία από τις βασικές εργασίες στη διαδικασία εξόρυξης γνώσης η οποία αποσκοπεί στη ανάθεση ενός στοιχείου σε ένα προκαθορισμένο σύνολο κατηγοριών (classes). Η κατηγοριοποίηση (classification) μπορεί να περιγράφει λοιπόν ως μια λειτουργία που αντιστοιχεί (κατηγοριοποιεί) ένα στοιχείο σε μία από τις διαφορετικές κατηγορίες που έχουν προκαθοριστεί. Αναφέρουμε ότι ο όρος ταξινόμηση χρησημοποιείται στη βιβλιογραφία ως συνώνυμο της κατηγοριοποίησης. Η κατηγοριοποίηση χαρακτιρίζεται από ένα καλά καθορισμένο σύνολο κατηγοριών καθώς και ένα σύνολο από προκατηγοριοποιημένα παραδείγματα. Γενικά, ο στόχος της διαδικασίας κατηγοριοποίησης είναι η δημιουργία ενός μοντέλου που θα μπορεί να χρησιμοποιηθεί για την κατηγοριοποίηση μελλοντικών δεδομένων των οποίων η κατηγοριοποίηση είναι άγνωστη. Πιο συγκεκριμένα,η κατηγοριοποίση δεδομένων μπορεί να περιγραφεί ως μία διαδικασία δύο βημάτων:

1. Εκμάθηση:Σε αυτό το βήμα χτίζεται ένα μοντέλο,περιγράφοντας ένα προκαθορισμένο σύνολο από κατηγορίες δεδομένων. Τα δεδομένα εκπαίδευσης αναλύονται από έναν αλγόριθμο κατηγοριοποίησης για να κατασκευάσουν στη συνέχεια το μοντέλο. Τα στοιχεία που αποτελούν το σύνολο κατάρτισης επιλέγονται τυχαία από έναν πληθυσμό δεδομένων και ανήκουν σε μίοα από τις προκαθορισμένες κατηγορίες. Δεδομένου ότι η κατηγορία των δειγμάτων εκπαίδευσης είναι γνωστή, αυτό το βήμα είναι επίσης γνωστό σαν

εποπτεθόμενη μάθηση.Το μοντέλο που ορίζεται,γνωσό και ως κατηγοριοποιητής, αναπαριστάται με τη μορφή κανόνων κατηγοριοποίησης ,δέντρων απόφασης ή μαθηματικών τύπων.

2. Κατηγοριοποίηση:Σε αυτό το βήμα χρησημοποιούνται τα δοκιμαστικά δεδομένα για να υπολογίσουν την ακρίβεια του μοντέλου. Υπάρχουν διάφορες μέθοδοι για να εκτιμηθεί η ακρίβεια του κατηγοριοποιητή . Τα δεδομένα εκπαίδευσης επιλέγονται τυχαία και είναι ανεξάρτητα. Το μοντέλο κατηγοριοποιεί κάθε ένα από τα δοκιμαστικά παραδείγματα. Στη συνέχεια η κατηγορία που ανήκουν τα δεδομένα με βάση το σύνολο δοκιμαστικών δεδομένων συκρίνεται με την πρόβλεψη που έκανε το μοντέλο για την κατηγορία.Η ακρίβεια του μοντέλου σε ένα πορκαθορισμένο σύνολο δεδομένων δοκιμής είναι το ποσοστό των δειγμάτων δοκιμής που κατηγοριοποιήθηκαν σωστά από το υπό εκπαίδευση μοντέλο. Εάν η ακρίβεια του μοντέλου θεωρείται ως αποδεκτή,το μοντέλο μπορεί πλέον να χρησημοποιηθεί για να κατηγοριοποιήσει τα μελλοντικά δείγματα δεδομένων (αντικείμενα), των οποίων η κατηγοριοποίηση είναι άγνωστη.

2.3.2 Bayesian Κατηγοροποίηση

Bayesian κατηγοριοποίηση βασίζεται στη στατιστική θεωρία κατηγοριοποίησης του Bayes. Ο στόχος είναι να κατηγοριοποιηθεί ένα δείγμα X σε μια απο τις δεδομένες κατηγορίες $C_1, C_2, ..., C_n$ χρησιμοποιώντας ένα μοντέλο πιθανότητας που ορίζεται σύμφωνα με τη θεωρία Bayes. Κάθε κατηγορία χαρακτηρίζεται από μια εκ των προτέρων πιθανότητα παρατήρησης της κλάσης C_i.Επίσης υποθέτουμε ότι το δεδομένο δείγμα X ανήκει σε μια κλάση C_i,με την υπό συνθήκη συνάρτηση πυκνότητας πιθανότητας $p(X/C_i) \in [0, 1]$. Κατόπιν,χρησιμοποιώντας τους ανωτέρω ορισμούς και βασιζόμενοι στη θεωρία Bayes, καθορίζουμε την εκ των υστέρων πιθανότητα $p(c_i/x)$ ως εξής:

$p(c_i|X = \frac{p(c_i|X)p(c_i)}{p(X)}$

Ο απλούστερος Bayesian κατηγοριοποιητής είναι ο γνωστός Naive Bayesian κατηγοριοποιητής. Αυτός υποθέτει ότι η επίδραση ενός γνωρίσματος σε μια δεδομένη κατηγορία είναι ανεξάρτηση από τις τιμές των άλλων γνωρισμάτων. Αυτή η υπόθεση γίνεται για να απλοποιήσει τους υπολογισμούς που εμπλέκονται και καλείται υπό συνθήκη ανεξαρτησία κατηγορίας.Ένας άλλος Baysian κατηγοριοποιητής είναι τα Bayesian Belief Networks .Είναι γραφικά μοντέλα όπου χρησιμοποιούμενα αντίθετα με τους Naive Bayesian κατηγοριοποιητές, επιτρέπουν τη

παρουσίαση των εξαρτήσεων μεταξύ των υποσυνόλων των γνωρισμάτων.

Naive Bayesian κατηγοριοποιητής

Υποθέστε ότι έχουμε ένα σύνολο δεδομένων S και έστω κάθε δείγμα δεδομένων αντιπροσωπεύεται από ένα n-διάστατο χαρακτηριστικό διάνυσμα,$X = (x_1, x_2, ..., x_n)$, το οποίο απεικονίζει τις n μετρήσεις που γίνοται στο δείγμα για τα n γνωρίσματα, $A_1, A_2, ..., A_n$. Υποθέστε ότι υπάρχουν m κατηγορίες $C_1, C_2, ..., C_m$.Κατόπιν δεδομένου ενός αγνώστου δείγματος δεδομένων ,X,ο κατηγοριοποιητής θα προβλέψει ότι το X ανήκει στην κατηγορία που έχει την υψηλότερη εκ των υστέρων πιθανότητα δεδομένου του X. Αυτό υπονοεί ότι ο κατηγοριοποιητής Naive Bayesian αναθέτει το δείγμα X στην κατηγορία C_i εάν και μόνο εάν:

$p(C_i|X) > p(C_j|X)$ for $1 \leq j \leq m, m \neq i$

Κατά συνέπεια , ο στόχος είναι να μεγιστοποιηθεί η εκ των υστέρων υπόθεση.Η κατηγορία C_i για την οποία η πιθανότητα $p(C_i|X)$ μεγιστοποιείται καλείται μέγιστη μεταγενέστερη υπόθεση.Ο Naive Bayesian κατηγοριοποιητής υπολογίζει τις υπο συνθήκη πιθανότητες της κατηγορίας υποθέτοντας υπό συνθήκη ανεξαρτησία. Κατόπιν, πρέπει να υποθέσουμε ότι $p(X|C_i) = p(x_1|C_i)...p(x_n|C_i)$ και κάθε μια από τις πιθανότητες $p(x_j|C_i)$ μπορεί να υπολογιστεί από τα δεδομένα εκαπίδευσης. Κατά συν[επεια, ο Naive Bayesian κατηγοριοποιητής είναι μια αποδεκτή τεχνική. Θεωριτικά, οι Bayesian κατηγοριοποιητές έχουν το ελάχιστο ποσοστό σφάλματος σε σύγκριση με όλους τους άλλους κατηγοριοποιητές.Στην πράξη,όμως,αυτό δεν συμβαίνει πάντα λόγω των υποθέσεαν που απαιτούνται να γίνουν κατά τη χρήση τους,όπως η υπό συνθήκη ανεξαρτησία , και η έλλειψη διαθέσιμων δεδομένων για τον ακριβή υπολογισμό των υπό συνθήκη πιθανοτήτων.Ωστόσο,έχει βρεθεί ότι είναι συκορίσιμοι με τα δέντρα απόφασης και τους κατηγοριοητές που βασίζονται σε νευρωτικά δίκτυα σε μερικές εφαρμογές.

Bayesian Belief Networks

Τα Bayesian Belief Networks προσδιορίζουν τις συνδεδεμένες υπό συνθήκη κατανομές πιθανότητας στοχεύοντας στο να λάβουν υπόψη τις εξαρτήσεις που μπορούν να υπάρξουν μεταξύ των μεταβλητών. Ένα Belief Network καθορίζεται από δύο στοιχεία.Το πρώτο είναι ένας κατευθυνόμενος ακυκλικός γράφος,όπου κάθε κόμβος αντιπροσωπεύει μια τυχαία μεταβλητή και κάθε τόξο αντιπροσωπεύειμια εξάρτηση πιθανοτήτων.Εάν το τόξο έχει αρχή έναν κόμβο Y και πέρας έναν κόμβο Z,τότε το Y είναι γονέας του Z και το Z είναι απόγονος του Y.Κάθε μετα-

βλητή είναι ανεξάρτητη από τους μη προγόνους της στο γράφο,δεδομένου των γονέων της. Το δεύτερο στοιχείο που καθορίζει έαν Belief Network ασποτελείται από έναν πίνακα υπό σηνθήκη πιθανότητας(Conditional Probability Table:CPT) για κάθε μεταβλητή. Ο CPT για μια μεταβλητή X προσδιορίζει τη δεσμευμένη κατανομή $p(X|parent(X))$. Η συνδυασμένη πιθανότητα κάθε συνόλου $(x_1, x_2, ..., x_n)$ που αντιστοιχεί στα γνωρίσματα $A_1, A_2, ..., A_n$ δίνεται απο την ακόλουθη εξίσωση:

$$p(x_1, x_2, ..., x_n) = \prod_{i=1}^{n} p(x_i|parent(x_i))$$

όπου $parent(x_i)$ είναι ο γονέας του x_i και $p(x_i|parent(x_i))$ αντιστοιχεί στις υπό συνθήκη καταχωρήσεις του CPT για το x_i. Ένας από τους κόμβους του δικτύου μπορεί να επιλεγεί ως output κόμβος αντιπροσωπεύντας τα γνωρίσματα μια κατηγορίας. Οι αλγόριθμοι συμπεράσματος για εκμάθηση μπορούν να εφαρμοστούν στο δίκτυο.

2.3.3 Δέντρα απόφασης

Τα δέντρα απόφασης είναι μια από τις ευρέως χρησιμοποιούμενες τεχνικές για την κατηγοριοποίηση και την πρόβλεψη.Διάφοροι δημοφιλείς κατηγοριοποιητές κατασκευάζουν τα δέντρα απόφασης ως μοντέλα κατηγοριοποίησης. Ένα δέντρο απόφασης κατασκευάζεται με βάση ένα σύνολο εκπαίδευσης προ-κατηγοριοποιημένων δεδομένων.Κάθε ένας από τους εσωτερικούς κόμβους του δέντρου απόφασης προσδιορίζει τον έλεγχο ενός γνωρίσματος και κάθε κλαδί που κατεβαίνει απίο εκείνον τον κόμβο αντιστοιχεί σε μια από τις πιθανές τιμές για το συγκεκριμένο γνώρισμα.Επίσης, κάθε φύλλο αντιστοιχεί σε μια από τις κατηγορίες που έχουν οριστεί.Η διαδικασία για την κατηγοροποίηση ενός νέου δείγματος με βάση ένα δέντρο απόφασης είναι η ακόλουθη: ξεκινώντας από την ρίζα του δέντρου και εξετάζοντας τα γνωρίσματα που καθορίζονται από τον κόμβο αυτό προσδιορίζονται διαδοχικά οι εσωτερικοί κόμβοι που θα επισκεφτούμε έως ότου καταλήξουμε σε ένα φύλλο.Σε κάθε εσωτερικό κόμβο ελέγχεται εάν το δείγμα ικανοποιεί το συγκεκριμένο κόμβο.Η έκβαση αυτής της δοκιμής σε έναν εσωτερικό κόμβο καθορίζει το κλαδί που θα διασχίσουμε στη συνέχεια καθώς και τον επόμενο κόμβο που θα επισκεφτούμε. Η κατηγορία του υπό μελέτη δείγματος είναι η κατηογορία του τελικού ο οποίος αντιστοιχεί σε φύλλο του δέντρου. Διάφοροι αλγόριθμοι κατασκευής των δέντρων απόφασης έχουν ανταπτυχθεί κατά την διάρκεια των τελευταίων ετών.Μερικοί από αυτούς είναι οι: ID3,C4.5,SPRINT,SLIQ,CART,RainForest κ.λπ. Στην συνέχεια παρατίθονται τρείς από αυτούς τους αλγόριθμους.

Αλγόριθμος ID3

Ο ID3 (Iterative Dichotomiser 3) είναι ένας αλγόριθμος, ο οποίος χρησιμοποιείται για να παραγάγει ένα δέντρο απόφασης.

Ο αλγόριθμος είναι βασισμένος στο Ξυράφι του Όκαμ: προτιμά τα μικρότερα δέντρα απόφασης (απλούστερες θεωρίες) από μεγαλύτερες. Εντούτοις, δεν παράγει πάντα το μικρότερο δέντρο, και για αυτό τον λόγο είναι ευρετικός. Το Ξυράφι του Όκαμ τυποποιείται χρησιμοποιώντας την έννοια της εντροπίας πληροφοριών: $I_E(I) = -\sum_{j=1}^{m} f(i,j) \log f(i,j)$

Ο αλγόριθμος ID3 μπορεί να συνοψιστεί ως εξής:

1. Πάρτε όλες τις αχρησιμοποίητες ιδιότητες και υπολογίστε την εντροπία τους λαμβάνοντας υπόψη δείγματα δοκιμής
2. Επιλέξτε την ιδιότητα για την οποία η εντροπία είναι ελάχιστη
3. Δημιουργήστε έναν κόμβο που να περιέχει αυτή την ιδιότητα

Ο αλγόριθμος

Ο πραγματικός αλγόριθμος είναι ο ακόλουθος:

- Δημιούργησε ένα αρχικό κόμβο για το δέντρο
- Εάν όλα τα παραδείγματα είναι θετικά, Επέστρεψε την Ρίζα ως δέντρο με ένα κόμβο, με ετικέτα= +.
- Εάν όλα τα παραδείγματα είναι αρνητικά, Επέστρεψε την Ρίζα ως δέντρο με ένα κόμβο, με ετικέτα= -.
- Εάν ο αριθμός των προβλεπόμενων ιδιοτήτων είναι κενός, τότε Επέστρεψε την Ρίζα ως δέντρο με ένα κόμβο, με ετικέτα = την πιο κοινή τιμή της ιδιότητας-στόχου των παραδειγμάτων.
- Αλλιώς Ξεκίνα
  - Α = Η ιδιότητα που κατηγοριοποιεί καλύτερα τα παραδείγματα.
  - Ιδιότητα Δέντρου απόφασης για τη ρίζα = Α.
  - Για κάθε πιθανή τιμή, v_i, του Α,

- Πρόσθεσε ένα νέο κλάδο κάτω από τη Ρίζα, που να αντιστοιχεί στη δοκιμή A = v_i.
- Θέσε Παραδείγματα(v_i), ως το υποσύνολο των παραδειγμάτων που έχουν την τιμή v_i για το A
- Αν Παραδείγματα(v_i) είναι κενό
- Τότε κάτω από αυτό τον νέο κλάδο πρόσθεσε έναν κόμβο-φύλλο με ετικέτα = την πιο κοινή τιμή στόχο στα παραδείγματα
- Αλλιώς κάτω από αυτό τον νέο κλάδο πρόσθεσε το υποδέντρο ID3

- Τέλος
- Επέστρεψε Ρίζα

Αλγοριθμος C4.5

Είναι ένας αλγόριθμος που χρησιμοποιείται για να παραγάγει ένα δέντρο απόφασης και αποτελεί μια επέκταση του προηγούμενου αλγορίθμου ID3. Ο C4.5 δημιουργεί δέντρα απόφασης από ένα σύνολο δεδομένων εκπαίδευσης, όμοια με τον αλγόριθμο ID3, χρησιμοποιώντας την έννοια της εντροπίας πληροφοριών. Τα δεδομένα εκπαίδευσης είναι ένα σύνολο $S = s_1, s_2, \ldots$ από ήδη ταξινομημένα δείγματα. Κάθε δείγμα $s_i = x_1, x_2, \ldots, x_p$ είναι ένα διάνυσμα όπου x_j αντιπροσωπεύει τις ιδιότητες ή τα χαρακτηριστικά γνωρίσματα του δείγματος. Επίσης, στα δεδομένα εκπαίδευσης αντιστοιχεί ένα διάνυσμα $C = c_1, c_2, \ldots$όπου $c_1, c_2, \ldots$ αντιπροσωπεύει την κατηγορία στην οποία ανήκει κάθε δείγμα. Ο C4.5 χρησιμοποιεί το γεγονός ότι κάθε χαρακτηριστικό των δεδομένων μπορεί να χρησιμοποιηθεί για να λάβει μια απόφαση, η οποία χωρίζει τα δεδομένα σε μικρότερα υποσύνολα. Ο C4.5 εξετάζει το ομαλοποιημένο κέρδος πληροφοριών (information gain -διαφορά στην εντροπία) που προκύπτει από την επιλογή ενός χαρακτηριστικού για το διαχωρισμό των δεδομένων. Το χαρακτηριστικό με το υψηλότερο ομαλοποιημένο κέρδος πληροφοριών είναι αυτό που χρησιμοποιείται για να ληφθεί μια απόφαση. Ο αλγόριθμος επαναλαμβάνεται για μικρότερες υπολίστες δεδομένων. Πιο κάτω παρουσιάζεται ο ψευδοκώδικας του αλγόριθμου C4.5

1. Για κάθε χαρακτηριστικό A.
2. Βρίσκουμε το ομαλοποιημένο κέρδος πληροφοριών από το διαχωρισμό στο A.

3. Έστω ότι A_best είναι το χαρακτηριστικό με το υψηλότερο ομαλοποιημένο κέρδος πληροφοριών
4. Δημιούργησε έναν κόμβων απόφασης Node που χωρίζεται στο A_best
5. Επανερχόμαστε στις υπολίστες που λαμβάνονται με το διαχωρισμό στο A_best και προσθέτουμε αυτούς τους κόμβους ως παιδιά του Node

Random Forests

Τα παρακάτω βήματα, δίνουν μία γενική ιδέα για το πώς λειτουργούν τα random forests:

1. Αρχικά, αναπτύσσονται πολλά classification decision trees
2. Κάθε tree δίνει μία ταξινόμηση – «Το δέντρο ψηφίζει αυτήν την κλάση».
3. Έτσι, κάθε κλάση έχει έναν αριθμό «ψήφων» (votes).
4. Η τελική και οριστική ταξινόμηση γίνεται με το «δάσος» να διαλέγει την κλάση με τις περισσότερες votes

2.3.4 Νευρωτικά Δίκτυα

Μια άλλη προσέγγιση της κατηγοριοποίησης που χρησημοποιείται σε πολλές εφαρμογές εξόρυξης γνώσης για πρόβλεψη και κατηγοριοποίηση βασίζεται στα νευρωτικά δίκτυα.Οι μέθοδοι αυτής της προσέγγισης χρησιμοποιούν τα νευρωτικά δίκτυα για να κατασκευάσουν ένα μοντέλο κατηγοριοποίησης ή πρόβλεψης.Τα κύρια βήματα αυτλης της διαδικασίας είναι:

- Αναγνώριση των χαρακτηριστικών εισόδου και εξόδου.
- Κατασκευή ενός δικτύου με την κατάλληλη τοπολογία.
- Επιλογή του σωστού συνόλου εκπαίδευσης.
- Εκπαίδευση του δικτύου με βάση ένα αντιπροσωπευτικό σύνολο δεδομένων.Τα δεδομένα πρέπει να απεικονίζονται με τέτοιον τρόπο ώστε να μεγιστοποιηθεί η δυνατότητα του δικτύου να αναγνωρίζει πρότυπα.
- Έλεγχος του δικτύου χρησιμοποιώντας ένα σύνολο ελέγχου το οποίο είναι ανεξάρτητο από το σύνολο εκπαίδευσης.

