ΟΜΑΔΟΠΟΙΗΣΗ ΔΕΔΟΜΕΝΩΝ ΥΨΗΛΗΣ ΔΙΑΣΤΑΣΗΣ

Η
ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ ΕΞΕΙΔΙΚΕΥΣΗΣ

Υποβάλλεται στην

ορισθείσα από την Γενική Συνέλευση Ειδικής Σύνθεσης
του Τμήματος Πληροφορικής
Εξεταστική Επιτροπή

από τον

ΣΤΕΦΑΝΟΣ – ΚΩΝΣΤΑΝΤΙΝΟΣ ΠΟΥΡΣΑΛΙΔΗΣ

ως μέρος των Υποχρεώσεων

για τη λήψη

του

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ

ΜΕ ΕΞΕΙΔΙΚΕΥΣΗ ΣΤΙΣ ΤΕΧΝΟΛΟΓΙΕΣ-ΕΦΑΡΜΟΓΕΣ

Νοέμβριος, 2005

Η παρούσα εργασία είναι το αποτέλεσμα της δραστηριότητάς μου κατά το δεύτερο έτους των μεταπτυχιακών μου σπουδών στο Τμήμα Πληροφορικής του Πανεπιστημίου Ιωαννίνων.

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή, κ. Αριστείδη Λύκα, που λόγω της υπομονής και καθοδήγησής του μπόρεσα να εκπονήσω αυτή την εργασία.

Επίσης, θέλω να ευχαριστήσω τους συναδέλφους μου, Κωνσταντίνο Κωνσταντινόπουλο και Δημήτρη Τζίκα για την βοήθεια που μου προσέφεραν κατά τη διάρκεια των μεταπτυχιακών μου σπουδών τόσο ηθικά όσο και σε περιπτώσεις όπου χρειάστηκα την δική τους εμπειρία στον τομέα της Μηχανικής Μάθησης.

Τέλος, θέλω να ευχαριστήσω τους γονείς μου πού όλα αυτά τα χρόνια με στήριξαν και με καθοδήγησαν με τις συμβουλές και την αγάπη τους και ελπίζω να ανταποδίδω λίγο από αυτήν τώρα.

Πρόλογος

Σκοπός αυτής της μεταπτυχιακής εργασίας εξειδίκευσης, που εκπονήθηκε στο Πανεπιστήμιο Ιωαννίνων κατα τη διάρκεια του ακαδημαϊκού έτους 2004/2005 υπο την επίβλεψη του κ. Αριστείδη Λύκα, είναι να μελετήσει τρόπους ομαδοποίησης δεδομένων υψηλής διάστασης. Αυτό το πρόβλημα παρουσιάζει ιδιαίτερο ενδιαφέρον, λόγω της λεγόμενης "κατάρας" της υψηλής διάστασης (curse of dimensionality) η οποία οδηγεί σε αποτυχία πολλούς κλασσικούς αλγορίθμους ομαδοποίησης.

Θα δούμε, διάφορες τεχνικές ομαδοποίησης δεδομένων υψηλής διάστασης και θα συγκρίνουμε τα αποτελέσματά μας με άλλους κλασσικούς αλγορίθμους σε συγκεκριμμένα και μάλιστα σε πολλές περιπτώσεις πραγματικά δεδομένα βιοπληροφορικής που έχουν το χαρακτηριστικό αυτό της υψηλής διάστασης.

Ιωάννινα, Νοέμβριος 2005

Στέφανος – Κωνσταντίνος Πουρσαλίδης

Περίληψη

Στην παρούσα εργασία, το θέμα που μας απασχολεί είναι η ομαδοποίηση δεδομένων υψηλής διάστασης. Γενικότερα το πρόβλημα της ομαδοποίησης δεδομένων βρίσκει πολλές εφαρμογές σε ποικίλους και φαινομενικά άσχετους μεταξύ τους τομείς, όπως της ιατρικής, της στατιστικής, της οικονομίας, της ψυχολογίας και άλλους ακόμα τομείς. Στο πρόβλημα αυτό, μας δίνεται ένα σύνολο X,

$$X = \{x_1, x_2, ..., x_N\}, x_i \in \mathbb{R}^d$$

που αποτελείται απο N δεδομένα διάστασης d. Σκοπός μας σε ένα πρόβλημα ομαδοποίησης δεδομένων είναι να χωρίσουμε τα δεδομένα του συνόλου X (συνήθως αναφερόμαστε σε αυτά με τον πιο εύστοχο όρο πρότυπα) σε ενα πλήθος k ομάδων, έτσι ώστε πρότυπα που ανήκουν στην ίδια ομάδα να είναι όμοια μεταξύ τους.

Ειδικά, όταν τα πρότυπα που εμφανίζονται στο σύνολο δεδομένων X, έχουν μεγάλη διάσταση, μιλάμε για προβλήματα ομαδοποίησης δεδομένων υψηλής διάστασης και αποτελεί μια περαιτέρω πρόκληση στον τομέα της Μηχανικής Μάθησης. Προβλήματα ομαδοποίησης δεδομένων υψηλής διάστασης συναντάμε σε όλους τους παραπάνω τομείς που αναφέραμε και επιπλέον, σχετικά πρόσφατα, σε έναν υπο εξέλιξη επιστημονικό τομέα που λέγεται Βιοπληροφορική και αντιμετωπίζει το πρόβλημα της ομαδοποίησης για υψηλής διάστασης δεδομένα που προέρχονται απο διάφορες μετρήσεις σε αλυσίδες του DNA.

Η τεράστια διάσταση τέτοιων προβλημάτων σε συνδυασμό με τα λίγα δεδομένα που μπορεί κανείς να συγκεντρώσει (σε σχέση πάντα με την διάσταση του προβλήματος που ανιμετωπίζει κανείς) σε τέτοιες περιπτώσεις, μας αναγκάζουν, σε συνδυασμό με αλγορίθμους ομαδοποίησης δεδομένων, να κάνουμε χρήση αλγορίθμων μείωσης της διάστασης των δεδομένων.