Κατόπιν το μοντέλο που παράγεται από το δίκτυο,εφαρμόζεται για να προβλέψει τις κατηγορίες (έξοδοι-outputs) των μη κατηγοριοποιημένων δειγμάτων (είσοδοι-inputs).

2.3.5 Κατηγοριοποίηση με βάση τα Νευρωτικά Δίκτυα

Τα νευρωτικά δικτύα αποτελούνται από "νευρώνες" με βάση τη νευρωτική δομή του εγκεφάλου.Επεξεργάζονται τα στοιχεία ένα κάθε φορά και μαθαίνουν συγκρίνοντας την κατηγοριοποίηση τους για μια εγγραφή (που στην έναρξη είναι κατά ένα μεγάλο μέρος αυθαίρετη) με τηη γνωστή πραγματική κατηγοριοποίηση της εγγραφής.Τα λάθη από την αρχική κατηγοριοποίηση της πρώτης εγγραφής επανατροφοδοτούνται στο δίκτυο,και χρησημοποιούνται για να τροποποιήσουν τον αλγόριθμο δικτύων τη δεύτερη φορά.Η διαδικασία αυτή συνεχίζεται επαναλήπτικά. Γενικά,ένας νευρώνας σε ένα τεχνηκό νευρωτικό δίκτυο είναι:

1. Ένα σύνολο εισερχομένων τιμών (x_i) και συσχετιζόμενων βαρών (w_i).
2. Μια συνάρτηση (g) που αθροίζει τα βάρη και αντιστοιχεί τα αποτελέσματα σε μια έξοδο (y).

Οι νευρώνες οργανώνονται σε επίπεδα.Το επίπεδο εισαγωγής αποτελείται όχι απο τους πλήρεις νευρ΄ψνες,άλλα μάλλον συνίσταται απλά από τιμές ενός στοιχείου του συνόλου δεδομένων,οι οποίες αποτελούν τις εισαγωγές στο επόμενο επίπεδο των νευρώνων.Το επόμενο επίπεδο καλείται κρυμμένο επίπεδο.Μπορούν να υπάρξουν διάφορα κρυμμένα επίπεδα.Το τελέυταίο επίπεδο είναι η έξοδος ,όπου υπάρχει ένας κόμβος για κάθε κατηγορία.Μια μόνο σάρωση προς τα εμπρός του δικτύου οδηγεί σετην ανάθεση μια τιμές σε κάθε κόμβο εξόδου,και η εγγραφή ανατίθεται στον κόμβο της κατηγορίας που είχε την υψηλότερη τιμή.

2.3.6 Κατηγοριοποίηση με βάση την τεχνική των Κοντινότερων Γειτόνων

Η τεχνική των κοντινότερων γειτόνων (Nearest Neighbor-NN) είναι μια απλή προσέγγιση του προβλήματος της κατηγοριοποίησης η οποία συγκεντρώνει αρκετό ενδιαφέρον. Σύμφωνα με τη μέθοδο αυτή ένα νέο στοιχείο κατηγιοποιείται χρησιμοποιώντας την πλειοψηφία μεταξύ των κατηγοριών από τα K παραδείγματα που είναι τα πιο κοντινά σ'αυτό που δίνεται να κατηγοριοποιηθεί.Μία τέτοια μέθοδος παράγει συνεχείς και επικαλυπτόμενες, παρά σταθερές,γειτονιές.Επιπρόσθετα έχει αποδειχτεί ότι ένας NN κανόνας έχει αδυμπτωτικό ποσοστό σφάλματος που είναι δύο φορές το ποσοστό σφάλματος Bayes,ανεξάρτητα από το μέτρο απόστασης που χρησιμοποείται. Η τεχνική NN έχει μειονεκτήματα σε χώρους υψηλών

διαστάσεων.Η αυστηρή πόλωση μπορεί να εισαχθεί στην τεχνική NN όταν υπάρχει ένας πεπερασμένος αριθμός από τα παραδείγματα σε χώρο υψηλών διαστάσεων .Για να βελτιωθεί η αποδοσή του έχουν προταθεί διαφορετικές προσεγγίσεις για να προσαρμοστούν τοπικά το μέτρο απόστασης που χρησιμοποιείται από τον NN κανόνα.

2.3.7 Support Vector Machines

Οι Support Vector Machines (SVMs) έχουν εφαρμοστεί πρόσφατα με ιδιαίτερη επιτυχία σε ποικίλες εφαρμογές.Σε αυτό το σημείο παρουσιάζω τις βασικές έννοιες και ιδιότητες των SVMs.Έστω ότι έχουμε n παρατηρήσεις.Κάθε παρατήρηση αποτελείται από ένα ζευγάρι της μορφής:διάνυσμα $x_i \in R^n$,και μια ετικέτα(label) της συσχετιζόμενης κατηγορίας y_j.Υποτίθεται ότι υπάρχει κάποια άγνωστη κατανομή πιθανότητας $P(x, y)$ με βάση την οποία τα στοιχεία παράγονται.Ο σκοπός είναι να μαθευτεί το σύνολο παραμέτρων α της συνάρτησης $f(x, \alpha)$ έτσι ώστε η f να πραγματοποιεί την αντιστοίχηση $x_i \rightarrow y_i$.Μια συγκεκριμένη επιλογή του α καθορίζει μοναδικά την αντίστοιχη μηχανή εκαπίδευσης,$f(x, \alpha)$. Αντίθετα από τις παρασοδιακές μεθόδους ,οι οποίες ελαχιστοποιούν τον εμπειρικό κίνδυνο,μια SVM στοχεύει στην αλαχιστοποίηση του ανωτέρου ορίου σφάλμαστος γενίκευσης.Επιτυγχάνει αυτόν το στόχο με την εκμάθηση του α της $f(x, \alpha)$ έτσι ώστε η μηχανή εκπαίδευσης να ικαναποιεί την ιδιότητα του μεγίστου περιθωρίου ,δηλαδή το όριο απόφασης που αντιπροσωπεύει,έχει τη μέγιστη-ελάχιστη απόσταση από το πιο κοντινό σημείο εκπαίδευσης. Το αναμενόμενο σφάλμα ελέχγου για μια εκαπιδευμένη μηχανή είναι $R(\alpha) = (\frac{1}{2} \int \mid y - f(z, \alpha) \mid \ \mathrm{dP}x, y)$. Η ποσότητα R(α) καλείται αναμενόμενος κίνδυνος.Παρότι δίνει έναν καλό τρόπο γραψίματος του σφάλματος πραγματικού μέσου όρου ,εάν δεν έχουμε εκτίμηση του $P(x, y)$ δεν είναι χρήσιμος.Στη συνέχεια,ο εμπειρικός κίνδυνος $R_{emp}(\alpha)$ ορίζεται ως το ποσοστό σφάλματος που μετριέται πάνω στο σύνολο εκπαίδευσης: $R_{emp}(\alpha) = (\frac{1}{2} \sum_{i=1}^{n} \mid y - f(x_i, \alpha) \mid$. Μια SVM μηχανή αντιστοιχεί τα δεδομένα σε έναν χώρο υψηλών διαστάσεων (χώρο των χαρακτηριστικών) και καθορίζει ένα διαχωριστικό υπερεπίπεδο εκεί.Η μετάφραση του συνόλου εκπαίδευσης σε έναν υψηλών χώρο διαστάσεων επιφέρει τόσο υπολογιστικό όσο και θεωρητικόμάθησης κόστος.Οι SVMs αποφεύγουν την υπερκάλυψη διαλέγοντας ένα συγκεκριμένο υπερ-επίπεδο ανάμεσα σε πολλά μου μπορούν να διαχωρίσουν τα δεδομένα στο χώρο των χαρακτηριστικών, συγκεκριμένα το μέγιστο υπερ-επίπεδο περιθωρίου. Το υπολογιστικό φορτίο του να αντιπροσωπεύουμε ρητά τα διανύσματα χαρακτηριστικών γνωρισμάτων αποφεύγεται με τον καθορισμό μιας συνάρτησης,

που ονομάζεται συνάρτηση πυρήνα.Επομένως, μια SVM μπορεί να εντοπίσει ένα διαχωριστικό επίπεδο σε ένα διάστημα χαρακτηριστικών γνωρισμάτων και να κατηγοριοποιήσει τα σημεία σε εκείνο το διάστημα χωρίς πάντα να αντιπροσωπεύει το διάστημα αυτό μοναδικά. Επιπρόσθετα για να αποφύγουμε την υπερκάλυψη ,η χρήση του μεγίστου επιπέδου περιθωρίου οδηγεί σε έναν αλγόριθμο εκπαίδευσης που μπορεί να μειωθεί σε ένα πρόβλημα βελτιστοποίησης.Προκειμένου να εκπαιδευτεί η SVM πρέπει να βρεθεί το μοναδικό ελάχιστο μια συνάρτησης .Κατά συνέπεια, οι SVMs δεν εχουν τοπικό πρόβλημα ελαχίστων που έχει επιπτώσεις σε πολλά σχήματα εκπαίδευσης και αντίθετα από τον αλγόριθμο ”back propagation learning” για τα νευρωτικά δίκτυα,μια δοσμένη SVM θα συγκλίνει πάντα ντετερμινιστικλα στην ίδια λύση για δεδομένο σύνολο δεδομένων,ανεξάρτητα από τις αρχικές υποθέσεις. Στην απλή περίπτωση των δύο γραμμικά διακριτών κατηγοριών,μια SVM επιλέγει,μεταξύ του άπειρου αριθμού γραμμικών κατηγοριοποιητών που διαχωρίζουν τα δεδομένα,εκείνον τον κατηγοριοποιητή που ελαχιστοποιεί το ανώτερο όριο του σφάλματος γενίκευσης.Μια SVM επιτυγχάνει αυτόν αυτόν το στόχο με τον υπολογισμό του κατηγοριοποιητή που ικανοποιεί την ιδιότητα μέγιστου περιθωρίου, δηλαδή τον κατηγοριοποιητή του οποίου η απόφαση ορίου έχει την μέγιστη-ελάχιστη απόσταση από το πιο κοντινό σημείο εκπαίδευσης. Εάν οι δυο κατηγορίες είναι μη διακριτές , η SVM ψάχνει ένα επίπεδο που μεγιστοποιεί το περιθώριο κια που ταυτόχρονα ελαχιστοποιεί μια ποσότητα ανάλογη του αριθμού των σφαλμάτων κατηγοριοποίησης.

2.4 Αλγόριθμοι Συσταδοποίησης

2.4.1 Εισαγωγή

Η συσταδοποίηση (clustering) είναι μια από τις πιο χρήσιμες διεργασίες στη διαδικασία εξόρυξης γνώσης για την ανακάλυψη συστάδων και για τον προσδιορισμό κατανομών ή προτύπων (patterns) που παρουσιάζουν ενδιαφέρον στα υπό μελέτη δεδομένα. Το πρόβλημα της συσταδοποίησης σχετίζεται με την τμηματοποίηση ενός συνόλου δεδομένων σε συστάδες έτσι ώστε τα στοιχεία του συνόλου των δεδομένων που ανήκουν σε μια συστάδα να είναι περισσότερο όμοια μεταξύ τους από ότι είναι με τα στοιχεία των άλλων συστάδων. Το βασικό μέλημα της διαδικασίας συσταδοποίησης είναι να αποκαλύψει την οργάνωση προτύπων σε ”λογικές” συστάδες , οι οποίες θα μας επιτρέψουν να ανακαλύψουμε ομοιότητες και διαφορές ,καθώς επίσης και να αποκομίσουμε χρήσιμα συμπεράσματα γι’αυτά. Η συσταδοποίηση μπορεί να βρεθεί σε διάφορα πεδία όπως στην αναγνώριση προτύπων,στη βιολογία,στην οικολογλια,στις κοινωνικές επιστήμες και στη θεωρία

των γράφων. Στη διαδικασία της συσταδοποίησης δεν υπάρχουν προκαθορισμένες κατηγορίες ούτε κάποιο παράδειγμα που θα έδειχνε ποιες επιθυμητές σχέσεις ήταν έγκυρες μεταξύ των δεδομένων.Από την άλλη η κατηγοριοποίηση είναι μια διαδικασία ανάθεσης ενός αντικειμένου από το σύνολο των δεδομένων σε μια προκαθορισμένη κατηγορία. Η διαδικασία συσταδοποίησης μπορεί να οδηγήσει σε διαφορετικές τμηματοποιήσεις ενός συνόλου δεδομένων ανάλογα με το κριτήριο που χρησιμοποιείται για τη συσταδοποίηση.Κατα συνέπεια υπάρχει ανάγκη προ-επεξεργασιας των δεδομένων προτού εφαρμοστεί η διεργασία της συσταδοποίησης σε ένα σύνολο δεδομένων.

2.4.2 Διαιρετικοί Αλγόριθμοι

Αλγοριθμος k-means

Η μεθοδος k-means αποτελει μια απο τισ πιο συχνα χρησημοποιουμενες μεθοδους συσταδοποιησης (MacQueen'67). Ανηκει στην κατηγορια της διαιρετικης συσταδοποιησης (partiotional clustering) καθως βασιζεται στην αμεση αποσυνθεση του συνολου των δεδομενων σε ενα συνολο ασυσχετιστων συσταδων. Η αντικειμενικη συναρτηση την οποια προσπαθει να ελαχιστοποιησει ο αλγοριθμος ειναι η μεση τετραγωνικη αποσταση των δεδομενων απο τα πλησιεστρερα κεντρα των συσταδων και δινεται απο την εξισωση:

$$E = \sum_{i=1}^{c} \sum_{x \in C_i} d(x, m_i)$$

Στην παραπάνω εξίσωση ,m_i είναι το κέντρο της συστάδας C_i,ενώ $d(x, m_i)$ είναι η Ευκλείδεια απόσταση μεταξύ ενός στοιχείου x και του κέντρου m_i. Κατά συνέπεια το κριτήριο - συνάρτηση E προσπαθεί να ελαχιστοποιήσει την απόσταση κάθε σημείου από το κέντρο της συστάδας στο οποίο το σημείο ανήκει. Πιο συγκεκριμένα ο αλγόριθμος ξεκινά με την αρχικοποίηση των κέντρων ή των συστάδων. Κατόπιν αναθέτει κάθε στοιχείο (αντικείμενο) του συνόλου δεδομένων στη συστάδα της οποίας το κέντρο είναι πιο κοντά και ξανά-υπολογίζει τα κέντρα. Η διαδικασία συνεχίζει έως ότου τα κέντρα των συστάδων σταματήσουν να αλλάζουν.
Ο βασικός αλγόριθμος για να ελαχιστοποιήσει την αντικειμενική συνάρτηση αρχίζει θεωρώντας ένα σύνολο από k σημεία σαν κέντρα των k συστάδων. Αν η σειρά των δεδομένων δεν έχει κάποια ιδιαίτερη σημασία, τότε παίρνουμε τις πρώτες k εγγραφές. Αλλιώς επιλέγουμε σημεία αντιπροσωπευτικά για τις θεωρούμενες συστάδες. Καθένα από τα κέντρα αντιπροσωπεύει μια συστάδα. Στο δεύτερο

βήμα,κάθε στοιχείο αντιστοιχείται στη συστάδα της οποίας το κέντρο βρίσκεται πιο κοντά.
Στη συνέχεια υπολογίζονται τα νέα κέντρα των συστάδων με χρήση του μέσο όρου των σημείων τους. Για άλλη μια φορά αντιστοιχείται κάθε σημείο στη συστάδα της οποίας το κέντρο είναι πιο κοντά. Η διαδικασία επαναλαμβάνεται συνεχώς έως ότου τα όρια των συστάδων παύουν να μεταβάλλονται, ή συνάρτηση E δεν μεταβάλλεται σημαντικά.
Ο αλγόριθμος k-means χρησιμοποιεί σταθερό και δεδομένο εξαρχής αριθμό συστάδων που θα δημιουργηθούν (όσα και τα κέντρα).
Στην παράγραφο αυτή περιγράφω την μορφή ψευδοκώδικα τα βασικά βήματα του αλγορίθμου k-means. Ο αλγόριθμος ξεκινά καθορίζοντας με τυχαίο τρόπο c κέντρα που θα αντιπροσωπεύουν τις c συστάδες. Στη συνέχεια προσδιορίζεται η απόσταση κάθε στοιχείου του σύνολου δεδομένων από το κέντρο κάθε συστάδας και κάθε στοιχείο τοποθετείται στη συστάδα από την οποία απέχει λιγότερο. Τα κέντρα των νέων συστάδων υπολογίζονται σαν ο μέσος όρος των στοιχείων που ανήκουν μέχρι στιγμής σε κάθε συστάδα. Η διαδικασία επαναλαμβάνεται μέχρις ότου οι συστάδες να σταματήσουν να μεταβάλλονται. Αυτό σημαίνει ότι η απόκλιση μεταξύ των κέντρων των συστάδων που προέκυψαν τελευταία από αυτά της προηγούμενης επανάληψης είναι κοντά στο μηδέν (τα κέντρα ταυτίζονται).
Τα βήματα του αλγορίθμου σε μορφή ψευδοκωδικα είναι τα εξής:

1. Εύρεση των αρχικών κέντρων,$v_i, i = 1, 2, .., c$, για τις c συστάδες. Για κάθε επανάληψη $r = 1, ..., r_{max}$:

2. Υπολογισμός της απόστασης κάθε στοιχείου του συνόλου δεδομένων από το κέντρο κάθε συστάδας

$$d_{ki} = (x_k - v_i)^2, k = 1, 2, ..., n \; i = 1, 2, ..., c$$

3. Κάθε στοιχείο x_k αντιστοιχίζεται για την συστάδα για την οποία ισχύει

$$min_{k,i}(d_{ik}), \; \forall \; i, k$$

4. Υπολογισμός των νέων κέντρων των συστάδων

$$m_i^{(r+1)} = \frac{\sum_{i=1}^{n_i} x_k}{n_i}$$

όπου n_i ο αριθμός των στοιχείων που ανήκουν στην i συστάδα μέχρι στιγμής.

5. if $\|m_i^{(r)} - m_i^{(r+1)}\| < \epsilon$ then
   stop
   else
   $r = r + 1, goto$ 2.

PAM(Partitioning Around Medoids)

Ο PAM αναπτύχθηκε από τους Kaufman και Rousseeuw και αποτελεί μια από τις γνώστες k-medoids μεθόδους συσταδοποίησης.
Προκείμενου να βρεθούν k συστάδες με την προσέγγιση PAM καθορίζεται ένα αντικείμενο-αντιπρόσωπος για κάθε συστάδα. Τα αντικείμενα-αντιπρόσωποι καλούνται medoids και είναι τα αντικείμενα εκείνα που βρίσκονται πιο κοντά στα κέντρα των συστάδων. Μόλις επιλέγουν τα medoids κάθε μη επιλεγμένο αντικείμενο ομαδοποιείται στη συστάδα του medoid με το οποίο μοιάζει περισσότερο. Ειδικότερα, εάν O_j είναι ένα μη επιλεγμένο αντικείμενο και O_i είναι ένα επιλεγμένο medoid λέμε ότι το O_j ανήκει στην συστάδα που αντιπροσωπεύεται από το O_i εάν $(_j, O_i) = min_{O_c} d(O_j, O_c)$ όπου η έκφραση min_{O_c} δηλώνει το ελάχιστο μεταξύ όλων των medoids O_c και το $d(O_a, O_b)$,δηλώνει την απόσταση μεταξύ όλων των αντικειμένων O_a, O_b. Όλες οι τιμές των αποστάσεων μεταξύ των αντικειμένων δίνονται ως είσοδος στο PAM. Η ποιότητα της συστηματοποίησης μετράται με βάση τη μέση απόσταση ενός αντικειμένου από το medoid της συστάδας που ανήκει.