Αναλυτικά, λοιπόν στην εργασία αυτή και στο πρώτο Κεφάλαιο θα κάνουμε μια εισαγωγή στο πρόβλημα της ομαδοποίησης δεδομένων υψηλής διάστασης και θα περιγράψουμε τις τυχαίες προβολές ως μέθοδο μείωσης της διάστασης δεδομένων. Στη συνέχεια θα την αντιπαραθέσουμε με μια άλλη γνωστή και ευρέως διαδεδομένη μέθοδο μείωσης της διάστασης, που είναι η μέθοδος PCA.

Στο δεύτερο Κεφάλαιο, θα παρουσιάσουμε και θα αναλύσουμε την έννοια και την φιλοσοφία των cluster ensembles, που αποτελεί επίσης μια νέα μεθοδολογία στην αντιμετώπιση προβλημάτων ομαδοποίησης δεδομένων. Θα δούμε γιατί αυτή η μέθοδος σε συνδυασμό με τις τυχαίες προβολές μπορεί να δώσει καλά αποτελέσματα ομαδοποίησης με μικρό σχετικά υπολογιστικό κόστος.

Στο τρίτο Κεφάλαιο, περιγράφουμε αναλυτικά τους αλγορίθμους που επιλέξαμε να υλοποιήσουμε. Θα δούμε συγκεκριμμένα, ποιους αλγορίθμους χρησιμοποιήσαμε, σε καθένα απο τα δύο βήματα των cluster ensembles.

Στο τέταρτο Κεφάλαιο, παραθέτουμε συγκεκριμμένα πειραματικά αποτελέσματα, τόσο για τεχνητά όσο και πραγματικά δεδομένα βιοπληροφορικής, που χρησιμοποιήσαμε για να αξιολογήσουμε τις μεθόδους ομαδοποίησης που υλοποιήσαμε.

Τέλος, στο πέμπτο Κεφάλαιο, παραθέτουμε κάποια συμπεράσματα που μπορούμε να εξάγουμε με βάση την παρούσα έρευνα και δίνουμε τις κατευθύνσεις που μπορεί να ακολουθήσει κανείς για μελλοντική εργασία πάνω σε αυτό το θέμα.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

ΕΥΡΕΤΗΡΙΟ ΣΧΗΜΑΤΩΝ

ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

1.1 Γενικά – Ορισμός προβλήματος

Ένα πρόβλημα ιδιαίτερης σημασίας στον τομέα της Μηχανικής Μάθησης είναι αυτό της ομαδοποίησης δεδομένων με βάση τα χαρακτηριστικά τους (data clustering). Σε αυτό το πρόβλημα μας δίνεται ένα σύνολο N δεδομένων $X = \{x_1, x_2, \ldots, x_N\}$, όπου x_i ($i = 1,\ldots,N$) είναι ένα διάνυσμα d χαρακτηριστικών (ή μεταβλητών), και καλούμαστε να διαμερίσουμε το σύνολο X σε k ομάδες έτσι ώστε τα δεδομένα της ίδιας ομάδας να μοιράζονται παρόμοιες ιδιότητες. Λύση σε τέτοια προβλήματα αναζητούμε πολύ συχνά σε προβλήματα οικονομίας, στατιστικής, ιατρικής καθώς και σε πολλούς ακόμη τομείς.

Στην παρούσα εργασία θα ασχοληθούμε με τέτοιου είδους προβλήματα και συγκεκριμμένα με την περίπτωση που έχουμε "πολλά" χαρακτηριστικά ή μετρήσεις για κάθε δεδομένο, δηλαδή θα δούμε πως ομαδοποιούμε δεδομένα υψηλής διάστασης. Για παράδειγμα, ένα από τα σύνολα δεδομένων που έχουμε χρησιμοποιήσει για πειραματισμό στην εργασία αυτή, αναπαριστά χειρόγραφα ψηφία. Το σύνολο αυτό έχει 76 χαρακτηριστικά και 2000 πρότυπα, 200 για κάθε ένα από τα ψηφία, δηλαδή έχουμε σύμφωνα με τα παραπάνω N = 2000 και d = 76.

Τι σημαίνει όμως υψηλή διάσταση; Ακόμα και διάσταση 10 θεωρείται υψηλή, αλλά συνήθως εννοούμε περιπτώσεις όπου είτε η διάσταση του προβλήματος είναι μεγάλη είτε το πλήθος των δεδομένων σε σχέση με την διάσταση είναι μικρό. Τότε παρουσιάζεται και η λεγομένη "κατάρα" της μεγάλης διάστασης (curse of dimensionality [2]).

Για να δώσουμε ενα παράδειγμα της επιρροής της διάστασης στη λύση ενός προβλήματος σκεφτείτε το εξής. Έστω ότι, έχουμε ένα πλήθος χαρακτηριστικών x_i, και θέλουμε να προβλέψουμε με βάση αυτά μια απεικόνισή τους σε μια τιμή y. Έχουμε, έστω ένα πλήθος από N πρότυπα, μαζί με τις αντίστοιχες απεικονίσεις τους,

που αποτελεί το σύνολο εκπαίδευσης. Αρχίζουμε, διαμερίζοντας κάθε χαρακτηριστικό x_i σε ενα πλήθος υποδιαστημάτων, έτσι ώστε να μπορούμε να προσεγγίσουμε την τιμή ενός χαρακτηριστικού με βάση το υποδιάστημα στο οποίο ανήκει. Έτσι, όσο πιο μικρές διαμερίσεις έχουμε, θα έχουμε καλύτερη προσέγγιση στην τιμή μιας μεταβλητής, αυξάνοντας όμως έτσι και το πλήθος των υποδιαστημάτων και συνεπώς τη διάσταση του προβλήματός μας. Έτσι, κάθε πρότυπο από το σύνολο εκπαίδευσης αντιστοιχεί σε ενα σημείο στον χώρο που μόλις δημιουργήσαμε, και έχει μια αντίστοιχη τιμή y. Αν μας δοθεί τώρα ενα νέο σημείο σε αυτό τον χώρο, μπορούμε να προβλέψουμε την τιμή y που θα έχει απο τον μέσο όρο των τιμών y των σημείων του συνόλου εκπαίδευσης που υπάρχουν σε εκείνο το χώρο. Αυτή η μέθοδος, έχει όμως ενα σημαντικό πρόβλημα. Αν κάθε χαρακτηριστικό έχει χωριστεί σε έστω M υποδιαστήματα, τότε θα έχουμε (αν υποθέσουμε οτι έχουμε d χαρακτηριστικά) συνολικά M^d υποδιαστήματα, που για να προβλέψουμε την τιμή ενός νέου προτύπου που θα πέσει εκεί μέσα χρειαζόμαστε τουλάχιστον άλλο ενα πρότυπο του συνόλου εκπαίδευσης που θα υπάρχει εκεί. Έτσι, για ενα σύνολο εκπαίδευσης διάστασης d, θα απαιτούνται M^d πρότυπα ώστε να έχουμε μια ικανοποιητική προσέγγιση στην πρόβλεψη τιμών y των άγνωστων προτύπων.

Αν είμαστε αναγκασμένοι να δουλέψουμε με περιορισμένο πλήθος δεδομένων, που στην πράξη αυτό συμβαίνει, τότε η αύξηση της διάστασης ενός προβλήματος, γρήγορα μας οδηγεί στο σημείο όπου τα δεδομένα είναι πολύ λίγα για να ανταποκριθούν στις απαιτήσεις μας [2] (κατάρα της μεγάλης διάστασης).

Επίσης, ακόμα και αν είχαμε ικανοποιητικό πλήθος δεδομένων, ο χρόνος που χρειάζονται οι κλασσικοί αλγόριθμοι ομαδοποίησης (π.χ. kmeans, EM [6]) για να λύσουν τέτοια μεγάλα προβλήματα είναι τις περισσότερες φορές απαγορευτικός. Έτσι, πρέπει να καταφύγουμε σε μια άλλη προσέγγιση, την λεγόμενη "μείωση της διάστασης" (dimensionality reduction). Σε αυτή την προσέγγιση η λύση του προβλήματος έρχεται σε δυο στάδια. Στο πρώτο προβάλλουμε τα δεδομένα σε μια μειωμένη διάσταση και έπειτα στο δεύτερο βήμα με τη βοήθεια ενός αλγορίθμου ομαδοποίησης δεδομένων ομαδοποιούμε τις προβολές των δεδομένων.

Αλγόριθμοι, που θα δούμε στα πλαίσια αυτής της εργασίας, για μείωση της διάστασης είναι ο πολύ γνωστός και ευρέως χρησιμοποιούμενος Principal Component Analysis (PCA) και ο σχετικά νέος των "τυχαίων προβολών" Random Projections (RP).

1.2 Εισαγωγή στις "τυχαίες προβολές" και σύγκριση με την μέθοδο PCA

Καταρχήν να εξηγήσουμε τι εννοούμε οταν λέμε οτι θέλουμε να μειώσουμε την διάσταση ενός προβλήματος. Έστω, ενα σύνολο δεδομένων X, με N δεδομένα διάστασης d, δηλαδή:

$$X = \{x_1, x_2, ..., x_N\}, x_i \in \mathbb{R}^d, i = 1, 2, ..., N$$

Θεωρούμε ότι, τα δεδομένα x_i αποτελούν τις γραμμές ενός πίνακα διαστάσεων $N \times d$. Γραμμική προβολή σε διάσταση $d' < d$, σημαίνει πολλαπλασιασμός του πίνακα X με έναν πίνακα προβολής P (διάστασης $d \times d'$), έτσι ώστε να προκύψει ο πίνακας X':

$$X' = X \cdot P, X' \in \mathbb{R}^{N \times d'}$$

Τώρα έχουμε τα δεδομένα X' σε μια μειωμένη διάσταση d'. Αυτό που θέλουμε απο τον πίνακα προβολής P, είναι να διατηρεί τις αποστάσεις, δηλαδή δεδομένα που ήταν μακριά μεταξύ τους στον πίνακα X να εξακολουθούν να είναι μακριά και στον πίνακα X' και, όμοια, δεδομένα που σχημάτιζαν ομάδα στην αρχική διάσταση d να συνεχίσουν να είναι κοντά μεταξύ τους στην διάσταση d'. Αν ο πίνακας προβολής P ικανοποιεί αυτές τις υποθέσεις, τότε η ομαδοποίηση που θα πάρουμε με κάποιον αλγόριθμο στο σύνολο X' θα είναι παρόμοια με την ομαδοποίηση που θα παίρναμε στο αρχικό σύνολο X. Το κέρδος από αυτήν την διαδικασία είναι ότι μειώνοντας τη διάσταση, ο αλγόριθμος είναι σε θέση να ομαδοποιήσει πολύ πιο αποτελεσματικά και γρήγορα τα δεδομένα που προκύπτουν.

Η μέθοδος PCA, αναζητά τις κατευθύνσεις που συγκεντρώνουν τη μεγαλύτερη διακυμάνση και προβάλοντας σε αυτές διατηρεί το μεγαλύτερο ποσοστό της διακύμανσης των δεδομένων, ενώ αγνοώντας κατευθύνσεις με μικρή διακύμανση, θεωρεί οτι αγνοεί ασήμαντα χαρακτηριστικά. Η μέθοδος PCA είναι η βέλτιστη γραμμική μέθοδος ως προς το σφάλμα ανακατασκευής των δεδομένων που έχει προβάλει. Έχει εφαρμοστεί με επιτυχία σε πολλές περιπτώσεις και είναι ίσως ο πιο διαδεδομένος αλγόριθμος γραμμικής προβολής.

Για να έχει επιτυχία ένας αλγόριθμος προβολής, το κριτήριο που βελτιστοποιεί, πρέπει να είναι τέτοιο που να ταιριάζει στα δεδομένα. Έτσι, ένας αλγόριθμος μείωσης διάστασης που δουλεύει ικανοποιητικά για κάποιο σύνολο

δεδομένων, μπορεί σε ένα άλλο σύνολο να αποτυγχάνει εντελώς, επειδή το κριτήριο που βελτιστοποιεί στην μια περίπτωση αντιπροσωπεύει την δομή των δεδομένων, ενώ στην άλλη όχι.