Περιγραφή αλγόριθμου

Έστω ότι έχουμε ένα σύνολο n αντικειμένων το οποίο θέλουμε να τμηματοποιήσουμε σε k συστάδες με βάση τον αλγόριθμο PAM. Ο PAM ξεκινά με την εύρεση των k medoids, επιλέγοντας με τυχαίο τρόπο k αντικείμενα. Στη συνέχεια σε κάθε βήμα, εκτελείται μια ανταλλαγή ανάμεσα σε ένα επιλεγμένο αντικείμενο O_i και ένα μη επιλεγμένο O_h αντικείμενο μέχρις ότου η ανταλλαγή αυτή να οδηγήσει στην βελτίωση της ποιότητας της συσταδοποίησης. Ειδικότερα, για να υπολογίσουμε το αποτέλεσμα μια τέτοιας ανταλλαγής ανάμεσα στα αντικείμενα O_i και O_h, ο PAM υπολογίζει το κόστος C_{jih} για όλα τα επιλεγμένα αντικείμενα O_j. Ο ορισμός του κόστους γίνεται σύμφωνα με τις τέσσερις ακόλουθες εκφράσεις ανάλογά τις περιπτώσεις των αντικειμένων O_j:

1. Το O_j ανήκει στην συστάδα που αντιπροσωπεύεται από το O_i. Επιπρόσθετα, ας υποθέσουμε ότι το O_j είναι πιο κοντά στο αντικείμενο $O_{j,2}$ από ότι στο αντικείμενο O_h , δηλαδή $d(O_j, O_h) \geq d(O_j, O_{j,2})$,όπου το $O_{j,2}$ είναι το δεύτερο medoid που είναι πιο κοντά στο O_j. Έτσι αν αντικαταστήσουμε το O_i με το O_h σαν medoid, το O_j θα ανήκει στη συστάδα που αντιπροσωπεύεται από το $O_{j,2}$. Το κόστος της ανταλλαγής θα δίνεται από την εξίσωση:

$$C_{jih} = d(O_j, O_{j,2}) - d(O_j, O_i)$$

Η ισότητα αυτή δίνει πάντα μη αρνητική τιμή για το κόστος, υποδηλώνοντας ότι το κόστος που προκύπτει για την αντικατάσταση του O_i με το O_h δεν είναι αρνητικό.

2. Το O_j ανήκει στην συστάδα που αντιπροσωπεύεται από το O_i. Αλλά αυτή τη φορά, το O_j είναι λιγότερο κοντά στο αντικείμενο $O_{j,2}$ σε σχέση με το αντικείμενο O_h δηλαδή $d(O_j, O_h) \leq d(O_j, O_{j,2})$. Έτσι εάν αντικαταστήσουμε το O_i με το O_h σαν medoid, το O_j θα ανήκει στη συστάδα που αντιπροσωπεύεται από το O_h. Το κόστος της ανταλλαγής θα δίνεται από την εξίσωση:

$$C_{jih} = d(O_j, O_h) - d(O_j, O_i)$$

Η τιμή του κόστους μπορεί να είναι αρνητική ή και θετική ανάλογα με το αν το αντικείμενο O_j προσεγγίζει περισσότερο το O_i ή το O_h.

3. Υποθέτουμε ότι το O_j ανήκει σε μια συστάδα διαφορετική από αυτή που αντιπροσωπεύεται από το O_i. Έστω ότι το $O_{j,2}$ είναι ο αντιπρόσωπος της συστάδας. Επίσης, θεωρούμε ότι το O_j προσεγγίζει περισσότερο το O_i από το O_h. Έτσι εάν αντικαταστήσουμε το O_i με το O_h, το O_j θα παραμείνει στη συστάδα που αντιπροσωπεύεται από το $O_{j,2}$. Το κόστος της ανταλλαγής θα δίνεται από την εξίσωση:

$$C_{jih} = 0$$

4. Το O_j ανήκει στην συστάδα που αντιπροσωπεύεται από το $O_{j,2}$. Αλλά το O_j είναι λιγότερο κοντά στο αντικείμενο $O_{j,2}$ από ότι με το αντικείμενο O_h. Εάν αντικαταστήσουμε το O_i με το O_h θεωρώντας σαν νέο medoid, το O_j θα μετακινηθεί στη συστάδα που αντιπροσωπεύεται από το O_h. Το κόστος της ανταλλαγής θα δίνεται από την εξίσωση:

$$C_{jih} = d(O_j, O_h) - d(O_j, O_{j,2})$$

Το κόστος στην περίπτωση αυτή θα είναι πάντοτε αρνητικό.

Συνδυάζοντας τις τέσσερις παραπάνω περιπτώσεις το συνολικό κόστος αντικατάστασης του αντικείμενου O_i με το O_hδίνεται από την εξίσωση:

$$TC_{ih} = \sum_j C_{ijh}$$

Βήματα αλγορίθμου PAM

Τα βασικά βήματα του PAM σε μορφή ψευδοκώδικα είναι:

1. Τυχαία επιλογή k αντιπρόσωπων για τις συστάδες.
2. Υπολογισμός του συνολικού κόστους TC_{ih} για όλα τα ζεύγη των αντικειμενων O_i, O_h όπου το O_i είναι το τρέχον επιλεγμένο αντικείμενο και το O_h είναι ένα μη επιλεγμένο αντικείμενο.
3. Επιλέγουμε το ζεύγος O_i,O_h το οποίο αντιστοιχεί στο $min_{Oi,Oh}TC_{ih}$.Εάν το συνολικό κόστος είναι αρνητικό αντικαθιστούμε το O_i με το O_h και επιστρέφουμε στο βήμα 2.
4. Διαφορετικά, για κάθε μη επιλεγμένο αντικείμενο, βρίσκουμε το αντικείμενο αντιπρόσωπο που προσεγγίζει περισσότερο.

Ο PAM δουλεύει ικανοποιητικά για μικρά σύνολα δεδομένων, ενώ αποδεικνύεται μη αποδοτικός για σύνολα δεδομένων μεσαίου και μεγάλου μεγέθους λόγω τη μεγάλης πολυπλοκότητας του. Στο βήμα 2 και 3 υπάρχουν συνολικά $k(n-k)$ ζεύγη αντικείμενων O_i,O_h, όπου n είναι ο αριθμός των στοιχείων του σύνολου δεδομένων και k είναι ο αριθμός των συστάδων. Για κάθε ζεύγος ο υπολογισμός του συνολικοί κόστους απαιτεί την εξέταση (n-k) μη επιλεγμένων αντικείμενων. Συνεπώς τα βήματα 2 και 3 έχουν για μια επανάληψη πολυπλοκότητα $O(k(n-k)^2)$. Είναι λοιπόν φανερό ότι ο PAM έχει μεγάλο κόστος για μεγάλες τιμές του n και k.

CLARA(Clustering Large Applications)

Ο αλγόριθμος CLARA αναπτύχθηκε από τους Kaufman και Rousseeuw προκείμενου να διαχειριστούν μεγάλα σύνολα δεδομένων. Η βασική διαφορά ανάμεσα στον CLARA και τον PAM είναι ότι ο πρώτος βασίζεται στην δειγματοποίηση (sampling). Ο CLARA αντίθετα με τον PAM δεν βρίσκει αντικείμενα αντιπροσώπους για ολόκληρο το σύνολο δεδομένων, αλλά λαμβάνει με τυχαίο τρόπο ένα δείγμα του συνόλου των δεδομένων , εφαρμόζει στο δείγμα τον PAM και βρίσκει τα medoids του δείγματος. Η ιδέα είναι ότι εάν το δείγμα είναι σχεδιασμένο με εντελώς τυχαίο τρόπο, τότε αναπαριστά ολόκληρο το σύνολο ικανοποιητικά και για το λόγο αυτόν τα αντικείμενα αντιπρόσωποι (medoids) του δείγματος θα προσεγγίζουν τα medoids ολοκλήρου του σύνολου δεδομένων. Ο αλγόριθμος σχεδιάζει πολλαπλά δείγματα και εξάγει την καλύτερη συσταδοποίηση από τα δείγματα αυτά.Τα πειράματα έχουν αποδείξει ότι πέντε δείγματα μεγέθους 40+2k δίνουν ικανοποιητικά αποτελέσματα.

Αλγοριθμος CLARA

Στην συνεχεία αναφέρονται υπό μορφή ψευδοκώδικα τα βασικά βήματα του αλγορίθμου:

1. Για i=1 ... 5 επαναλαμβάνουμε τα ακόλουθα βήματα:
2. Σχεδιάζουμε ένα δείγμα 40+2k αντικειμένων με τυχαίο τρόπο από το σύνολο των δεδομένων και εφαρμόζουμε τον αλγόριθμο PAM για να βρούμε τους k αντιπρόσωπους για τις συστάδες.
3. Για κάθε αντικείμενο O_j στο σύνολο δεδομένων καθορίζουμε ποιο από όλα τα k medoids προσεγγίζει περισσότερο το O_i.
4. Υπολογίζουμε τη συνολική ανομοιότητα για τη συσταδοποίηση που λαμβάνεται από το προηγούμενο βήμα. Εάν αυτή η τιμή είναι μικρότερη από το τρέχον ελάχιστο, χρησιμοποιούμε αυτή την τιμή του ελάχιστου σαν τρέχον ελάχιστο και διατηρούμε τα k medoids που βρήκαμε στο βήμα 2 σαν το καλύτερο σύνολο medoids που έχουμε μέχρι στιγμής.
5. Επιστρέφουμε στο βήμα 1 και ξεκινάμε την επόμενη επανάληψη.

Ο CLARA εφαρμόζει τον PAM μόνο σε δείγματα και έτσι σε κάθε επανάληψη η πολυπλοκότητα είναι $O(k(40+k)^2 + k(n-k))$. Συνεπώς ο CLARA είναι πιο αποδοτικός από τον PAM για μεγάλες τιμές του n (αριθμός στοιχείων σύνολου δεδομένων).

CLARANS (Clustering Large Applications Based on Randomized Search)

Ο αλγόριθμος CLARANS προσπαθεί να συνδυάσει τους αλγορίθμους PAM και CLARA εκτελώντας κάθε φορά αναζήτηση μόνο σε ένα υποσύνολο του σύνολου των δεδομένων ενώ δεν περιορίζεται σε κάποιο δείγμα σε μια δεδομένη στιγμή. Άνω ο CLARA έχει ένα καθορισμένο δείγμα σε κάθε βήμα της αναζήτησης,ο CLARANS σχεδιάζει ένα δείγμα με τυχαίο τρόπο σε κάθε βήμα της αναζήτησης. Η διαδικασία συσταδοποίησης μπορεί να αναπαρασταθεί σαν ένα γράφημα όπου κάθε κόμβος είναι μια πιθανή λύση δηλαδή ένα σύνολο από k medoids. Η συσταδοποιηση που λαμβάνεται μετά την αντικατάσταση ενός medoid καλείται *γείτονας(neighbor)* της τρέχουσας συσταδοποίησης. Ο αριθμός των γειτόνων που μπορούν να εξεταστούν ως medoids πιθανής λύσης καθορίζεται από μια παράμετρο που καλείται *maxneighbor*. Εάν βρεθεί ένας καλύτερος γείτονας ο CLARANS μετακινείται στον κόμβο του γείτονα και η διαδικασία ξεκινάει πάλι από τον κόμβο αυτό,ενώ σε διαφορετική περίπτωση η τρέχουσα συσταδοποίηση παράγει ένα τοπικό βέλτιστο.
Εάν βρεθεί ένα τοπικό βέλτιστο, ο αλγόριθμος CLARANS αρχίζει με έναν νέο τυχαία επιλεγμένο κόμβο για την αναζήτηση ενός νέου τοπικού βέλτιστου.Ο αριθμός των τοπικών βέλτιστων που θα αναζητηθούν καθορίζεται επίσης από μια παράμετρο που καλείται *numlocal*. Ο αλγόριθμος αυτός έχει αποδειχθεί πιο αποδοτικός σε σχέση με τον CLARA και τον PAM και η υπολογιστική πολυπλοκότητα του για κάθε επανάληψη εξαρτάται από τον αριθμό των αντικείμενων,$O(n^2)$. Ωστόσο,λόγω της τυχαίας προσέγγισης του CLARANS, για μεγάλες τιμές του ν,η ποιότητα των αποτελεσμάτων δεν είναι εγγυημένη.

Βήματα αλγορίθμου CLARANS

Τα βασικά βήματα του αλγόριθμου μπορούν να συνοψιστούν στα εξής:

1. Αρχικοποίηση των παραμέτρων numlocal και maxneighbor. Αρχικοποιούμε το i σε 1 και θέτουμε ως ελάχιστο κόστος mincost έναν μεγάλο αριθμό.

2. Καθορισμός της μεταβλητής current (τρέχον κόμβος προς εξέταση) ώστε να αναφέρεται σε έναν αρχικό κόμβο $G_{n,k}$.

3. Θέτουμε το j ίσο με 1.

4. Θεωρούμε ένα τυχαίο γείτονα S του τρέχοντος και υπολογίζουμε το κόστος αντικατάστασης του τρέχοντος κόμβου από τον γειτονικό κόμβο.

5. Εάν ο S έχει μικρότερο κόστος,θέτουμε ως τρέχων κόμβο (current) τον S και επιστρέφουμε στο βήμα 3.

6. Διαφορετικά,αυξάνουμε το j κατά 1.Εάν $j \leq maxneighbor$, επιστρέφουμε στο βήμα 4.

7. Διαφορετικά, όταν το $j > maxneighbor$,συγκρίνουμε το κόστος του τρέχοντος κόμβου current με το ελάχιστο κόστος mincost. Εάν το πρώτο είναι μικρότερο από το mincost, θέτουμε ως mincost το κόστος του currrent και ορίζουμε ως καλύτερο κόμβο (bestnode) τον current.

8. Αυξάνουμε το i κατά 1. Εάν $i > numlocal$ εξάγουμε τον καλύτερο κόμβο και η διαδικασία σταμάτα. Διαφορετικά, επιστρέφουμε στο βήμα 2.

Άσο μεγαλύτερος είναι ο αριθμός των γειτόνων (macneighbor) που εξετάζονται τόσο ο αλγόριθμος CLARANS προσεγγίζει τον PAM και η αναζήτηση για την εύρεση του τοπικού ελάχιστου έχει μεγαλύτερη διάρκεια. Αλλά η ποιότητα ενός τέτοιου ελάχιστου είναι μεγαλύτερη και έτσι λιγότερα τοπικά ελάχιστα χρειάζονται να εξεταστούν.

2.4.3 Ιεραρχικοί αλγόριθμοι

Οι ιεραρχικοί αλγόριθμοι συσταδοποίησης σύμφωνα με την μέθοδο που παράγουν τις συστάδες μπορούν να περαιτέρω να διαιρεθούν σε:

- *Συσσωρευτικούς (Agglomerative) Ιεραρχικούς Αλγορίθμους*. Οι αλγόριθμοι αυτοί παράγουν μια ακολουθία σχημάτων συσταδοποίησης μειώνοντας τον αριθμό συστάδων σε κάθε βήμα.Το σχέδιο συσταδοποίησης που παράγεται σε κάθε βήμα οδηγεί από το προηγούμενο με τη συγχώνευση των δυο πλησιέστερων συστάδων. Για να βρει την ομοιότητα δυο συστάδων, χρησιμοποιείται ένα από τα ακόλουθα χαρακτηριστικά κριτήριά:η ελάχιστη, μεγίστη, ή μέση pairwise απόσταση μεταξύ των σημείων των δυο συστάδων.

- *Διαιρετικούς (Divisive) Ιεραρχικούς Αλγορίθμους*. Αυτού οι αλγόριθμοι παράγουν μια ακολουθία σχημάτων συσταδοποίησης που αυξάνουν τον αριθμό συστάδων σε κάθε βήμα. Σε αντίθεση με τους συσσωρευτικούς αλγορίθμους η συσταδοποίηση που παράγεται σε κάθε βήμα από τον προηγούμενο αλγόριθμο οδηγεί στο διαχωρισμό μιας συστάδας σε δυο.

Στη συνεχεία περιγράφω μερικούς αντιπροσωπευτικούς ιεραρχικούς αλγορίθμους συσταδοποίησης:

CURE

Ο CURE είναι ένας ιεραρχικός αλγόριθμος συσταδοποίησης του οποίου τα βασικά χαρακτηριστικά είναι ότι :

- μπορει να αναγνωρίζει συστάδες αυθαίρετων σχημάτων (π.χ. ελλειψοειδή)
- είναι εύρωστος στην παρουσία των outliers.
- οι απαιτήσεις του χώρου αποθήκευσης είναι γραμμική συνάρτηση του αριθμού των στοιχείων εισόδου και η χρονική πολυπλοκότητα του είναι O_{n^2} για δεδομένα μικρών διαστάσεων, όπου n είναι ο αριθμός των στοιχείων εισόδου.

ο αλγόριθμος μπορεί να εφαρμοστεί αποδοτικά και για συσταδοποίηση μεγάλων βάσεων δεδομένων συνδυάζοντας τεχνικές τυχαίας δειγματοποίησης και τμηματοποίησης . Επομένως, τα δεδομένα που εισάγονται στον αλγόριθμο μπορεί να είναι ένα δείγμα που επιλέχθηκε τυχαία από το αρχικό σύνολο δεδομένων ή ένα υποσύνολο αυτού του δείγματος εάν προηγουμένως εφαρμοστεί η συσταδοποίηση.

Περιγραφή Αλγόριθμου

Ο αλγόριθμος αρχίζει λαμβάνοντας κάθε σημείο εισόδου σαν ξεχωριστή συστάδα και σε κάθε βήμα που ακολουθεί συγχωνεύει τα πλησιέστερα ζευγάρια συστάδων. Προκειμένου να υπολογίσουμε την απόσταση μεταξύ των συστάδων, αποθηκεύονται για κάθε συστάδα c αντιπρόσωποι. Οι αντιπρόσωποι αυτοί καθορίζονται επιλέγοντας αρχικά τα c πιο διάσπαρτα σημεία μέσα σε μια συστάδα και στη συνέχεια μετακινούμε τα σημεία προς το μέσο της συστάδας κατά ένα ποσοστό α.Η απόσταση μεταξύ των συστάδων είναι η απόσταση των πιο κοντινών αντιπροσώπων δύο συστάδων.Έτσι μόνο τα σημεία αντιπρόσωποι μιας συστάδας χρησημοποιούνται για να υπολογίσουμε την απόσταση της μίας από την άλλη συστάδα.

Οι c αντιπρόσωποι προσπαθούν να προσδιορίσουν με φυσικό σχήμα και τη γεωμετρία της συστάδας. Επιπρόσθετα,μετακινώντας τα διάσπαρτα σημεία προς το μέσο κατά ένα ποσοστό απομακρύνουμε το θόρυβο και μετριάζουμε τις επιδράσεις των outliers.Ο λόγος που γίνεται αυτό είναι ότι οι outliers βρίσκονται τυπικά μακριά από το κέντρο της συστάδας και έτσι η συρρίκνωση θα κάνει τους outliers να κινηθούν περισσότερο προς το κέντρο ενώ οι αντιπρόσωποι που θα

απομείνουν θα υποστούν ελάχιστη μετακίνηση. Οι μεγάλες μετακινήσεις στους outliers θα μειώσουν τη δυνατότητα τους να προκαλέσουν συγχώνευση λάθος συστάδων.Η παράμετρος α μπορεί επίσης να να χρησιμοποιηθεί και για τον έλεγχο του σχήματος των συστάδων.Μια μικρή τιμή για το α συρρικνώνει τα διάσπαρτα σημεία πολύ λίγο και έτσι ενισχύει την ύπαρξη συστάδων που δεν είναι σφαιρικές.Αντίθετα,μεγάλες τιμές για το α έχουν σαν αποτέλεσμα τη δημιουργία συμπαγών συστάδων καθώς τα διάσπαρτα σημεία τοποθετούνται πιο κοντά στο μέσο των συστάδων.