Σε αντίθεση με τον PCA, ο αλγόριθμος των "τυχαίων προβολών" (απο εδώ και στο εξής RP), δεν έχει κάποιο κριτήριο που να βελτιστοποιεί. Για την ακρίβεια δεν λαμβάνει υπόψη τα δεδομένα που θέλουμε να προβάλουμε. Το μόνο που απαιτείται για να κατασκευάσουμε τον πίνακα προβολής P, είναι η αρχική και η τελική διάσταση. Αυτό ακριβώς είναι και το στοιχείο που κάνει τους RP-πίνακες προβολής ενδιαφέροντες. Απο τη μία η βέλτιστη, απο άποψη πολυπλοκότητας, ταχύτητα με την οποία κατασκευάζονται, λόγω του ότι ένας RP-πίνακας κατασκευάζεται σε χρόνο γραμμικό ως προς το γινόμενο των διαστάσεων ($\Theta(d \times d')$). Το μόνο στοιχείο απο το οποίο εξαρτάται είναι η αρχική και η τελική διάσταση. Ενδιαφέρον είναι το γεγονός ότι η μείωση διάστασης ενός προβλήματος με RP-πίνακες οδηγεί σε προβλήματα όπου τα δεδομένα ομαδοποιούνται πολλές φορές πολύ αποτελεσματικά, παρότι οι τυχαίες προβολές δεν έχουν κάποιο κριτήριο το οποίο βελτιστοποιείται κατά την εφαρμογή τους. Επίσης, ένας παράγοντας που είναι υπέρ των τυχαίων προβολών σε σχέση με την μέθοδο PCA, είναι ότι εφαρμόζεται πάντα, σε οποιαδήποτε διάσταση και αν θέλουμε να προβάλουμε τα δεδομένα.

Τελικά, ακριβώς το γεγονός το ότι δεν έχουν κάποιο κριτήριο βελτιστοποίησης τις κάνει τόσο γενικές και εφαρμόσιμες σε πολλά προβλήματα. Αυτά τεκμηριώνονται απο κάποια θεωρήματα τα οποία και παραθέτουμε χωρίς αποδείξεις. Επίσης, παρακάτω δίνουμε τους πιο συνηθισμένους τρόπους κατασκευής RP-πινάκων. Οι πίνακες προβολής κατασκευάζονται με στοχαστικό τρόπο σε κάθε περίπτωση.

1.3 Μείωση διάστασης με τυχαίες προβολές

Όπως είπαμε και παραπάνω, ο αλγόριθμος μείωσης διάστασης δεδομένων μέσω των τυχαίων προβολών, δεν λαμβάνει υπόψη του τα δεδομένα που προβάλει, διότι δεν βελτιστοποιεί κάποιο κριτήριο σχετικό με τα δεδομένα. Η μόνη είσοδος στον αλγόριθμο αυτό είναι η αρχική διάσταση d των δεδομένων, μαζί με την διάσταση d' στην οποία θέλουμε να προβληθούν. Με αυτά τα δεδομένα ως είσοδο, ο αλγόριθμος

των τυχαίων προβολών κατασκευάζει εναν $P(d \times d')$ πίνακα τυχαίων προβολών με στοιχεία p_{ij} που δημιουργούνται με έναν απο τους παρακάτω τρεις τρόπους [7].

- $p_{ij} = \begin{cases} +1, \mu\varepsilon\ \pi\iota\theta\alpha\nu\acute{o}\tau\eta\tau\alpha\ 0.5 \\ -1, \mu\varepsilon\ \pi\iota\theta\alpha\nu\acute{o}\tau\eta\tau\alpha\ 0.5 \end{cases}$
- $p_{ij} = \begin{cases} \sqrt{3}\cdot(+1), \mu\varepsilon\ \pi\iota\theta\alpha\nu\acute{o}\tau\eta\tau\alpha\ 1/6 \\ \sqrt{3}\cdot(-1), \mu\varepsilon\ \pi\iota\theta\alpha\nu\acute{o}\tau\eta\tau\alpha\ 1/6 \\ 0, \qquad \mu\varepsilon\ \pi\iota\theta\alpha\nu\acute{o}\tau\eta\tau\alpha\ 2/3 \end{cases}$
- $p_{ij} \sim N(0,1)$, i.i.d (όπου $N(0,1)$ η τυπική κανονική κατανομή)

Στην βιβλιογραφία δεν υπάρχει μέχρι τώρα κάποια σύκριση των παραπάνω τρόπων κατασκευής του πίνακα προβολής, σχετικά με το ποιος τρόπος υπερτερεί έναντι των άλλων με βάση κάποιο κριτήριο. Εμείς, στα επόμενα χρησιμοποιούμε τον τρίτο τρόπο, δηλαδή τα στοιχεία του πίνακα προβολής τα παίρνουμε απο μια τυπική κανονική κατανομή.

Σημειώνουμε οτι οι στήλες των πινάκων προβολής που κατασκευάζονται με έναν απο τους παραπάνω τρόπους, δεν κανονικοποιούνται. Μπορούμε πειραματικά να πειστούμε οτι οι στήλες πινάκων που προκύπτουν κατα αυτόν τον τρόπο, είναι (με μικρή απόκλιση) ορθοκανονικές [1]. Αποδεικνύεται, οτι οι παραπάνω πίνακες προβολής διατηρούν τις αποστάσεις των δεδομένων με μεγάλη πιθανότητα αν επιλέξουμε σωστά την διάσταση στην οποία προβάλλουμε, σύμφωνα με το επόμενο Λήμμα.

<u>Λήμμα</u>:

Δοθέντων N σημείων στον χώρο $\mathbb{R}^d$

διαλέγουμε $\varepsilon,\beta > 0$ και $q \geq \dfrac{4+2\cdot\beta}{\left(\dfrac{\varepsilon^2}{2}-\dfrac{\varepsilon^3}{3}\right)}\ln(N)$, και θέτουμε $P' = \dfrac{1}{\sqrt{q}}\cdot X \cdot P$,

όπου P είναι ένας πίνακας προβολής του οποίου τα στοιχεία προκύπτουν απο μια τυπική κανονική κατανομή. Τότε η προβολή αυτή διατηρεί τις αποστάσεις σε βαθμό $1\pm\varepsilon$ για όλα τα σημεία του X με πιθανότητα $\left(1-\mathrm{N}^{-\beta}\right)$ [7]. □

Απόδειξη του παραπάνω Λήμματος, που αναφέρεται και ως θεώρημα των Jonhnson και Lindenstrauss, δίνεται στο [5]. Το κέρδος αυτού του Λήμματος είναι ότι μπορούμε να υπολογίσουμε εκ των προτέρων, την μεταβολή στις αποστάσεις των σημείων μεταξύ τους που πρόκειται να προβληθούν. Επίσης, άλλο ένα σημαντικό θεώρημα είναι οτι δεδομένα που προβάλονται με πίνακες τυχαίων προβολών τείνουν (στην μειωμένη διάσταση) να γίνονται σφαιρικά [11].

ΚΕΦΑΛΑΙΟ 2. CLUSTER ENSEMBLES

Σε αυτό το κεφάλαιο θα δούμε τι είναι και πώς χρησιμοποιούνται τα cluster ensembles, που θα τα χρησιμοποιήσουμε για το πρόβλημα της ομαδοποίησης δεδομένων υψηλής διάστασης. Όσα αναφέρονται σε αυτό το Κεφάλαιο στηρίζονται κυρίως στις εργασίες των Fern και Brodley (βλ. αναφορές [8] και [9]).

2.1 Γενικά

Είδαμε ότι οι τυχαίες προβολές προσφέρουν μια καλή αναπαράσταση των δεδομένων σε μειωμένη διάσταση με έναν απλό τρόπο που καταναλώνει λίγους υπολογιστικούς πόρους, τόσο σε μνήμη όσο και σε επεξεργαστικό χρόνο. Θα ήταν πολύ βολικό αν παράλληλα η ποιότητα των προβολών ήταν πάντα εξίσου καλή, αλλά αυτό δεν συμβαίνει πάντα, όπως είναι λογικό, αφού οι τυχαίες προβολές, όπως προδίδει το όνομά τους, βασίζονται στην στοχαστικότητα και έτσι η ποιότητα των προβολών διαφέρει κάθε φορά. Κατά συνέπεια κάθε φορά που προβάλουμε τα δεδομένα μας με τυχαίους πίνακες προβολής και στη συνέχεια τα ομαδοποιούμε παίρνουμε μια διαφορετική λύση, αφού οι πίνακες προβολών θα διαφέρουν κάθε φορά μεταξύ τους.

Έτσι, με βάση την παραπάνω παρατήρηση, θα πρέπει κατά κάποιον τρόπο οι προβολές με βάση τις οποίες ομαδοποιούμε τα δεδομένα να είναι ανεξάρτητες απο τυχαίους παράγοντες, ώστε να είναι πιο ποιοτικές. Αυτό, με την σειρά του οδήγησε στην σκέψη οτι αν παίρναμε έναν αριθμό απο προβολές και τις αντίστοιχες ομαδοποιήσεις αυτών, τότε σε ένα δεύτερο βήμα συνδυάζοντας αυτές τις λύσεις να πάρουμε μια λύση καλύτερη απο όλες τις επιμέρους λύσεις του πρώτου βήματος. Η λύση του δεύτερου βήματος μπορεί να θεωρηθεί ως η τελική λύση που θα είναι σχετικά ανεπηρέαστη απο την τυχαιότητα που εισάγαμε με τις τυχαίες προβολές, ώστε το πείραμα να μπορεί να επαναληφθεί.

Η μεθοδολογία αυτή ονομάζεται cluster ensemble, και περιλαμβάνει ως στάδια τον κατασκευαστή (ensemble constructor) και την συνάρτηση απόφασης – συνδυασμού (consensus function).

2.2 Cluster Ensembles

Ένα *cluster ensemble* δημιουργείται σε δυο βήματα. Στο πρώτο βήμα κατασκευάζουμε ένα πλήθος απο r ομαδοποιήσεις (*clustering solutions*) που το ονομάζουμε *ensemble*. Στο δεύτερο βήμα συνδυάζοντας αυτές τις r λύσεις, παίρνουμε την τελική ομαδοποίηση που είναι και η λύση του προβλήματός μας. Το δεύτερο αυτό βήμα ονομάζεται *consensus function*. Σκοπός μας, όπως είπαμε και παραπάνω, είναι η ομαδοποίηση που υπολογίζεται στο δεύτερο βήμα να είναι καλύτερη απο τις r ομαδοποιήσεις που υπολογίστηκαν στο πρώτο βήμα.

2.2.1 Ορισμός του Ensemble

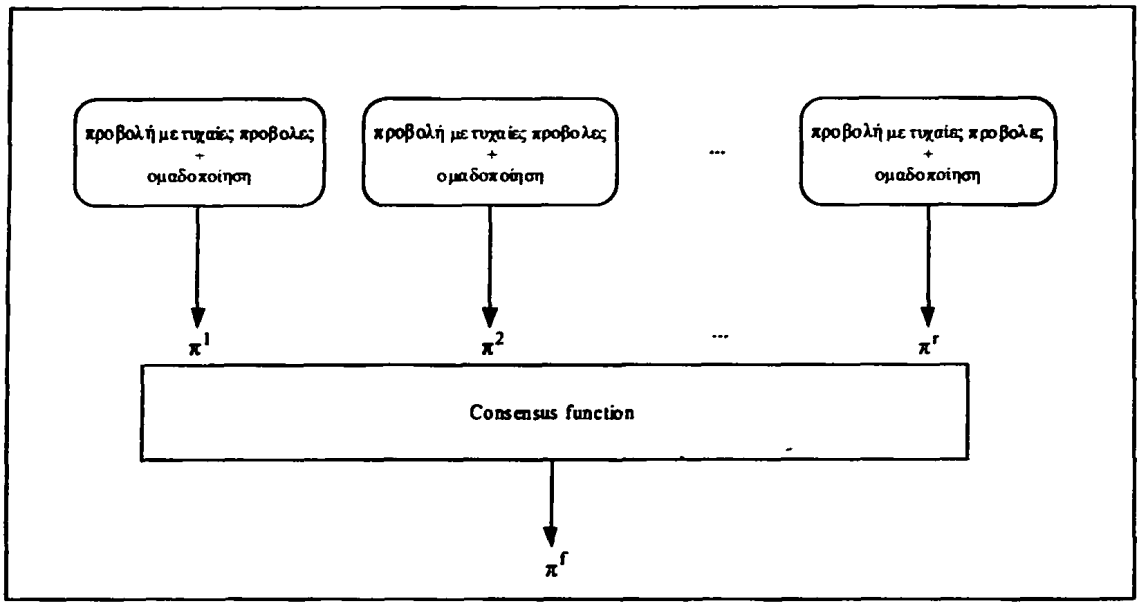
Δοθέντος ενός συνόλου N σημείων: $X = \{x_1, x_2, ..., x_N\}, x_i \in \mathbb{R}^d$ στο πρώτο βήμα παίρνουμε ένα σύνολο r ομαδοποιήσεων, που συμβολίζεται $\Pi = \{\pi^1, \pi^2, ..., \pi^r\}$. Κάθε λύση π^i είναι μια ομαδοποίηση των δεδομένων X σε K^i ξένες μεταξύ τους ομάδες (clusters), που τις συμβολίζουμε με: $\pi^i = \{c_1^i, c_2^i, ..., c_{K^i}^i\}$, με $\bigcup_k c_k^i = X$.