2.4.4 Ιεραρχική και βασισμένη σε γράφους συσταδοποίηση

CHAMELEON

Ένας αλγόριθμος συσταδοποίησης αυτής της κατηγορίας είναι ο CHAMELEON.Είναι ένας συσσωρευτικός (agglomerative) ιεραρχικός αλγόριθμος που μετρά την ομοιότητα δύο συστάδων που βασίζονται σε ένα δυναμικό μοντέλο.Ο CHAMELEON βρίσκει τις συστάδες του συνόλου δεδομένων χρησιμοποιώντας έναν αλγόριθμο δύο φάσεων.Κατά την διάρκεια της πρώτης φάσης, ο CHAMELEON χρησιμοποιεί έναν αλγόριθμο βασισμένο σε γράφους για να τμηματοποιήσει τα δεδομένα σε έναν μεγάλο αριθμό σχετικά μικρών υποσυστάδων.Κατά τη διάρκεια της δεύτερης φάσης,χρησιμοποιεί έναν συσσωρευτικό ιεραρχικό αλγόριθμο για να βρει τις συστάδες από επαναληπτικούς συνδιασμούς των υποσυστάδων που προέκυψαν από την πρώτη φάση.Η ομοιότητα μεταξύ των συστάδων καθορίζεται με τον έλεγχο της *σχετικής ενδο-συνδετικότητας* (inter-connectivity) και της *σχετικής εγγύτητας* (closeness) αυτών.Η αναπαραγωγή των δεδομένων βασίζεται στη συνήθως χρησημοποιημένη προσέγγιση του k-πλησιέστερου γράφου γειτνίασης. Οι κορυφές του k-πλησιέστερου γράφου γειτνίασης αντιπροσωπέυουν τα αντικείμενα του συνόλου δεδόμενων και υπάρχει μια ακμή μεταξύ δύο κόμβων v_i, v_j εάν το το αντικείμενο που αντιστοιχεί στον v_i είναι μεταξύ των k κοντινότερων γειτόνων του v_j. Κατόπιν ο αλγόριθμος βρίσκει τις αρχικές υποσυστάδες χρησιμοποιώντας έναν αλγόριθμο τμηματοποίησης γράφου ώστε να κατατμηθεί ο k-πλησιέστερος γράφος γειτνίασης του συνόλου δεδομένων σε έαν μεγάλο αριθμό τμημάτων.Κατά τη διάρκεια της επόμενης φάσης ο CHAMELEON χρησιμοποιεί έναν συσσωρευτικό αλγόριθμο συσταδοποίησης ο οποίος συνδυάζει μαζί αυτές τις υποσυστάδες του γράφου.Για τη συγχώνευση των υποσυστάδων λαμβάνει τη σχετική ενδοδυνδετικότητα ακι την εγγύτητα των υποσυστάδων.Κατα συνέπεια,εκείνα ζευγάρια των συστάδων των οποίων η σχετική ενδο-συνδετικότητα και εγγύτητα είναι πάνω από το όριο που ορίζεται από τους χρήστες συγχωνεύεται.

2.4.5 Βασισμένη στη πυκνότητα συσταδοποίηση (Density-based clustering)

Οι βασισμένοι στην πυκνότητα αλγόριθμοι θεωρούν τις συστάδες ως πυκνές περιοχές αντικειμένων στο χώρο των υπό μελέτη δεδομένων, οι οποίες χωρίζονται από περιοχές της χαμηλής πυκνότητας.

BDSCAN

Ο DBSCAN είναι ένας αλγόριθμος συσταδοποίησης ο οποίος βασίζεται στην πυκνότητα.Η βασική ιδέα είναι ότι η περιοχή που επεκτείνεται σε συγκεκριμένη ακτίνα (Eps) γύρω από κάθε σημείο μιας συστάδας (γειτονιά αντικειμένου) θα πρέπει να περιέχει έναν ελάχιστο αριθμό από αντικείμενα.

Βασικές έννοιες αλγορίθμου

Στην συνέχεια θα αναφερθούμε εν συντομία στις κυριότερες έννοιες που αποτελούν βάση του αλγορίθμου.Θεωρώντας λοιπόν ότι έχουμε ένα σύνολο αντικειμένων D στο οποίο η γειτονιά κάθε αντικείμενου εκτείνεται σε ακτίνα Eps γύρω από αυτό και ο ελάχιστος αριθμός στοιχείων που μπορεί να περιέχει είναι MinPts, μπορούμε να ορίσουμε τις εξής έννοιες:

- Ένα αντικείμενο p είναι άμεσα πυκνά-προσεγγίσιμο από ένα αντικείμενο q εάν
  1. το αντικείμενο ανήκει στο υποσύνολο των αντικειμένων που βρίσκονται στη γειτονιά του q
  2. ο αριθμός των αντικειμένων που περιέχονται στη γειτονιά του q είναι μεγαλύτερος από ένα όριο MinPts
- Ένα αντικείμενο p είναι πυκνά-προσεγγίσιμο από ένα αντικείμενο q, $p >_D q$, εάν υπάρχει μια ακολουθία από αντικείμενα p_1,…,p_n, $p_1 = q$, $p_n = p$ τέτοια ώστε το $p_i + 1$ να είναι άμεσα πυκνά-προσεγγίσιμο από το p_i.
- Ένα αντικείμενο p είναι πυκνά-σννδεδεμένο με ένα αντικείμενο q εάν υπάρχει ένα αντικείμενο o τέτοιο ώστε τόσο το p όσο και το q να είναι πυκνά-προσεγγίσιμα από το o.
- Μία συστάδα C στο σύνολο των δεδομένων D είναι ένα μη-κενό υποσύνολο του D το οποίο ικανοποιεί τις ακόλουθες συνθήκες:
  1. Για κάθε $p, q \in D$: εάν $p \in c$ και $q >_D p$, τότε $q \in C$

2. Για κάθε $p, q \in D$: το p είναι πυκνά-συνδεδεμένο με το q.

- Έστω ότι C_1, C_2,..., C_n είναι οι συστάδες του συνόλου δεδομένων D. Ορίζουμε ως θόρυβο το σύνολο των αντικειμένων στην βάση δεδομένων D τα οποία δεν ανήκουν σε καμία συστάδα Ci.

Βήματα αλγορίθμου DBSCAN

Ο αλγόριθμος συσταδοποίησης BDSCAN μπορεί να περιγραφεί με τη μορφή ψευδοκώδικα ως εξής:

Algorithm DBSCAN (**D, Eps, MinPts**)
//Προϋπόθεση: Όλα τα αντικείμενα στο σύνολο δεδομένων **D** δεν έχουν τοποθετηθεί σε συστάδες.
FORALL objects **o** in D **DO:**
IF o δεν έχει ταξινομηθεί **THEN**
Κάλεσε την συνάρτηση expand_cluster προκειμένου να κατασκευαστεί μία συστάδα με ακτίνα Eps και ελάχιστο αριθμό στοιχείων MinPts το οποίο θα περιέχει το **o.**
Function expand_cluster(o, D, Eps, MinPts):
Ανάκτηση της Eps-γειτονιάς N_{Eps}(o) του o;
if |NEPS(o)| <MinPts //δηλαδή **o** δεν είναι αντικείμενο πυρήνα **then**
Σημείωσε το **o** σαν θόρυβο, **RETURN;**
else //το **o** είναι αντικείμενο πυρήνα
Επέλεξε ένα νέο cluster_id και σημείωσε όλα τα αντικείμενα στο N_{EPS}(o)
με το τρέχον cluster_id;
ώθησε όλα τα αντικείμενα από το N_{EPS}(o)-{o}στην στοίβα seeds;
while not seeds.empty() **do**
currentObject:=seed.top();
ανάκτηση της Eps-γειτονιάς του τρέχοντος αντικειμένου;
if |N_{EPS}(currentObject)| > MinPts **then**
επέλεξε όλα τα αντικείμενα N_{EPS}(currentObject) που δεν έχουν ταξινομηθεί ακόμα ή έχουν σημειωθεί ως θόρυβος, τοποθέτησε τα μη κατηγοριοποιημένα αντικείμενα στην στοίβα **seeds** και σημείωσε όλα τα αυτά αντικείμενα με το τρέχον cluster_id;
seeds.pop();
RETURN

2.4.6 Βασισμένοι σε πλέγμα (Grid-based)

Προσφατα διάφοροι αλγόριθμοι συσταδοποίησης έχουν παρουσιαστεί για χωρικά δεδομένα,οι οποίοι είναι γνωστοί ως αλγότιθμοι βασισμένοι σε πλέγμα.Αυτοί οι αλγόριθμοι κβαντοποιούν το διάστημα σε έναν πεπερασμένο αριθμό κελιών και κάνουν έπειτα όλες τις διαδικασίες στο κβαντοποιημένο διάστημα.

STING (Statistical Information Grid-based method)

Η μέθοδος STING είναι αντιποσωπευτικη αυτης της κατηγοριας. Διαιρει την χωρικη περιοχή σε ορθογώνια κελια χρησημοποιώντας μια ιεραρχική δομή. Η μέθοδος STING πηγαίνει μέσω της συστάδας δεδομένων και υπολογίζει τις στατιστικές παραμέτρους (όπως μέσος, διακύμανση, ελάχιστο, μέγιστο και τύπος κατανομής) κάθε αριθμητικού γνωρίσματος των δεδομένων μέσα στα κελιά.Κατόπιν παράγει μια ιεραρχική δομή των κελιών πλέγματος ώστε να αντιπροσωπευθούν οι πληροφορίες συσταδοποίησης σε διαφορετικά επίπεδα. Βασισμένος σε αυτήν την δομή ο αλγόριθμος STING επιτρέπει τη χρήση πληροφοριών συσταδοποίησης στην αναζήτηση των ερωτήσεων ή της αποδοτικής ανάθεσης ενός νέου αντικειμένου σε συστάδες.

2.4.7 Συσταδοποίηση υποχώρων (Subspace Clustering)

Η συσταδοποίηση υποχώρων (subspace clustering) εξετάζει τα προβλήματα που προκύπτουν από τις υψηλές διαστάσεις (high dimensionality) των δεδομένων. Λόγω των πολλών διαστάσεων και της ύπαρξης των διαστάσεων που αντιστοιχούν σε θόρυβο, σχεδόν κάθε περιοχή στο χώρο έχει χαμηλή πυκνότητα σημείων, και όλα τα σημεία είναι μακριά το ένα από το άλλο.

CLIQUE

Με βάση τα παραπάνω προβλήματα των πολλαπλών διαστάσεων στην εργασία εξετάστηκε το πρόβλημα του αυτόματου ορισμού διάφορων υποχώρων (subspaces) του αρχικού χώρου οι οποίοι επιτρέπουν ”καλύτερη” συσταδοποίηση των στοιχείων ενός συνόλου δεδομένων. Ειδικότερα χρησιμοποιήθηκε μια βασισμένη στην πυκνότητα προσέγγιση για να προσδιοριστούν οι συστάδες.Ο αλγόριθμος CLIQUE προχωρά από χαμηλότερης έως υψηλότερης διάσταση υποχώρους και ανακαλύπτει τις πυκνές περιοχές σε κάθε υποχώρο. Για να προσεγγίσει την πυκνότητα των σημείων,το διάστημα εισόδου χωρίζεται στα κελιά με τη διαίρεση κάθε διάστασης στον ίδιο αριθμό,x_i,ίσου μήκους διαστημάτων. Για ένα δεδομένο σύνολο

διαστάσεων, ο συνδυασμός των αντίστοιχων διαστημάτων (ένα για κάθε διάσταση του συνόλου) καλείται μονάδα (unit) στον αντίστοιχο υποχώρο. Μία μονάδα είναι πυκνή εάν ο αριθμός σημείων είναι επάνω από ένα δεδομένο όριο t. Τα x_i και t είναι παράμετροι που καθορίζονται από το χρήστη. Ο αλγόριθμος βρίσκει όλες τις πυκνές μονάδες σε κάθε k-διάστατο υποχώρο με τη δημιουργία των πυκνών (k-1)-διάστατων υποχώρων,και στη συνέχεια τις συνδέει για να περιγράψουν συστάδες ως ένωση των μέγιστων ορθογωνίων.

3. ΕΙΔΙΚΟ ΜΕΡΟΣ

3.1 Λογισμικό

3.1.1 R

Η R ειναι μια γλωσσα προγραμματισμου και λογισμικο περιβάλλον για στατιστικους σκοπους.Η R χρησημοποιειται ευρέως απο μαθηματικούς και data miners για δημιουργία στατιστικων προγραμμάτων και ανάλυση δεδομένων.Τα τελευταία χρόνια εχει γίνει μια τεράστια αυξηση ενδιαφέροντος της R απο τους αναλυτες δεδομένων.Η R είναι μια υλοποίηση της γλώσσας προγραμματισμού S σε συνδυασμό με λεξιλογικές σημασιολογία οριοθέτησης εμπνευσμένο από τον Scheme.Η S δημιουργήθηκε από τον John Chambers , στα Bell Labs . Υπάρχουν κάποιες σημαντικές διαφορές , αλλά ένα μεγάλο μέρος του κώδικα της S μένει ίδιο .Η R δημιουργήθηκε απο τους Ross Ihaka και Robert Gentleman του Πανεπιστημίου του Οκλαντ της Νεας Ζηλανδίας και πλέον αναπτύσεται απο την R Development Core Team,στην οποία ο Chambers ειναι μέλος. Ο πηγαίος κώδικας της R είναι κυρίως γραμμένος σε C,Fortran και R.Η άδεια χρησης της R ειναι GNU General Public License την οποία την καθειστά στην κατηγορία του ελεύθερου λογισμικού.

Στατιστικά χαρακτηριστικά

Η R και οι βιβλιοθήκες της εφαρμόζουν μια ευρεία ποικιλία των στατιστικών και γραφικών, συμπεριλαμβανομένων των γραμμικών και μη γραμμικών μοντέλων, κλασικούς στατιστικούς ελέγχους, ανάλυση χρονοσειρών, ταξινόμηση, ομαδοποίηση, και άλλα.Η R είναι εύκολα επεκτάσιμη μέσω λειτουργιών και των επεκτάσεων, και η R κοινότητα είναι γνωστή για την ενεργό συμβολή της όσον αφορά τα πακέτα. Πολλές από τις τυπικές λειτουργίες, είναι γραμμένες μέσα στο ίδιο το R, το οποίο το καθιστά εύκολο για τους χρήστες να ακολουθούν τις αλγοριθμικές επιλογές που γίνονται. Για υπολογιστικά εντατικές εργασίες, κώδικα γραμμένου σε C, C ++, και Fortran μπορούν να συνδεθούν και να ονομάζεται κατά το χρόνο

εκτέλεσης. Οι προχωρημένοι χρήστες μπορούν να γράψουν C, C ++, Java,.NET ή και Python για να χειραγωγήσουν R αντικείμενα άμεσα.

Η R είναι εξαιρετικά επεκτάσιμη μέσω της χρήσης των πακέτων τα οποία υποβάλλονται για συγκεκριμένες λειτουργίες ή συγκεκριμένες περιοχές της μελέτης. Λόγω της S κληρονομιάς,η R έχει τους ισχυρότερους δεσμούς με την έννοια του αντικειμενοστραφούς προγραμματισμού από τις περισσότερες γλώσσες στατιστικούς υπολογισμούς.

Ένα ακόμα ισχυρό χαρακτηριστικό της R είναι τα στατιστικα γραφήματα τα οποία μπορουν να είναι επιπέδου δημοσίευσης και λόγω οτι η R εχει την δικη του Latex μορφή για την εκδοση αναφορών η οποία χρησιμοποιείται για να παρέχει πλήρη τεκμηρίωση, τόσο on-line τόσο και σε έντυπη μορφή.

Πακέτα (Packages)

Οι δυνατότητες της R έχουν επεκταθεί μέσω πακέτων που έχουν δημιουργήσει χρήστες, οι οποίες επιτρέπουν εξειδικευμένες στατιστικές τεχνικές, γραφικές παραστάσεις (ggplot2), δυνατότητα εισαγωγής / εξαγωγής δεδομένων, εργαλεία για την υποβολή εκθέσεων(knitr, Sweave) , κλπ Αυτά τα πακέτα που αναπτύσσονται κυρίως στο R, και μερικές φορές σε Java, C, C ++ και Fortran. Ένα βασικό σύνολο πακέτων περιλαμβάνεται με την εγκατάσταση του R, με περισσότερα από 5.800 επιπλέον πακέτα και 120.000 λειτουργιών (από τον Ιούνιο του 2014) να διατίθενται στο Comprehensive R Archive Network (CRAN), Bioconductor, Omegahat, GitHub και άλλα αποθετήρια.

Στην ιστοσελίδα CRAN μπορεί κάποιος να βρει ένα ευρύ φάσμα από τομείς όπως τα χρηματοοικονομικά,γενετική, High Performance Computing, Μηχανική Μάθηση, Ιατρικής Απεικόνισης, Κοινωνικών Επιστημών και Χωρική Στατιστική στους οποίους η R έχει εφαρμοστεί και για τους οποίους τα πακέτα είναι διαθέσιμα.Η R έχει επίσης χαρακτηριστεί από το FDA ως κατάλληλο για την ερμηνεία των στοιχείων από τις κλινικές έρευνες.

Το έργο Bioconductor παρέχει R πακέτα για την ανάλυση των γονιδιακών δεδομένων, όπως τον χειρισμό δεδομένων μικροσυστοιχιών και εργαλεία ανάλυσης τους, και έχει αρχίσει να παρέχει εργαλεία για την ανάλυση των δεδομένων από επόμενης γενιάς μεθόδους υψηλής απόδοσης.

Τα πακέτα τα οποία χρησιμοποίησα για την διπλωματική μου εργασία από το CRAN είναι:

- orclus: Πακέτο με το subspace clustering.
- rgl: Πακέτο που παρέχει εργαλεία για 3D γραφικές παραστάσεις.

- stringr: Πακέτο για την καλύτερη εξαγωγή κειμένου την ώρα της εκτέλεσης του κώδικα.

3.1.2 Biocoductor

Το Bioconductor είναι ένα δωρεάν, ανοιχτού κώδικα και ανοιχτής ανάπτυξης έργου λογισμικού για την ανάλυση και την κατανόηση των γονιδιωματικών δεδομένων που παράγονται σε εργαστηριακά πειράματα στη μοριακή βιολογία.

Το Bioconductor βασίζεται κυρίως στη στατιστική R γλώσσα προγραμματισμού, αλλά περιέχει συνεισφορές σε άλλες γλώσσες προγραμματισμού. Έχει δύο κυκλοφορίες κάθε χρόνο που ακολουθούν τα εξαμηνιαία δελτία της R. Ανά πάσα στιγμή υπάρχει μια έκδοση, η οποία αντιστοιχεί στην επίσημη έκδοση της R, και μια έκδοση υπό ανάπτυξη, η οποία αντιστοιχεί στην έκδοση ανάπτυξη της R. Οι περισσότεροι χρήστες θα βρούν την νέα έκδοση κατάλληλη για τις ανάγκες τους. Επιπλέον, υπάρχει ένας μεγάλος αριθμός των πακέτων σχολιασμού γονιδιώματος διαθέσιμος που είναι κυρίως, αλλά όχι αποκλειστικά, προσανατολισμένος σε διαφορετικά είδη μικροσυστοιχιών.

Πακέτα (Packages)

Τα πακέτα τα οποία χρησιμοποίησα για την διπλωματική μου εργασία από το Bioconductor είναι:

- flowCore: Παρέχει εργαλεία για την εισαγωγή των δεδομένων.
- flowDensity: Παρέχει εργαλεία για τις γραφικές παραστάσεις δεδομένων κυτταρομετρίας ροής και τον καθαρισμλο των δεδομένων.
- flowTrans: Παρέχει εργαλεία για την προεπεξεργασία των δεδομένων και ειδικότερα για το μετασχηματισμό των δεδομένων.
- flowStats: Παρέχει εργαλεία για την προεπεξεργασία των δεδομένων και ειδικότερα για την κανονικοποίηση των δεδομένων.