Δηλαδή, κάθε μια απο τις r ομαδοποιήσεις μπορεί να έχει διαφορετικό πλήθος ομάδων. Επίσης, όλα τα στοιχεία του συνόλου X ανήκουν σε κάποια ομάδα σε κάθε μια απο τις r λύσεις, δηλαδή κάθε πρότυπο ομαδοποιείται r φορές. Οι ομαδοποιήσεις μπορεί να είναι αποτέλεσμα είτε *hard clustering* αλγορίθμων (π.χ. kmeans) είτε *soft clustering* αλγορίθμων (π.χ. ΕΜ για μικτές κατανομές).

2.2.2 Ορισμός της Consensus Function

Δοθέντος ενός cluster ensemble Π και ενός ακεραίου K, που είναι το επιθυμητό πλήθος ομάδων, μια consensus function Γ χρησιμοποιεί την πληροφορία που περιέχει το Π και διαμερίζει το σύνολο των δεδομένων X σε K ξένες μεταξύ τους ομάδες.

Αυτή η ομαδοποίηση είναι και η τελική μας λύση που ονομάζουμε π^f. Σε μια πιο γενική εκδοχή η consensus function Γ θα μπορούσε να χρησιμοποιεί και τα αρχικά δεδομένα X σε συνδυασμό με το cluster ensemble Π για να δώσει την τελική λύση, αλλά αυτή η εκδοχή δεν θα μας απασχολήσει παρακάτω.



Σχήμα 2.1: Κατασκευή του ensemble – Consensus Function

2.3 Κατασκευή του Ensemble

Στη συνέχεια θα παρουσιάσουμε τρείς μεθόδους κατασκευής των ensembles. Επειδή ο σκοπός μας είναι να ομαδοποιήσουμε δεδομένα υψηλής διάστασης, θα αναφερθούμε αποκλειστικά σε μεθοδολογίες που στηρίζονται σε τεχνικές μείωσης διάστασης με πίνακες τυχαίων προβολών.

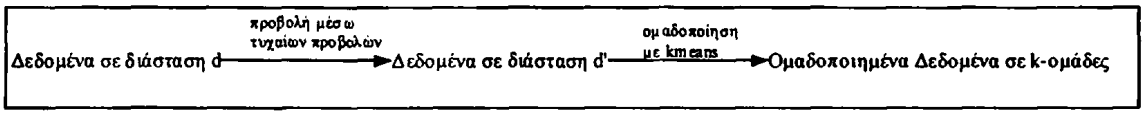
Μετά την μείωση διάστασης τα δεδομένα ομαδοποιούνται με κάποιο αλγόριθμο ομαδοποίησης. Μπορεί να χρησιμοποιηθεί οποιοσδήποτε αλγόριθμος ομαδοποίησης δεδομένων. Εμείς στα πειράματά μας (βλέπε σε επόμενο Κεφάλαιο) χρησιμοποιήσαμε εναλλακτικά τον αλγόριθμο kmeans (και συγκεκριμμένα εκτός απο τον κλασσικό αλγόριθμο kmeans, δυο άλλες παραλλαγές του που θα δούμε παρακάτω) και τον αλγόριθμο EM (expectation maximization) για μικτές κατανομές.

2.3.1 Τυχαίες προβολές (Random Projections)

Όπως είπαμε και στο πρώτο Κεφάλαιο οι τυχαίες προβολές (random projections) δείχνουν να έχουν καλές ιδιότητες για μείωση διάστασης σε δεδομένα υψηλής διάστασης. Επίσης, μια ένδειξη που μας παροτρύνει να χρησιμοποιήσουμε τυχαίες προβολές σε cluster ensembles, είναι το ότι διαφορετικές προβολές μπορεί να αποκαλύπτουν διαφορετικές πτυχές των δεδομένων και έτσι να αλληλοσυμπληρώνονται.

Όπως είπαμε, για να πάρουμε μια τυχαία προβολή δεδομένων διάστασης d σε διάσταση d', κατασκευάζουμε έναν πίνακα $(d \times d')$ με στοιχεία απο μια τυπική κανονική κατανομή $N(0,1)$, κανονικοποιώντας στη συνέχεια τις στήλες του.

Για να πάρουμε μια αναπαράσταση των δεδομένων X σε μειωμένη διάσταση, πολλαπλασιάζουμε τον $N \times d$ πίνακα X με τον $d \times d'$ πίνακα προβολής R. Τελικά, ομαδοποιούμε τα προβληθέντα δεδομένα με τον αλγόριθμο ομαδοποίησης kmeans. Για να πάρουμε ένα ensemble μεγέθους r, επαναλαμβάνουμε την παραπάνω διαδικασία με διαφορετικό κάθε φορά πίνακα προβολής r φορές.



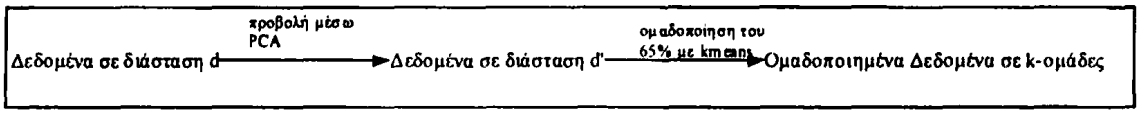
Σχήμα 2.2: Τυχαίες προβολές ως ensemble constructor.

2.3.2 Συνδυάζοντας PCA με Random Subsampling

Η μέθοδος μείωσης διάστασης PCA, είναι η βέλτιστη γραμμική μέθοδος προβολής δεδομένων που ελαχιστοποιεί το σφάλμα ανακατασκευής τους. Επειδή, όμως η PCA τεχνική είναι ντετερμινιστική, δεν έχει νόημα να πάρουμε ομαδοποιήσεις σε δεδομένα που προκύπτουν μετά την εκτέλεση της PCA, γιατί θα είχαμε πάντα ακριβώς τα ίδια δεδομένα να ομαδοποιήσουμε, οπότε χάνεται το νόημα των ensembles, όπου απαιτούνται r διαφορετικές ομαδοποιήσεις.

Έτσι, αυτό που κάνουμε για να διαφοροποιήσουμε τις επιμέρους ομαδοποιήσεις είναι το εξής. Αρχικά μέσω του PCA προβάλλουμε τα δεδομένα από διάσταση d σε διάσταση d'. Έπειτα, επιλέγουμε τυχαία κάποιο ποσοστό των δεδομένων (π.χ. 65%) και ομαδοποιούμε αυτά τα δεδομένα. Το κάθε δεδομένο συμμετέχει στην ομαδοποίηση με την ίδια πιθανότητα και η επιλογή γίνεται χωρίς

επανατοποθέτηση. Επίσης, επειδή τα μόνα δεδομένα που συμμετέχουν στην ομαδοποίηση είναι κάθε φορά ενα υποσύνολο του αρχικού συνόλου και για να μην παρατηρηθεί το φαινόμενο κάποια δεδομένα να μην ομαδοποιηθούν ποτέ, στο τέλος κάθε ομαδοποίησης αντιστοιχούμε σε μια ομάδα και εκείνα τα δεδομένα που δεν είχαν επιλεγεί να συμμετέχουν στην ομαδοποίηση. Έτσι, όλα τα δεδομένα θα έχουν ομαδοποιηθεί r φορές στο τέλος του ensemble constructor.



Σχήμα 2.3: PCA και subsampling ως ensemble constructor.

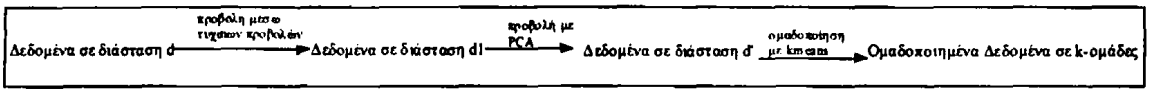
2.3.3 Συνδυάζοντας Random Projection και PCA

Ένα πρόβλημα που υπάρχει στην παραπάνω προσέγγιση (συνδυασμός PCA και subsampling) είναι οτι ο PCA δίνει μια μοναδική αναπαράσταση των δεδομένων σε μειωμένη διάσταση. Έτσι, αν αυτή η αναπαράσταση δεν είναι καλή και οδηγεί σε απώλεια χρήσιμης πληροφορίας, τότε όλες ανεξαιρέτως οι ομαδοποιήσεις θα κληρονομήσουν αυτή την αδυναμία. Από την άλλη η μέθοδος των τυχαίων προβολών έχει το πρόβλημα οτι μια συγκεκριμμένη προβολή δεν είναι πάντα ποιοτική, ωστόσο διαφορετικές τυχαίες προβολές αποκαλύπτουν, όπως είπαμε, διαφορετικές πτυχές των δεδομένων μας.

Έτσι, για να ωφεληθούμε από τα πλεονεκτήματα των παραπάνω δύο μεθόδων μπορούμε να τις συνδυάσουμε για να πάρουμε μια τρίτη μέθοδο. Δεδομένου ενός συνόλου X, προβάλουμε τα δεδομένα μέσω τυχαίων προβολών σε διάσταση d_1. Έπειτα, ο PCA μειώνει περαιτέρω τη διάσταση απο d_1 σε d', που αποτελεί σε κάθε βήμα την διάσταση στην οποία θα ομαδοποιήσουμε τα δεδομένα μας.

Από την παραπάνω μεθοδολογία έχουμε δυο πλεονεκτήματα. Πρώτον, οι τυχαίες προβολές μας βοηθάνε να κατέβουμε γρήγορα απο την πολύ μεγάλη διάσταση d σε μια μικρότερη διάσταση d_1, στην οποία ο PCA μπορεί να λειτουργήσει πιο αποδοτικά και να μας δώσει την τελική διάσταση d'. Δεύτερον, με την σειρά που πραγματοποιούμε τις μειώσεις στη διάσταση του προβλήματος εξασφαλίζουμε ότι η μέθοδος είναι στοχαστική και ότι δεν θα έχουμε το πρόβλημα που αναφέραμε παραπάνω για τον PCA. Έτσι, με τον συνδυασμό των δυο αυτών μεθόδων,

ευελπιστούμε η τρίτη μεθοδολογία να κληρονομήσει τα πλεονεκτήματα των δυο πρώτων μεθόδων.



Σχήμα 2.4: Τυχαίες προβολές και PCA ως ensemble constructor.

2.4 Consensus Functions

Όπως αναφέραμε και παραπάνω ο ρόλος μιας *consensus function* είναι παίρνοντας σαν είσοδο τις r ομαδοποιήσεις που κατασκεύασε ο ensemble constructor, να τις συνδυάσει κατάλληλα και να παράγει μια τελική ομαδοποίηση. Επειδή δεν υπάρχει σαφής συσχέτιση μεταξύ των διαφορετικών ομαδοποιήσεων (δεν γνωρίζουμε ποια υπερέχει σε ποιότητα έναντι κάποιας άλλης) δεν μπορούμε να εφαρμόσουμε απλές μεθοδολογίες, όπως ψηφοφορία (majority vote), για το σε ποιά ομάδα θα καταχωρηθεί ενα πρότυπο.