3.2 Δεδομένα

Το σύνολο δεδομένων που χρησιμοποιηθήκατε αφορά αιματολογικά δεδομένα και μου δόθηκαν από το Πανεπιστήμιο Πιερ και Μαρί Κιουρί .Πιο συγκεκριμένα πρόκειται για ένα σετ δεδομένων με τα ακόλουθα χαρακτηριστικά.

Αντιγόνα CD (cluster of differentiation antigens): Συνάθροισμα αντιγόνων διαφοροποίησης που εντοπίζονται σε ορισμένα κύτταρα μέσω των ειδικών αντισωμάτων. Πολλά αντιγόνα διαφοροποίησης παίζουν σημαντικούς λειτουργικούς ρόλους χαρακτηριστικούς των διαφοροποιημένων φαινοτύπων των κυττάρου στο οποίο εκφράζονται, όπως η ανοσοσφαιρίνη κυτταρικής επιφάνειας στα κύτταρα Β.

Το αρχείο Lympho B.fcs περιέχει τα εξής δεδομένα(εκτός των scatter δεδομένων):

- IGD FITC
- CD10 PE
- CD5 ECD
- CD27 PC5.5
- CD38 PC7
- IGM APC
- CD19 AA700
- CD24 A750
- CD21 PB
- CD45 KO

Αντίστοιχα το αρχείο Polarisation.fcs περιέχει τα εξής δεδομένα(εκτός των scatter δεδομένων):

- CXCR3 FITC
- CCR10 PE
- CCR7 PE
- CCR6 PERCPCY5.5

- CCR4 PC
- CXCR5 APC
- CD3 AA700
- CD45RA AA750
- CD4 PB
- CD8 KO

Τέλος το αρχείο Naive Memory.fcs περιέχει τα εξής δεδομένα(εκτός των scatter δεδομένων):

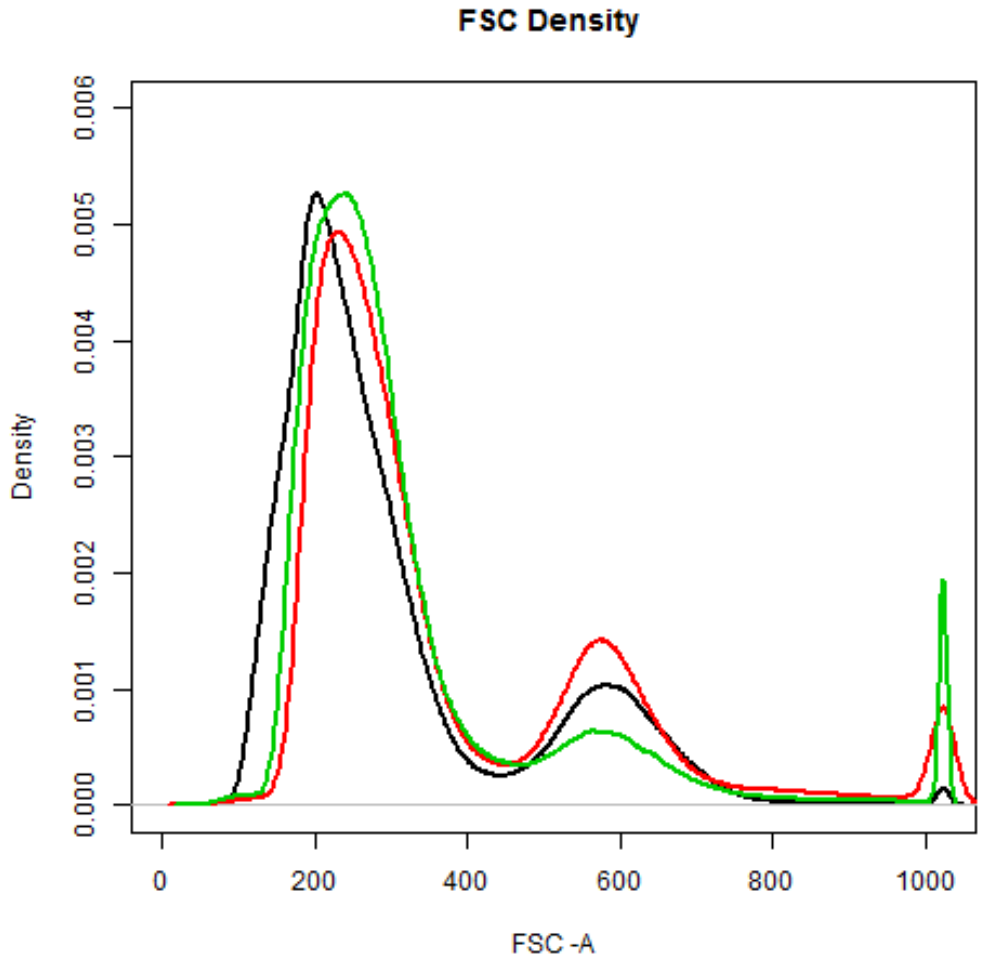
- CD95 FITC
- CCR7 PE
- HLA DR ECD
- CD25 PC5.5
- CD45 RA PC7
- ICOS APC
- CD3 AA700
- CD127 AA750
- CD4 PB 1024
- CD8 KRO

Όλα τα παραπάνω δεδομένα περιέχουν την ώρα που έγινε το πείραμα η οποία αφαιρείται στην προεπεξεργασία και τα scatter δεδομένα (FSC-A,SSC-A,SSC-H,SSC-W,FSC-W).

3.3 Προεπεξεργασία

Πριν από την ανάλυση, τα δεδομένα της κυτταρομετρίας ροής θα πρέπει κατά κανόνα να υποβάλλονται σε προεπεξεργασία για να αφαιρούνται αντικείμενα όπως νεκρά κύτταρα ώστε να αποφεύγεται η κακή ποιότητα των δεδομένων, και να μετατρέπεται σε μια βέλτιστη κλίμακα για τον εντοπισμό των κυττάρων του πληθυσμού ενδιαφέροντος. Παρακάτω είναι τα διάφορα στάδια σε μια τυπική προεπεξεργασία δεδομένων κυτταρομετρίας ροής.

Με μία απλή μέθοδο μπορώ να δω αν είναι καλής ποιότητας τα δεδομένα.Αυτό γίνεται με μια γραφική παράσταση του FSC-A και των τριών συνόλων.Πρέπει να είναι η γραφική παράσταση της ίδιας μορφής.Αλλιώς κάποιο τεχνικό πρόβλημα υπάρχει.


FSC Density
Density
0.000
0.001
0.002
0.003
0.004
0.005
0.006
0
200
400
600
800
1000
FSC -A

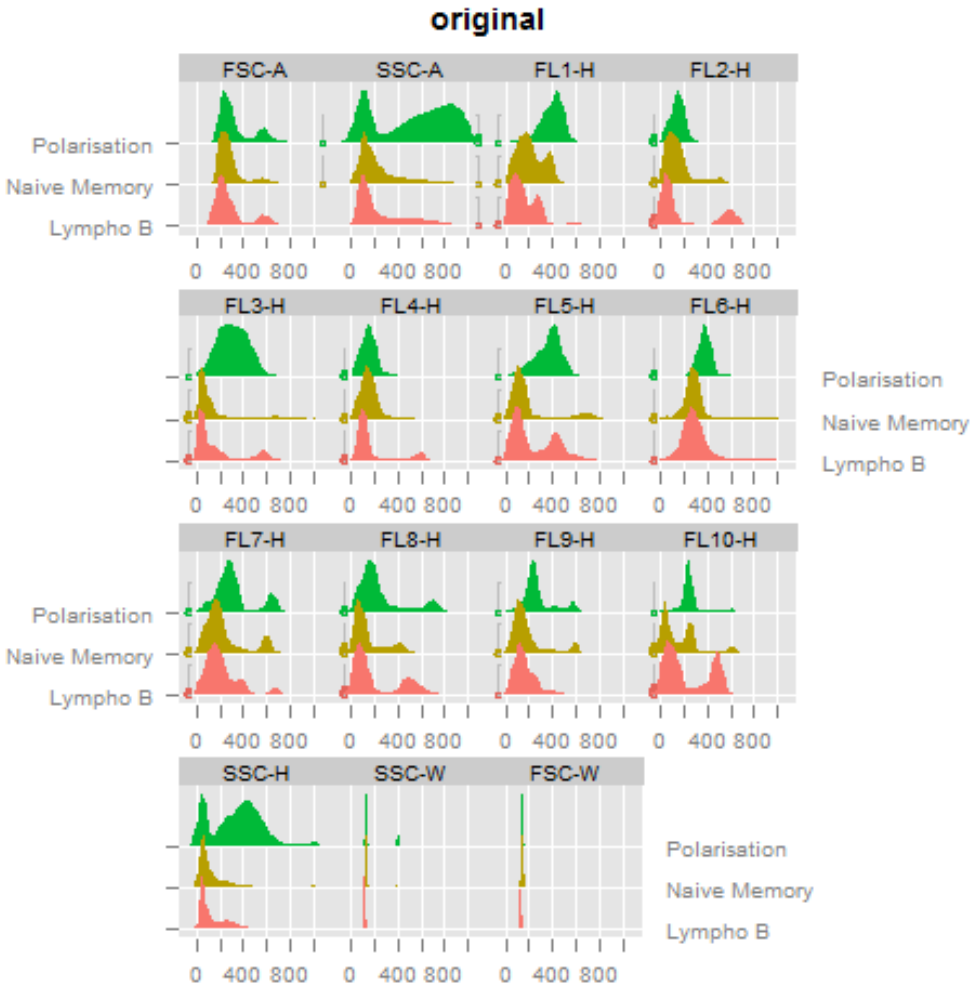


Fig. 3.1: Δεδομένα πριν την προεπεξεργασία

- Καθαρισμός δεδομένων (Margin Events): Αφαίρεση νεκρών κυττάρων και απομάκρυνση ακραίων δεδομένων (outliers).

- Αντιστάθμιση Φθορισμών (Compensate): Όταν περισσότερα από ένα φθοριοχρώμιο χρησιμοποιείται με το ίδιο λέιζερ, τα φάσματα εκπομπής τους συχνά αλληλεπικαλύπτονται. Κάθε συγκεκριμένο φθορόχρωμα τυπικά μετράται χρησιμοποιώντας ένα ζωνοπερατό οπτικό φίλτρο ρυθμισμένο σε μια στενή ζώνη στο ή πλησίον στην κορυφή έντασης εκπομπής του φθοριοχρω-

μίου του. Το αποτέλεσμα είναι ότι η ανάγνωση για κάθε δοσμένο φθορόχρωμα είναι στην πραγματικότητα το άθροισμα της μέγιστης έντασης εκπομπών που φθοριοχρώμιου, καθώς και η ένταση των φασμάτων όλων των άλλων φθοριοχρώματων όπου επικαλύπτονται με την εν λόγω ζώνη συχνοτήτων. Η επικάλυψη αυτή ονομάζεται spill-over, και η διαδικασία αφαίρεσης διάχυσης από τα δεδομένα κυτταρομετρίας ροής ονομάζεται Αντιστάθμιση Φθορισμών (compensate).

Η αντιστάθμιση φθορισμών συνήθως επιτυγχάνεται με την εκτέλεση μιας σειράς αντιπροσωπευτικών δειγμάτων από κάθε βάφονται μόνο ένα φθοριοχρώμιο, για να δώσει μετρήσεις της συνεισφοράς του κάθε φθοριοχρώμου σε κάθε κανάλι. Το συνολικό σήμα για την απομάκρυνση από κάθε κανάλι μπορεί να υπολογιστεί από την επίλυση ενός συστήματος γραμμικών εξισώσεων με βάση αυτά τα δεδομένα για να παραχθεί ο πίνακας spill-over , ο οποίος όταν αναστρέφεται και πολλαπλασιάζεται με τα ανεπεξέργαστα δεδομένα παράγει τα δεδομένα από την αντιστάθμιση φθορισμών . Οι διαδικασίες υπολογισμού του πίνακα spill-over , ή εφαρμόζοντας τον πίνακα spill-over ,ο οποίος μπορεί να δίνεται,στα ανεπεξέργαστα δεδομένα της κυτταρομετρία ροής , είναι στα βασικά χαρακτηριστικά του λογισμικού κυτταρομετρίας ροής

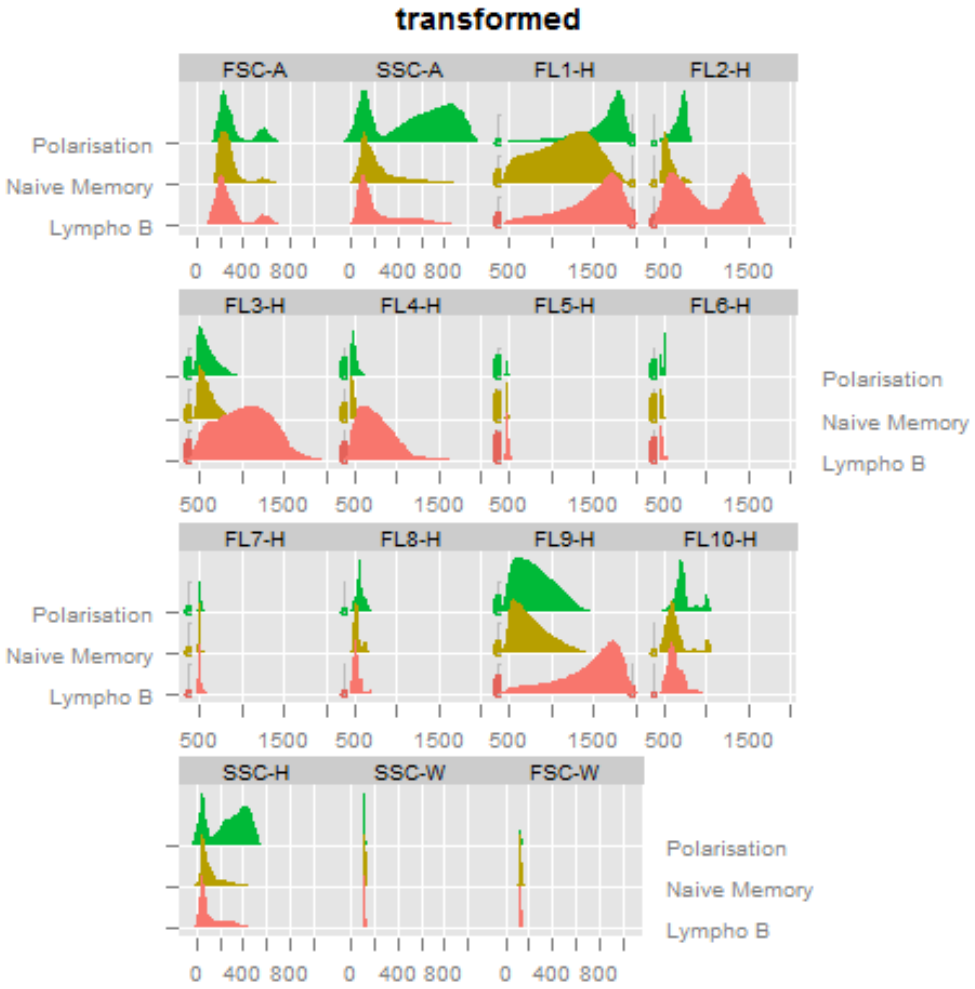
- Μετασχηματισμός δεδομένων (Transformation):

  Ανιχνεύονται πληθυσμοί κυττάρων στην κυτταρομετρία ροής οι οποίοι συνήθως περιγράφονται έχοντας περίπου λογαριθμική μορφή.Οπότε αυτά συνήθως μετασχηματίζονται σε λογαριθμική κλίμακα.Στους αρχικές συσκευές κυτταρομετρίας ροής αυτό συχνά επιτυγχάνονταν ακόμη και πριν από την απόκτηση δεδομένων με τη χρήση ενός ενισχυτή καταγραφής. Σε σύγχρονα μέσα, τα δεδομένα αποθηκεύονται συνήθως σε γραμμική μορφή, και μετατρέπονται σε ψηφιακά πριν από την ανάλυση.

  Ωστόσο τα δεδομένα μετά την αντιστάθμιση φθορισμών περιέχουν αρνητικές τιμές και εμφανίζονται πληθυσμοί κυττάρων οι οποίοι έχουν μικρούς μέσους και κανονικές κατανομές.Οι λογαριθμικοί μετασχηματισμοί δεν μπορούν να διαχειριστούν επαρκώς τις αρνητικές τιμές. Αντιθέτως υβριδικοί λογαριθμικοί-γραμμικοί μετασχηματισμοί όπως ο Hyperlog ομοίως ο ημιτονοειδής μετασχηματισμός και ο Box-Cox.

  Στην παρούσα διπλωματική χρησημοποίησα έναν υβριδικό λογαριθμικόγραμμικό μετασχηματισμό ο οποίος ονομάζεται Bioexponential και χρησιμοποιεί λογαριθμικό μετασχηματισμό για τις μεγάλες τιμές και γραμμικό

για τις μικρές τιμές.

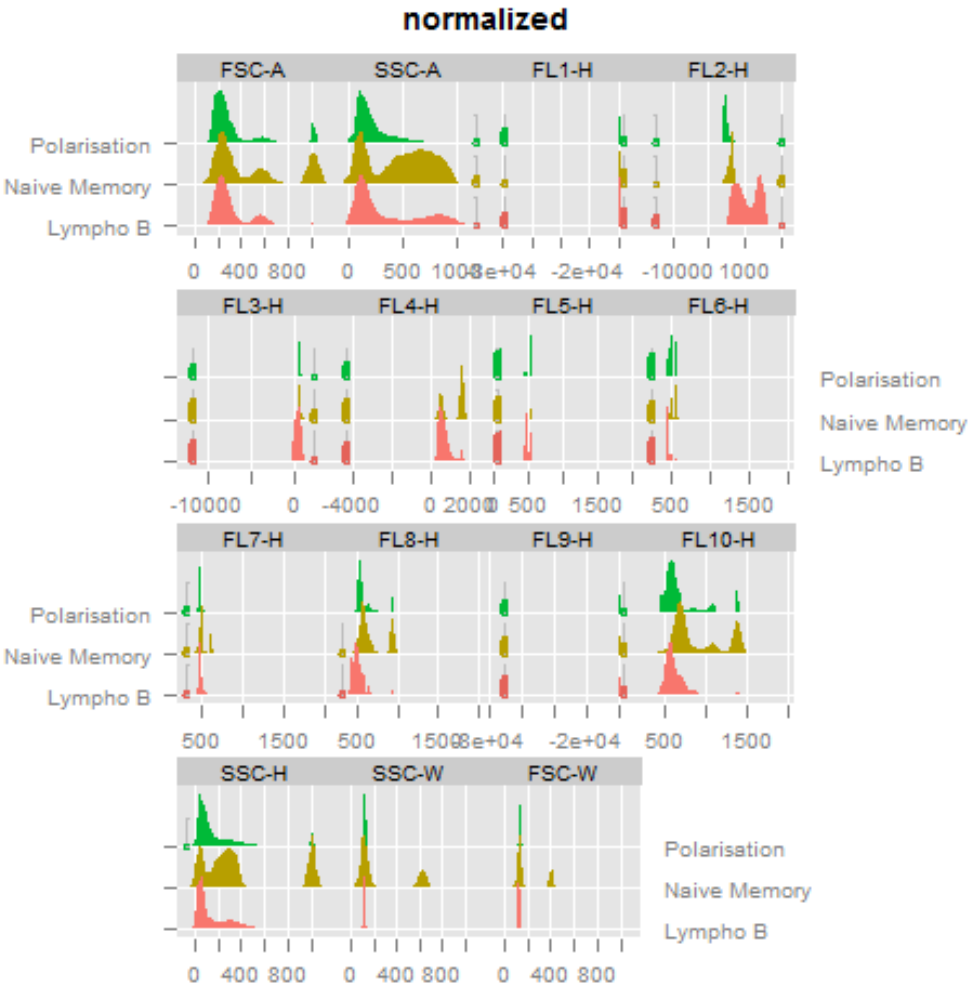


- Κανονικοποίηση (Nornmalize):

  Ιδιαίτερα σε πολυκεντρικές μελέτες,η τεχνική διακύμανση μπορεί να κάνει βιολογικά ισοδύναμους πληθυσμούς των κυττάρων δύσκολο να ταιριάζουν με άλλα δείγματα. Μέθοδοι κανονικοποίησης για την εξάλειψη των τεχνικών διακύμανσης, συχνά προερχόμενα από τις τεχνικές εγγραφής εικόνας, είναι ένα κρίσιμο βήμα σε πολλές αναλύσεις δεδομένων κυτταρομετρίας ροής.

Στην παρούσα διπλωματική χρησιμοποιήθηκε η συνάρτηση warpSet που περιέχεται στο πακέτο flowStats και είναι για κανονικοποίηση δεδομένων κυτταρομετρίας ροής.

Η κανονικοποίηση επιτυγχάνεται όταν πρώτα εντοπίζονται περιοχές υψηλής πυκνότητας (ορόσημα-landmarks) για ένα από τα σύνολα δεδομένων (flowSet) για ένα μόνο κανάλι και στη συνέχεια υπολογίζοντας για κάθε flowFrame (σύνολο όλων των δεδομένων) που ευθυγραμμίζουν καλύτερα αυτά τα ορόσημα . Αυτό βασίζεται στον αλγόριθμο εφαρμόζεται στην λειτουργία landmarkreg στο πακέτο FDA.



3.4 Συσταδοποίηση

Στην παρούσα διπλωματική εργασία χρησημοποποίησα συσταδοποίηση υποχώρων (subspace clustering) σύμφωνα με τον αλγόριθμο CLIQUE. Το αρχικό σύνολο υποχώρων είναι δεκαπέντε και εγώ θεώρησα ότι το τελικό σύνολο να είναι δέκα όσο και το άθροισμα των αντιγόνων κάθε δείγματος.Επίσης έδωσα στον αλγόριθμο συνολικό αριθμό συστάδων να είναι πενήντα.

3.5 Αποτελέσματα

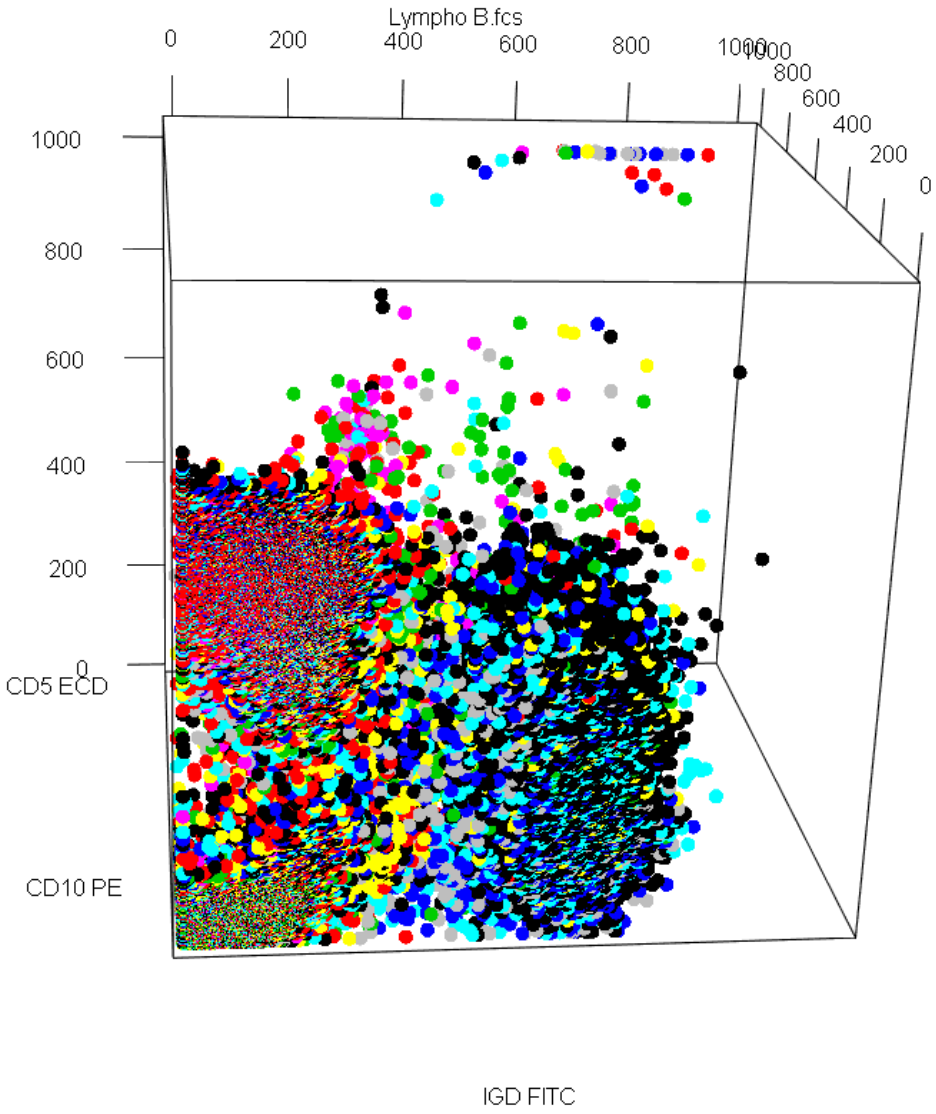
Τα αποτελέσματα που επιστρέφει ο κώδικας είναι:

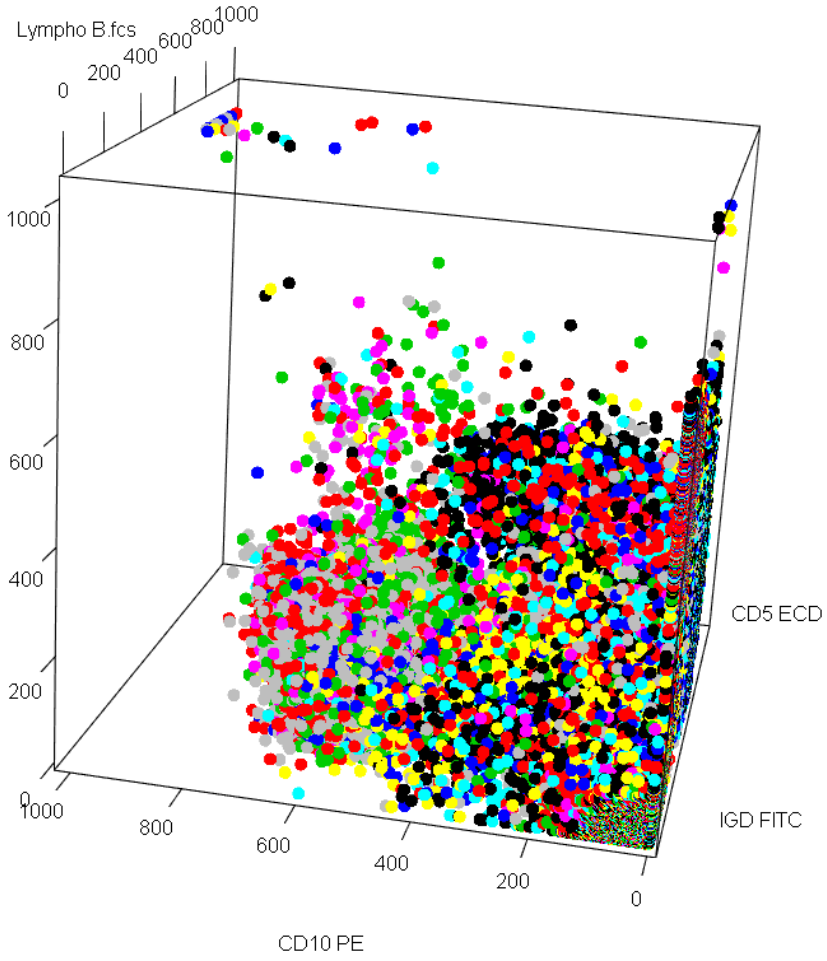
- Πίνακας με τις συστάδες (cluster).
- Πίνακας με τα κέντρα κάθε συστάδας (centers).
- Λίστα με το άθροισμα των στοιχείων κάθε συστάδας (size).
- Λίστα που περιέχει έναν πίνακα για κάθε συστάδα με τα όρια των υποχώρων (subspaces).
- Έναν αριθμό με τον τελική διάσταση των υποχώρων (subspace.dimension).
- Μία τιμή η οποία εάν είναι κοντά στον αριθμό ένα τότε ίσως ο αριθμός των υποχώρων να είναι μεγάλος.Εάν είναι κοντά στο μηδέν τότε δείχνει ότι υπάρχει ισχυρή δομή στις συστάδες (sparsity.coefficient).
  Στα δεδομένα που μου δόθηκαν οι τιμές ήταν κοντά στο μηδέν.

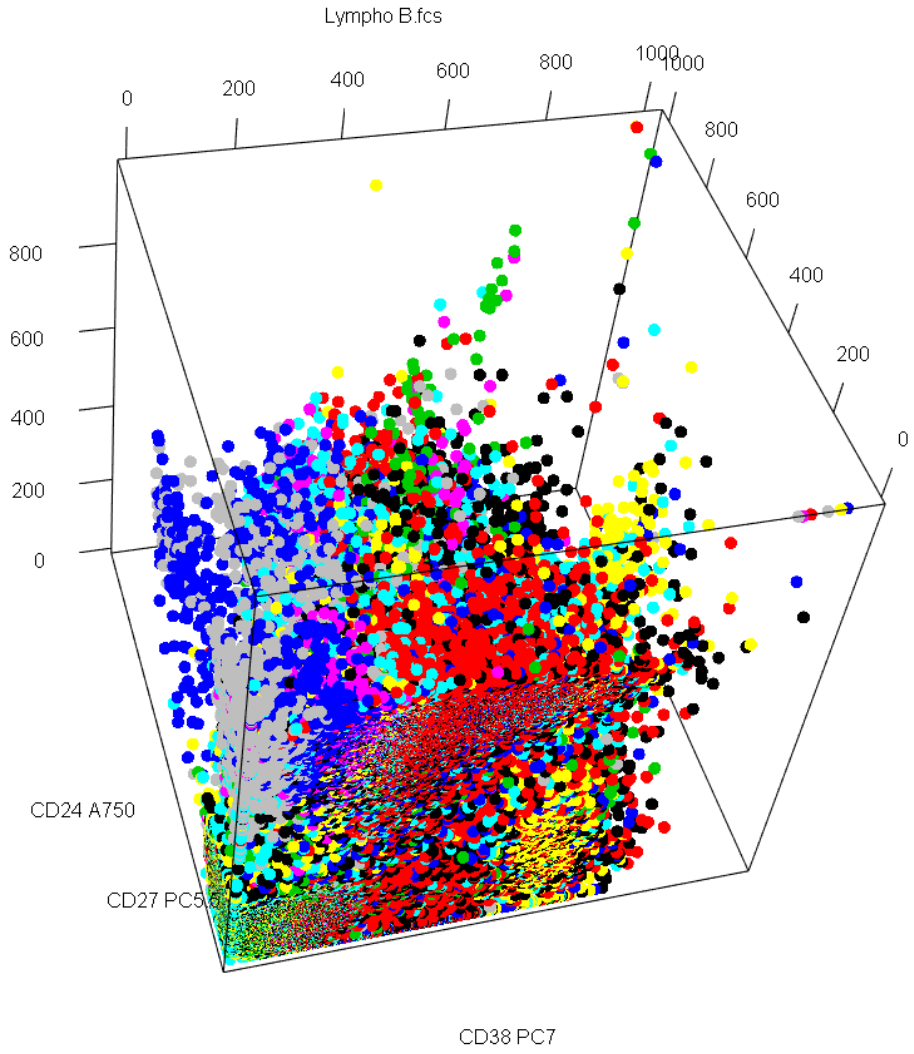
Για καλύτερη απεικόνιση χρησιμοποίησα το πακέτο rgl το οποίο παρέχει τρισδιάστατη απεικόνιση.Μάλιστα για μεγαλύτερη ευκολία υλοποίησα ένα μικρό κομμάτι κώδικα για ακόμα καλύτερη απεικόνιση.

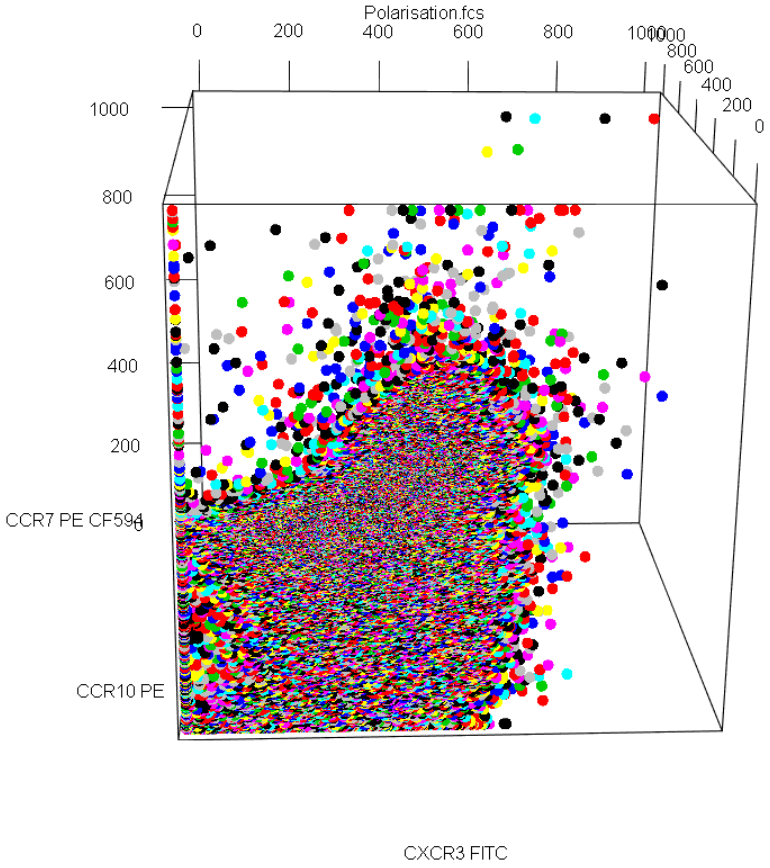
Όλος ο κώδικας χρειάζεται δύο ώρες και σαράντα λεπτά για να ολοκληρωθεί σε έναν επεξεργαστή με τέσσερα νήματα.

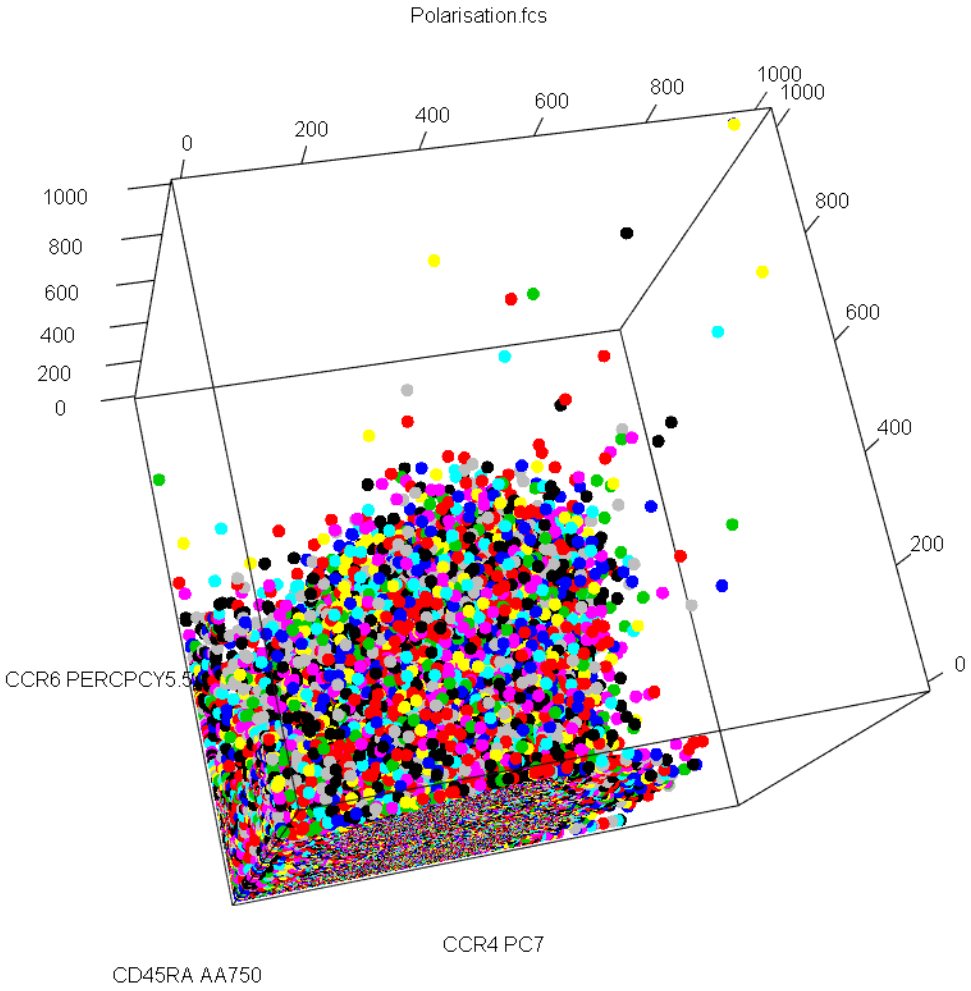
Εδώ είναι εικόνες με τα αποτελέσματα:

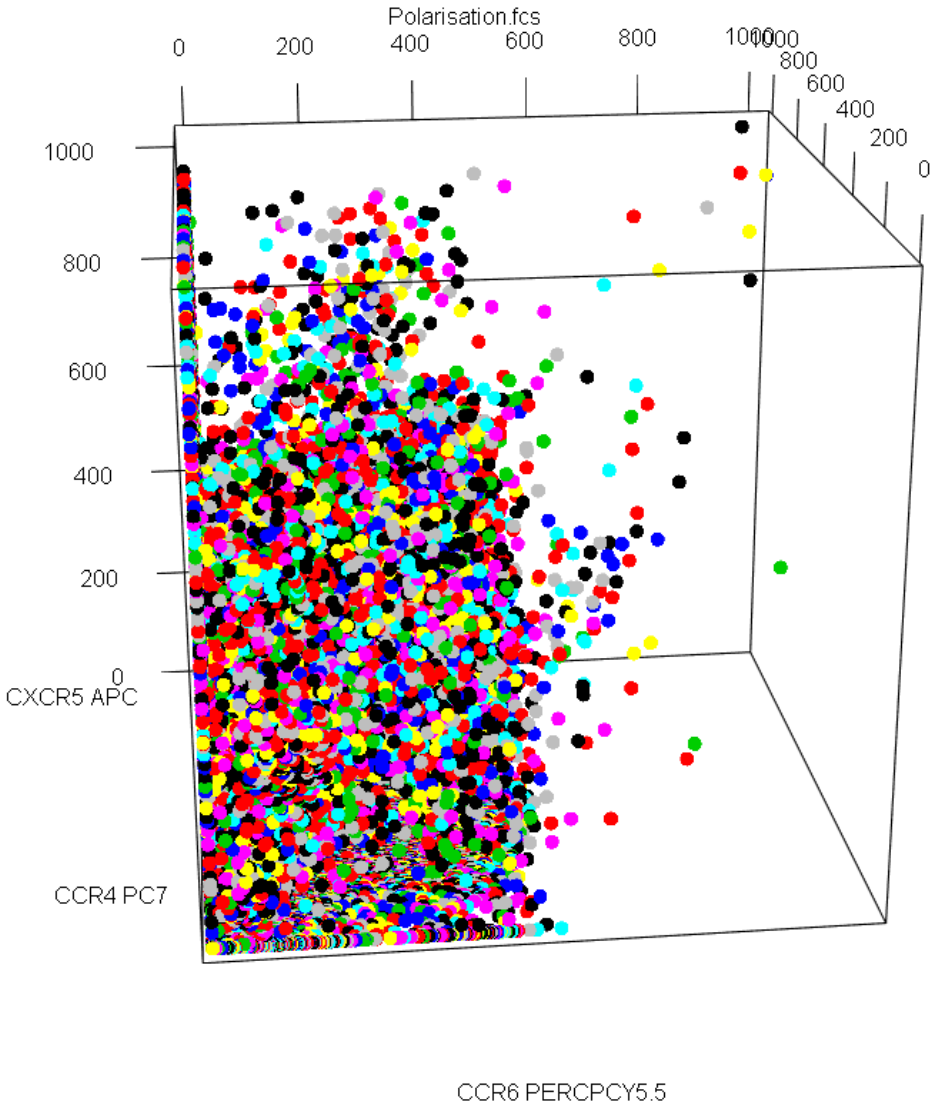

Lympho B.fcs
0
200
400
600
800
1000
1000
800
600
400
200
0
1000
800
600
400
200
0
CD5 ECD
CD10 PE
IGD FITC

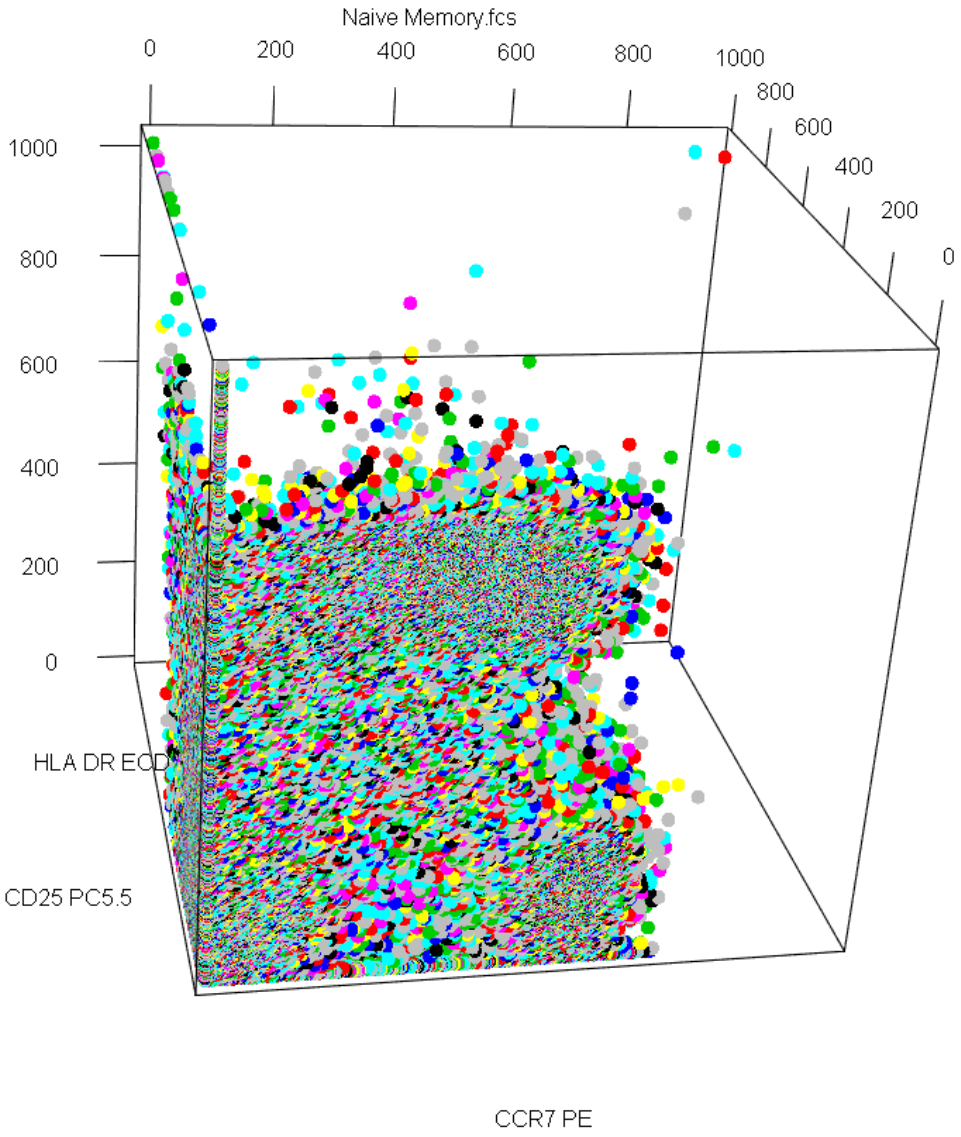

Lympho B.fcs
0
200
400
600
800
1000
1000
800
600
400
200
0
1000
800
600
400
200
0
CD10 PE
CD5 ECD
IGD FITC

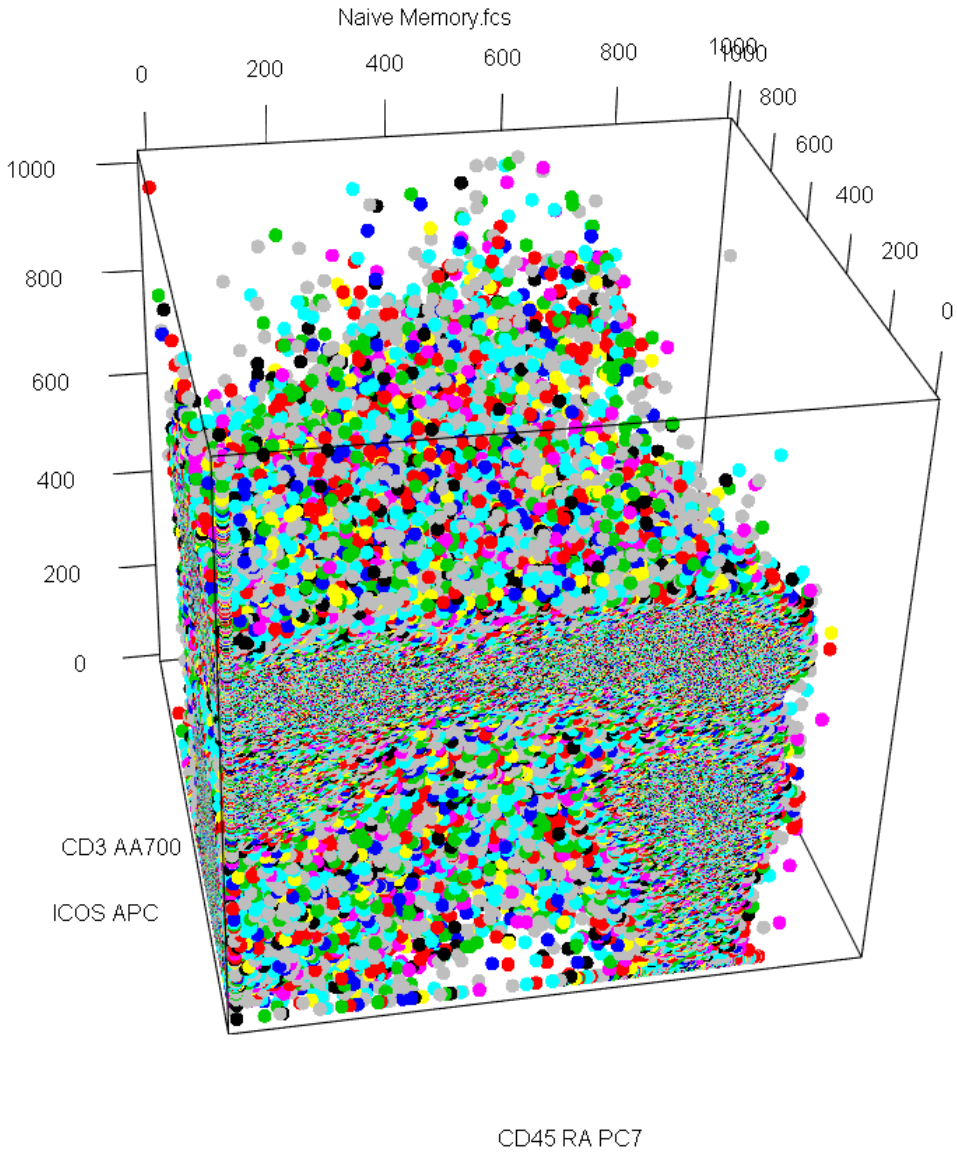

Lympho B.fcs
0
200
400
600
800
1000
1000
800
600
400
200
0
800
600
400
200
0
CD24 A750
CD27 PC5
CD38 PC7

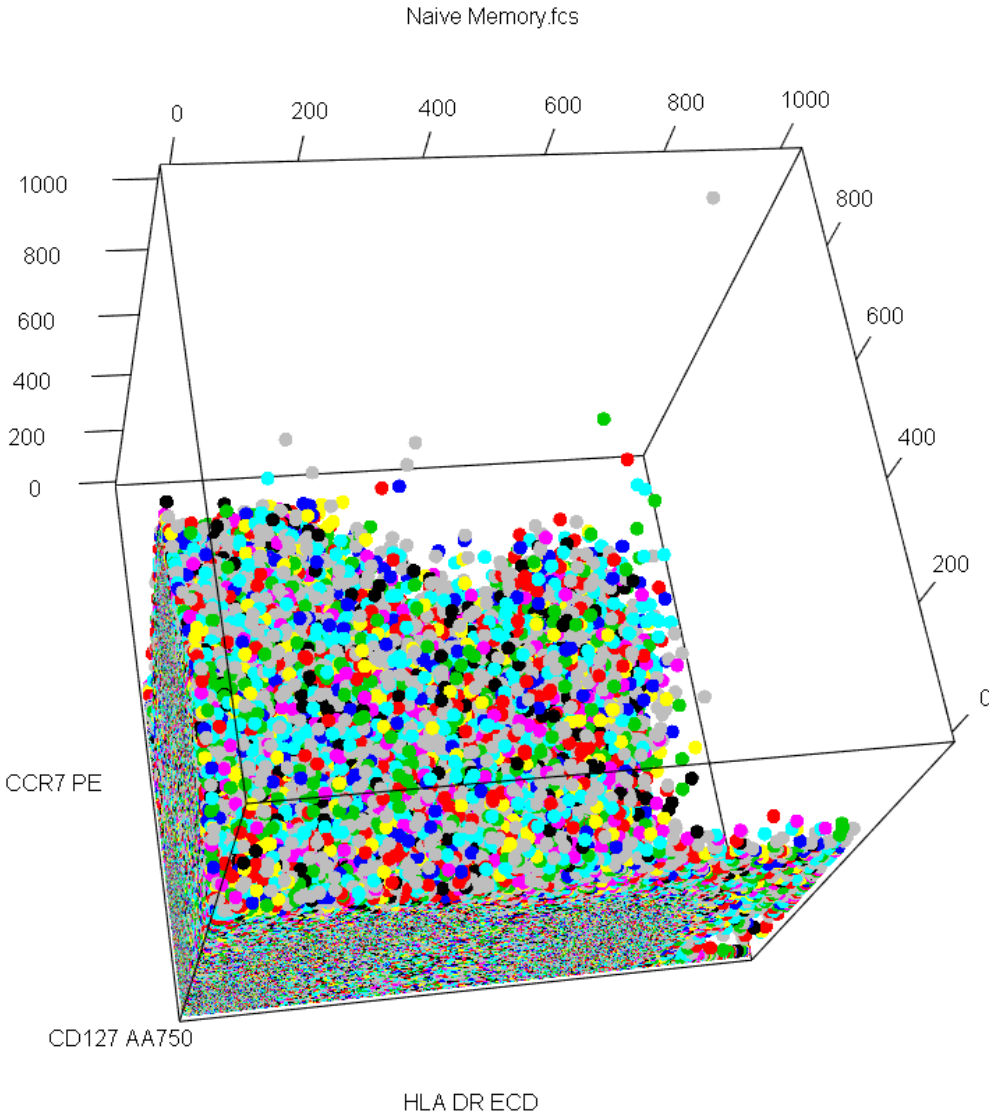

Polarisation.fcs
0
200
400
600
800
1000
1000
800
600
400
200
0
1000
800
600
400
200
0
CCR7 PE CF594
CCR10 PE
CXCR3 FITC


Polarisation.fcs
0
200
400
600
800
1000
1000
800
600
400
200
0
0
200
400
600
800
1000
CCR6 PERCPCY5.5
CCR4 PC7
CD45RA AA750


Polarisation.fcs
0
200
400
600
800
1000
1000
800
600
400
200
0
1000
800
600
400
200
0
CXCR5 APC
CCR4 PC7
CCR6 PERCPCY5.5


Naive Memory.fcs
0
200
400
600
800
1000
800
600
400
200
0
1000
800
600
400
200
0
HLA DR ECD
CD25 PC5.5
CCR7 PE


Naive Memory.fcs
0
200
400
600
800
1000
1000
800
600
400
200
0
1000
800
600
400
200
0
CD3 AA700
ICOS APC
CD45 RA PC7



Τα αποτελέσματα χρήζουν περισσότερη ανάλυση από ειδικούς ερευνητές που εργάζονται πάνω σε δεδομένα κυτταρομετρίας ροής.Τα αποτελέσματα στάλθηκαν στο Πανεπιστήμιο Πιερ και Μαρί Κιουρί της Γαλλίας το οποίο μου έδωσε τα αρχικά δεδομένα.

3.6 Επεκτάσεις για το μέλλον

Κάποιες επεκτάσεις για το μέλλον θα ήταν σίγουρα πιο ολοκληρωμένα πακέτα από το Bioconductor όπως ένα πακέτο με την διαδικασία της συσταδοποίησης που χρησιμοποίησα και κυρίως πακέτα που μπορούν να αξιοποιήσουν την παραγόμενη γνώση.Μία ακόμα επέκταση είναι ότι λόγω του υψηλού υπολογιστικού κόστους ο κώδικας να τρέχει σε ένα υπολογιστικό νέφος και η διαχείριση του να γίνεται από οποιονδήποτε προσωπικό υπολογιστή.

Αυτό έχει ήδη να ξεκινήσει να γίνεται με το εργαλείο GenePattern το οποίο υλοποιείται από το National Cancer Institute's Informatics Technology for Cancer Research program και το National Institute of General Medical Sciences των Η.Π.Α. με συνεργασία του MIT το οποίο δείχνοντας την "δύναμη" της R τρέχει κυρίως πάνω σε αυτήν όπως και σε Matlab και Java.Το πρόβλημα είναι ότι προς το παρών και εκεί είναι περιορισμένος ο αριθμός των εργαλείων για διαχείριση δεδομένων κυτταρομετρίας ροής.

3.7 Κώδικας Υλοποίησης

```
start<-function()
{
  rm(list = ls())
   library(flowMeans)
   library(flowCore)
   library(flowClust)
   library(flowDensity)
   library(flowTrans)
   library(flowStats)
   library(flowPeaks)
   library(stringr)
   library(orclus)
   library(rgl)
   str=getwd()
   source(str_c(str,"/thesis.R",""))
   source(str_c(str,"/transform.R",""))
   source(str_c(str,"/margin.R",""))
   source(str_c(str,"/QA.r",""))
   source(str_c(str,"/comp.R",""))
   source(str_c(str,"/my_normalize.R",""))
   source(str_c(str,"/clustering.R",""))
   source(str_c(str,"/lb.spillover.R",""))
   source(str_c(str,"/pol.spillover.R",""))
```

```
24    source(str_c(str,"/nv.spillover.R",""))
25    source(str_c(str,"/plot.R",""))
26  }
```

Intiliaze system

```
1   thesis <- function(graphics=TRUE,margin=TRUE,transform=TRUE,
        normalize=TRUE,clustering=TRUE) {
2     ptm <- proc.time()[3]
3     if(graphics==FALSE)
4     {
5       graphics.off()
6     }
7     ##############################
8     #       IMPORT DATA
9     ##############################
10    cat("Import Data(0/3)\n")
11    lb <<- read.FCS("Lympho B.fcs", transformation=FALSE);
12    cat("Import Data(1/3)\n")
13    nv<<- read.FCS("Naive Memory.fcs", transformation=FALSE);
14    cat("Import Data(2/3)\n")
15    pol<<- read.FCS("Polarisation.fcs", transformation=FALSE);
16    cat("Import Data(3/3)\n")
17    ##################################################
18    #           CHANGING NAMES TO COLUMNS
19    ##################################################
20    colnames(lb)<<-c("FSC-A","SSC-A","FL1-H","FL2-H","FL3-H","FL4-
          H","FL5-H","FL6-H","FL7-H","FL8-H","FL9-H","FL10-H","TIME",
          "SSC-H","SSC-W","FSC-W")
21    colnames(pol)<<-c("FSC-A","SSC-A","FL1-H","FL2-H","FL3-H","FL4
          -H","FL5-H","FL6-H","FL7-H","FL8-H","FL9-H","FL10-H","TIME"
          ,"SSC-H","SSC-W","FSC-W")
22    colnames(nv)<<-c("FSC-A","SSC-A","FL1-H","FL2-H","FL3-H","FL4-
          H","FL5-H","FL6-H","FL7-H","FL8-H","FL9-H","FL10-H","TIME",
          "SSC-H","SSC-W","FSC-W")
23    ##################################################
24    #                 Remove Time
25    ##################################################
26    lb<-lb[,-c(13)]
27    pol<-pol[,-c(13)]
28    nv<-nv[,-c(13)]
29    ##############################################
30    #               QUALITY ASSURANCE
31    ##############################################
32    if (graphics==TRUE)
33    {
```

```
34      cat("Quality Assurance(0/1)\n")
35      QA(lb,pol,nv)
36      cat("Quality Assurance(1/1)\n")
37    }
38    ################################################
39    #                DENSITY PLOT
40    ################################################
41    frames<-list("Lympho B"=lb,"Polarisation"=pol,"Naive Memory"=
          nv)
42    fs<<-as(frames,"flowSet")
43    if (graphics==TRUE)
44    {
45      cat("DensityPlot of all raw data\n")
46      d1 <- densityplot(~ ., fs, main="original")
47      png( file= "DensityPlot of all raw data.png")
48      plot(d1)
49      dev.off()
50    }
51    ###############################################
52    #             Margin Events
53    ###############################################
54    if (margin==TRUE)
55    {
56      lb.clean<<-margin(lb,graphics)
57      pol.clean<<-margin(pol,graphics)
58      nv.clean<<-margin(nv,graphics)
59      frames<-list("Lympho B"=lb.clean,"Polarisation"=pol.clean,"
            Naive Memory"=nv.clean)
60      fs<<-as(frames,"flowSet")
61    }
62    ########################################
63    #            Compensate
64    ########################################
65    if (margin==TRUE)
66    {
67      comp()
68      frames<-list("Lympho B"=lb.comp,"Polarisation"=pol.comp,"
            Naive Memory"=nv.comp)
69      fs<<-as(frames,"flowSet")
70    }
71
72    ################################################
73    #               tranform
74    ################################################
75    if (transform==TRUE)
```

```
76  {
77    cat("tranforming(0/3)\n")
78    transforming(lb.comp,graphics)
79    lb.trans<<-trans
80    cat("tranforming(1/3)\n")
81    transforming(pol.comp,graphics)
82    pol.trans<<-trans
83    cat("tranforming(2/3)\n")
84    transforming(nv.comp,graphics)
85    nv.trans<<-trans
86    cat("tranforming(3/3)\n")
87    frames<-list("Lympho B"=lb.trans,"Polarisation"=pol.trans,"
          Naive Memory"=nv.trans)
88    fs.trans<<-as(frames,"flowSet")
89    fs<<-as(frames,"flowSet")
90  }
91
92  ################################################################
93  #              DENSITY PLOT transformed data
94  ################################################################
95  if (graphics==TRUE)
96  {
97    cat("DensityPlot of all transformed data\n")
98    d1 <- densityplot(~ ., fs, main="transformed")
99    png(file="Tranformed.png")
100   plot(d1)
101   dev.off()
102 }
103 ###################################
104 #           NORMALIZE
105 ###################################
106 if (normalize==TRUE)
107 {
108   cat("normalize...\n")
109   my_normalize(fs,graphics)
110   lb.norm<<-normalized[[1]]
111   pol.norm<<-normalized[[2]]
112   nv.norm<<-normalized[[3]]
113   fs.norm<<-normalized
114   fs<<-normalized
115 }
116 ################################################################
117 #                        CLUSTERING
118 ################################################################
119 if (clustering==TRUE)
```

```
120 {
121   cat("Clustering...\n")
122   subSpace(fs)
123   cat("Clustering...OK\n")
124   cat("THE END\n")
125 }
126 total<-proc.time()[3] - ptm
127 hours<-total%/%3600
128 minutes<-(total-hours*3600)%/%60
129 secs<-round(total%%60,digits=0)
130
131 ######################################################
132 #              CHANGING NAMES TO COLUMNS
133 ######################################################
134 colnames(lb)<<-c("FSC-A","SSC-A","IGD FITC","CD10 PE","CD5 ECD
      ","CD27 PC5.5","CD38 PC7","IGM APC","CD19 AA700","CD24 A750
      ","CD21 PB","CD45 KO","TIME","SSC-H","SSC-W","FSC-W")
135 colnames(pol)<<-c("FSC-A","SSC-A","CXCR3 FITC","CCR10 PE","
      CCR7 PE CF594","CCR6 PERCPCY5.5","CCR4 PC7","CXCR5 APC","
      CD3 AA700","CD45RA AA750","CD4 PB","CD8 KO","TIME","SSC-H",
      "SSC-W","FSC-W")
136 colnames(nv)<<-c("FSC-A","SSC-A","CD95 FITC","CCR7 PE","HLA DR
       ECD","CD25 PC5.5","CD45 RA PC7"," ICOS APC","CD3 AA700","
      CD127 AA750","CD4 PB","CD8 KRO","TIME","SSC-H","SSC-W","FSC
      -W")
137 colnames(lb.clean)<<-c("FSC-A","SSC-A","IGD FITC","CD10 PE","
      CD5 ECD","CD27 PC5.5","CD38 PC7","IGM APC","CD19 AA700","
      CD24 A750","CD21 PB","CD45 KO","SSC-H","SSC-W","FSC-W")
138 colnames(pol.clean)<<-c("FSC-A","SSC-A","CXCR3 FITC","CCR10 PE
      ","CCR7 PE CF594","CCR6 PERCPCY5.5","CCR4 PC7","CXCR5 APC",
      "CD3 AA700","CD45RA AA750","CD4 PB","CD8 KO","SSC-H","SSC-W
      ","FSC-W")
139 colnames(nv.clean)<<-c("FSC-A","SSC-A","CD95 FITC","CCR7 PE","
      HLA DR ECD","CD25 PC5.5","CD45 RA PC7"," ICOS APC","CD3
      AA700"," CD127 AA750","CD4 PB","CD8 KRO","SSC-H","SSC-W","
      FSC-W")
140 colnames(lb.comp)<<-c("FSC-A","SSC-A","IGD FITC","CD10 PE","
      CD5 ECD","CD27 PC5.5","CD38 PC7","IGM APC","CD19 AA700","
      CD24 A750","CD21 PB","CD45 KO","SSC-H","SSC-W","FSC-W")
141 colnames(pol.comp)<<-c("FSC-A","SSC-A","CXCR3 FITC","CCR10 PE"
      ,"CCR7 PE CF594","CCR6 PERCPCY5.5","CCR4 PC7","CXCR5 APC","
      CD3 AA700","CD45RA AA750","CD4 PB","CD8 KO","SSC-H","SSC-W"
      ,"FSC-W")
142 colnames(nv.comp)<<-c("FSC-A","SSC-A","CD95 FITC","CCR7 PE","
      HLA DR ECD","CD25 PC5.5","CD45 RA PC7"," ICOS APC","CD3
```

```
        AA700","CD127 AA750","CD4 PB","CD8 KRO","SSC-H","SSC-W","
        FSC-W")
143   colnames(lb.trans)<-c("FSC-A","SSC-A","IGD FITC","CD10 PE","
        CD5 ECD","CD27 PC5.5","CD38 PC7","IGM APC","CD19 AA700","
        CD24 A750","CD21 PB","CD45 KO","SSC-H","SSC-W","FSC-W")
144   colnames(pol.trans)<-c("FSC-A","SSC-A","CXCR3 FITC","CCR10 PE
        ","CCR7 PE CF594","CCR6 PERCPCY5.5","CCR4 PC7","CXCR5 APC",
        "CD3 AA700","CD45RA AA750","CD4 PB","CD8 KO","SSC-H","SSC-W
        ","FSC-W")
145   colnames(nv.trans)<-c("FSC-A","SSC-A","CD95 FITC","CCR7 PE","
        HLA DR ECD","CD25 PC5.5","CD45 RA PC7","ICOS APC","CD3
        AA700","CD127 AA750","CD4 PB","CD8 KRO","SSC-H","SSC-W","
        FSC-W")
146   colnames(lb.norm)<-c("FSC-A","SSC-A","IGD FITC","CD10 PE","
        CD5 ECD","CD27 PC5.5","CD38 PC7","IGM APC","CD19 AA700","
        CD24 A750","CD21 PB","CD45 KO","SSC-H","SSC-W","FSC-W")
147   colnames(pol.norm)<-c("FSC-A","SSC-A","CXCR3 FITC","CCR10 PE"
        ,"CCR7 PE CF594","CCR6 PERCPCY5.5","CCR4 PC7","CXCR5 APC","
        CD3 AA700","CD45RA AA750","CD4 PB","CD8 KO","SSC-H","SSC-W"
        ,"FSC-W")
148   colnames(nv.norm)<-c("FSC-A","SSC-A","CD95 FITC","CCR7 PE","
        HLA DR ECD","CD25 PC5.5","CD45 RA PC7","ICOS APC","CD3
        AA700","CD127 AA750","CD4 PB","CD8 KRO","SSC-H","SSC-W","
        FSC-W")
149
150   cat("Finished at")
151   sprintf("%02d:%02d:%02d", hours, minutes,round(secs,digits=0))
152 }
```

Main code

```
1  margin<-function(f,graphics=TRUE)
2  {
3    mypath=getwd()
4    str<-f@description$GUID
5    str<-paste(str,"\n")
6    cat(str)
7    f.clean<-nmRemove(flow.frame = f,channels = c(1,2),neg = TRUE,
       talk=TRUE)
8
9    if(graphics==TRUE)
10   {
11
12     par(xpd=FALSE)
13     str<-f@description$GUID
14     str<-gsub(".fcs","",str)
```

```
15        str<-paste(mypath,str,"DensPlotClean.png",sep="/")
16        png(file = str)
17        plotDens(f.clean,c("FSC-A","SSC-A"))
18        dev.off()
19    }
20
21    return(f.clean)
22 }
```

Code for margin events

```
1  comp<-function(){
2     cat("compensate...\n")
3     comp_lb()
4     comp_pol()
5     comp_nv()
6     cat("compensate OK\n")
7  }
```

Code for compansate data

```
1  comp_lb<-function(){
2     spill<-c(0.0,8.0,2.0,1.0,0.0,0.0,0.0,0.0,0.0,1.0,
3              3.0,0.0,33.0,9.0,1.0,0.0,0.0,0.0,0.0,3.0,
4              1.0,11.0,0.0,40.0,8.0,1.0,0.0,0.0,0.0,0.0,
5              0.0,1.0,0.0,0.0,40.0,6.0,54.0,11.0,0.0,0.0,
6              0.0,1.0,0.0,0.0,0.0,0.0,1.0,8.0,0.0,0.0,
7              0.0,0.0,0.0,0.0,0.0,0.0,26.0,3.0,0.0,0.0,
8              0.0,0.0,0.0,0.0,0.0,8.0,0.0,14.0,0.0,0.0,
9              0.0,0.0,0.0,0.0,1.0,40.0,28.0,0.0,0.0,0.0,
10             0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,4.0,
11             2.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,4.0,0.0)
12
13    lb.spill<-matrix(data = spill,nrow = 10,ncol = 10)
14    colnames(lb.spill)<-c("FL1-H","FL2-H","FL3-H","FL4-H","FL5-H",
          "FL6-H","FL7-H","FL8-H","FL9-H","FL10-H")
15    lb.comp<<-compensate(lb.clean,lb.spill)
16 }
```

Helping code for Lympho B data

```
1  comp_pol<-function(){
2     spill<-c(
3       0.0,7.9,1.8,0.3,0.0,0.0,0.0,0.0,0.0,1.1,
4       2.6,0.0,33.0,7.5,1.4,0.1,0.0,0.0,0.0,2.7,
5       0.6,11.6,0.0,37.3,8.3,1.0,0.5,0.1,-0.1,0.3,
6       0.2,1.6,0.5,0.0,40.8,6.2,57.9,10.8,0.0,0.1,
```

```
      0.2,1.3,0.4,0.2,0.0,0.0,1.1,6.7,0.0,0.0,
      0.0,0.0,0.0,0.1,0.0,0.0,26.0,3.1,0.0,0.0,
      0.0,0.0,0.0,0.1,0.1,11.6,0.0,13.6,0.0,0.0,
      0.0,0.0,0.0,0.0,1.2,56.0,27.8,0.0,0.0,0.0,
      0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,4.1,
      0.5,0.0,0.0,0.0,0.0,0.2,0.1,0.0,3.8,0.0)

  pol.spill<-matrix(data = spill,nrow = 10,ncol = 10)
  colnames(pol.spill)<-c("FL1-H","FL2-H","FL3-H","FL4-H","FL5-H"
      ,"FL6-H","FL7-H","FL8-H","FL9-H","FL10-H")
  pol.comp<<-compensate(pol.clean,pol.spill)}
```

Helping code for Polarisation data

```
comp_nv<-function(){
  spill<-c(
    0.0,7.9,1.8,0.3,0.0,0.0,0.0,0.0,0.0,1.1,
    2.6,0.0,33.0,7.5,1.4,0.1,0.0,0.0,0.0,2.7,
    0.6,11.6,0.0,37.3,8.3,1.0,0.5,0.1,-0.1,0.3,
    0.2,1.6,0.5,0.0,40.8,6.2,57.9,10.8,0.0,0.1,
    0.2,1.3,0.4,0.2,0.0,0.0,1.1,6.7,0.0,0.0,
    0.0,0.0,0.0,0.1,0.0,0.0,26.0,3.1,0.0,0.0,
    0.0,0.0,0.0,0.1,0.1,11.6,0.0,13.6,0.0,0.0,
    0.0,0.0,0.0,0.0,1.2,56.0,27.8,0.0,0.0,0.0,
    0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,4.1,
    0.5,0.0,0.0,0.0,0.0,0.2,0.1,0.0,3.8,0.0)

  nv.spill<-matrix(data = spill,nrow = 10,ncol = 10)
  colnames(nv.spill)<-c("FL1-H","FL2-H","FL3-H","FL4-H","FL5-H",
      "FL6-H","FL7-H","FL8-H","FL9-H","FL10-H")
  nv.comp<<-compensate(nv.clean,nv.spill)
}
```

Helping code for Naive Memory data

```
transforming <-function(f,graphics=TRUE)
{
  mypath=getwd()
  biexpTrans <- flowJoTrans(channelRange=4096, maxValue=262144 ,
      pos=4.5,neg=0, widthBasis=-10)
  tf <- transformList(colnames(f)[3:12], biexpTrans)
  f.trans<-transform(f,tf)
  trans<<-f.trans
  str<-f@description$GUID
  str<-gsub(".fcs","",str)
  str<-paste(mypath,str,"DensPlotTrans.png",sep="/")
```

```
  if(graphics==TRUE)
  {
    png(file = str)
    par( mfrow = c(4 ,3) ,mar = c(2, 2, 2, 1))
    for (i in 3:12){
      plotDens ( f.trans , c(i,2))
    }
    dev.off()
  }

}
```

Code for transforming data)

```
my_normalize<-function(fs,graphics=TRUE)
{

  normalized<<-warpSet(fs, colnames(fs), grouping = NULL, monwrd
      = TRUE, subsample=NULL,peakNr=NULL, clipRange=0.01,
     nbreaks=11, bwFac=2,warpFuns=FALSE,target=NULL)
  cat("normalize...Done\n")
  if(graphics==TRUE)
  {
    cat("Let's have some pretty plots\n")
    cat("DensityPlot of all normalized data\n")
    d2 <- densityplot(~ ., normalized, main="normalized")
    png(file="normalized.png")
    plot(d2)
    dev.off()
  }

}
```

Code for normalize data

```
subSpace<-function(x)
{
  cat("Clustering...(0/3)\n")
  lb.clust<<-orclus(x = x[[1]]@exprs, k = 50, l = 10, k0 = 60,
      verbose = FALSE)
  cat("Clustering...(1/3)\n")
  pol.clust<<-orclus(x = x[[2]]@exprs, k = 50, l = 10, k0 = 60,
      verbose = FALSE)
  cat("Clustering...(2/3)\n")
  nv.clust<<-orclus(x = x[[3]]@exprs, k = 50, l = 10, k0 = 60,
      verbose = FALSE)
```

```
    cat("Clustering...(3/3)\n")
}
```

Code for subspace clustering

```
plot3D.clusters<-function(frame,clust,x,y,z,size=5,main="")
{
plot3d(x=frame[,c(x)]@exprs,y=frame[,c(y)]@exprs,z=frame[,c(z)]
    @exprs,col=clust$cluster,xlab=colnames(frame[,c(x)]),ylab=
    colnames(frame[,c(y)]),zlab=colnames(frame[,c(z)]),size=size,
    main=main)
}
```

Code for ploting

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Shapiro HM. Practical flow cytometry. 4th ed. New York, NY:Wile y-Liss;2003.

[2] Givan AL. Principles of flow cytometry: an overview. Methods Cell Biol. 2001;63:19-50.

[3] Seamer LC, Bagwell CB, Barden L, Redelman D, Salzman GC, Wood JC,Murphy RF. Proposed new data file standard for flow cytometry, version FCS 3.0. Cytometry 1997; 28:118–122.

[4] Watson JV. Flow cytometry data analysis: basic principles and statistics .Cambridge , Cambridge University Press; 1992.

[5] Brown M, Wittwer CT. Flow cytometry: principles and clinical applications in hematology. Clinical Chemistry 2000; 46:1221-9.

[6] Riley RS, Idowu M. Principles and applications of flow cytometry. http://www.pathology.vcu.edu/education/lymph/documents/Flow.pdf

[7] Bouckaert R. Naïve Bayes classifiers that perform well with continuous variables.In Proc. of AI 2004; 1089-94

[8] Breiman L, Friedman JH, Olshen RA, Stone CJ. Classification and regression trees . Chapman & Hall 1984.

[9] Burges JC. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 1998; 2(2):121-67.

[10] Carpenter G, Grossberg S. A massively parallel architecture for a selforganizing neural pattern recognition machine . Computer Vision, Graphics and Image Processing 1987; 37:54-115.

[11] Carpenter G, Grossberg S. Stable self-organization of pattern recognition codes for analog input patterns . Applied Optics 1987; 26:4919-30.

[12] Cheung Y-M. k*-Means : A new generalized k-means clustering algorithm.Pattern Recognition Letters 2003; 24:2883–93.

[13] Estivill-Castro V, Yang J. Fast and robust general purpose clustering algorithms . In Proc. of PRICAI 2000; 208–18.

[14] Fraley C, Raftery A. Model-based clustering, discriminant analysis, and density estimation. Journal of the American Statistical Association 2002; 97:611-31.

[15] Friedman N, Goldszmidt M. Discretizing continuous attributes wOEΠe learning Bayesian networks . In Proc. of ICML 1996; 157-65.

[16] Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers .Machine Learning 1997; 29(2-3):131-63.

[17] Handl J, Knowles J, Kell DB. Computational cluster validation in postgenomic data analysis. Bioinformatics 2005; 21(15):3201-12.

[18] Heckerman D, Geiger D, Chickering DM. Learning Bayesian networks : The combination of knowledge and statistical data . Machine Learning 1995; 20:197–243.

[19] Hoppner F, Klawonn F, Kruse R. Fuzzy cluster analysis : Methods for classification, data analysis, and image recognition. Wile y 1999.

[20] Kohonen T. The self-organizing map. In Proc. of IEEE 1990; 78(9):1464–80.

[21] Liu H, Yu L. Toward integrating feature selection algorithms for classification and clustering. IEEE Transactions on Knowledge and Data Engineering 2005; 17(4):491-502.

[22] Lowd D, Domingos P. Naive Bayes models for probability estimation. In Proc. of ICML 2005; 529-36

[23] Martinez-Arroyo M, Sucar LE. Learning an optimal naïve Bayes classifier. In Proc. of ICPR 2006; 3:1236-9.

[24] Neapolitan RE. Learning Bayesian networks . Prentice Hall 2004.

[25] Pelleg D, Moore A. Accelerating exact k-means algorithms with geometric reasoning. In Proc. of ACM SIGKDD 1999; 277-81.

[26] Pelleg D, Moore A. X-means: Extending k-means with efficient estimation of the number of clusters . In Proc. of ICML 2000; 727-34.

[27] Tan PN, Steinbach M, Kumar V. Introduction to data mining. Addison-Wesley 2005.

[28] Xu R, Wunsch D. Survey of clustering algorithms . IEEE Transactions on Neural Networks 2005; 16:645-78

[29] Zhang G. Neural networks for classification: a survey. IEEE Transactions on Systems , Man, and Cybernetics , Part C 2000; 30(4): 451-62.

[30] J.O. Ramsay and B.W. Silverman: Applied Functional Data Analysis, Springer 2002

[31] Aggarwal, C. and Yu, P. (2000): Finding generalized projected clusters in high dimensional spaces, Proceedings of ACM SIGMOD International Conference on Management of Data, pp. 70-81.

[32] Μεταπτυχιακή Εργασία Σακελλαράκη Παναγιώτα 2007: Μελέτη της κυτταρικής ανασοαπόκρισης στην ιδιοπαθή θρομβοπενική πορφύρα.

[33] Athanasia Mouzaki, Maria Theodoropoulou, Ioannis Gianakopoulos, Vassiliki Vlaha, Maria-Christina Kyrtsonis, and Alice Maniatis. Expression patterns of Th1 and Th2 cytokine genes in childhood idiopathic thrombocytopenic purpura (ITP) at presentation and their modulation by intravenous immunoglobulin G (IVIg) treatment: their role in prognosis. Blood.2002;100:1774-1779

[34] John W. Semple. T cell and cytokine abnormalities in patients with autoimmune thrombocytopenic purpura. Transfusion and Apheresis Science.2003; 28:237–242

[35] Paul Imbach. Development and Research in Idiopathic Thrombocytopenic Purpura: An Inflammatory and Autoimmune Disorder. Pediatr Blood Cancer. 2006;47:685–686

[36] Gruel Y, Boizard B, Daffos F, Forestier F, Caen J, Wautier JL. Determination of platelet antigens and glycoproteins in the human fetus. Blood 1986;68:488–92.

[37] BD Biosciences. Introduction to flow cytometry: A learning Guide, Manual Part, Number1: 1-11032-01.

[38] Γκριτζάπης Α. Κυτταρικός διαχωρισμός. Βιβλίο πρακτικών 3ου Πανελλήνιου συνεδρίου κυτταρομετρίας:15.

[39] Herzenberg,L.A.,Parks,D.,Sahaf,B.,Perez,O.,Roederer,M.,Herzenberg,L. A.The History and Future of the Fluorescence Activated Cell Sorter and Flow Cytometry: A View from Stanford, Clinical Chemistry,2002; 48(10):1819-1827.

[40] Nicole Baumgarth , Mario Roederer. A practical approach to multicolor flow cytometry for Immunophenotyping. Journal of Immunological Methods 2000;243:77–97

[41] Graeme V. Chapman.Instrumentation for flow cytometry Journal of Immunological Methods 2000;243:3–12

[42] Heddy Zola. Immunological applications of flow cytometry. Journal of Immunological Methods 2000;243:1–2

[43] Guenther Boeck. Current Status of Flow Cytometry in Cell and Molecular Biology. International Review of Cytology,2000;204:239-298.