Έτσι, για να αντιμετωπίσουμε αυτό το πρόβλημα καταφεύγουμε σε μια εναλλακτική λύση, ανάγοντας το πρόβλημά μας σε ενα άλλο πιο γνωστό πρόβλημα, για το οποίο έχουμε εναλλακτικές μεθόδους να εφαρμόσουμε. Οι αλγόριθμοι που θα δούμε παρακάτω ανήκουν όλες στην κατηγορία της διαμέρισης γράφων (graph partitioning), που περιγράφουμε στη συνέχεια.

Η είσοδος σε έναν αλγόριθμο διαμέρισης γράφων είναι ενας γράφος $G=(V,W)$, όπου V είναι το σύνολο κορυφών και W ενας μη-αρνητικός και συμμετρικός $|V|\times|V|$ πίνακας ομοιότητας, με μια τιμή ομοιότητας για κάθε ζεύγος κορυφών. Για να διαμερίσουμε εναν γράφο σε K μέρη πρέπει να βρούμε K ξένα μεταξύ τους υποσύνολα κορυφών $P=\{P_1,P_2,...,P_K\}$, με $\bigcup_k P_k = V$. Εφ΄όσον ο γράφος μας δεν αποτελείται απο K συνεκτικές συνιστώσες, κάθε K-διαμέρισή του θα περιέχει ακμές που ενώνουν διαφορετικές ομάδες κορυφών. Το άθροισμα αυτών των βαρών των ακμών που τέμνουν την διαμέρισή μας ονομάζεται cut: $Cut(P,W)=\sum W(i,j)$, με τις ακμές i και j να μην ανήκουν στο ίδιο υποσύνολο P_k. Στόχος ενός αλγορίθμου διαμέρισης γράφων είναι να βρεί μια K-διαμέριση του γράφου που να ελαχιστοποιεί το παραπάνω άθροισμα.

Ο λόγος για τον οποίο το πρόβλημα της ομαδοποίησης δεδομένων σε Κ ομάδες, μπορεί να αναχθεί στο πρόβλημα της Κ-διαμέρισης γράφων, είναι ο ακόλουθος. Αν με τις κορυφές του γράφου αναπαριστούμε τα πρότυπα του συνόλου δεδομένων που θέλουμε να διαμερίσουμε και στις ακμές χρησιμοποιήσουμε ως βάρη μια τιμή ομοιότητας που υπολογίζουμε για κάθε ζεύγος προτύπων, τότε λύνοντας το πρόβλημα της Κ-διαμέρισης γράφων και ελαχιστοποιόνταις το cut θα χωρίσουμε τις κορυφές του γράφου (δηλαδή τα πρότυπα του συνόλου δεδομένων) σε Κ-ομάδες βάζοντας στην ίδια ομάδα πρότυπα που έχουν μεγαλύτερη ομοιότητα μεταξύ τους.

Παρακάτω θα δώσουμε τρεις διαφορετικές consensus functions, με κάθε μια να διατυπώνει και να λύνει ενα διαφορετικό πρόβλημα διαμέρισης γράφου, δοθέντος ενος cluster ensemble. Η πρώτη προσέγγιση αναπαριστά τα πρότυπα ως κορυφές του γράφου, ενώ η δεύτερη αναπαριστά τις ομάδες (clusters) ως κορυφές του [9]. Εδώ θα αναφερόμαστε σε αυτές με τα ονόματα instance-based και cluster-based graph formulation, αντίστοιχα. Τέλος, η τρίτη προσέγγιση αναπαριστά τα πρότυπα αλλά και τις ομάδες ταυτόχρονα ως κορυφές του γράφου σχηματίζοντας έτσι την προσέγγιση hybrid bipartite graph formulation (Fern και Brodley [10]).

2.4.1 Η προσέγγιση IBGF (Instance-Based Graph Formulation)

Στην προσέγγιση Instance-Based Graph Formulation (IBGF) κατασκευάζουμε έναν γράφο που αναπαριστά τις συσχετίσεις που έχουν τα ζεύγη σημείων του συνόλου Χ. Δοθέντος, δηλαδή ενός cluster ensemble $\Pi = \{\pi^1, \pi^2, ..., \pi^R\}$ η IBGF προσέγγιση κατασκευάζει εναν πλήρη συνδεδεμένο γράφο $G = (V, W)$, όπου:

- V είναι ενα σύνολο N κορυφών, όπου κάθε μια να αντιστοιχεί σε ένα πρότυπο του συνόλου Χ
- W είναι ενας πίνακας ομοιότητας, με $W(i,j) = \frac{1}{r}\sum_{l=1}^{r} I\left(g_l(X_i) = g_l(X_j)\right)$, με $I(\cdot)$ να είναι μια συνάρτηση που επιστρέφει 1 αν το όρισμά της είναι αληθές, και 0 διαφορετικά. Η $g_l(\cdot)$ δέχεται ως όρισμα ενα σημείο και επιστρέφει το cluster στο οποίο ανήκει στην ομαδοποίηση π^l.

Με άλλα λόγια, το $W(i,j)$ είναι το ποσοστό των λύσεων στις οποίες τα σημεία X_i, X_j ανήκουν στην ίδια ομάδα. Απο τη στιγμή που έχουμε κατασκευάσει τον παραπάνω γράφο, μπορούμε να χρησιμοποιήσουμε οποιονδήποτε αλγόριθμο διαμέρισης γράφου και να λάβουμε το αποτέλεσμα ως την τελική ομαδοποίηση.

2.4.2 Η προσέγγιση CBGF (Cluster-Based Graph Formulation)

Είναι δυνατόν οι ομάδες που σχηματίστηκαν σε διαφορετικές ομαδοποιήσεις, να περιέχουν το ίδιο σύνολο σημείων ή να περιέχουν πολλά κοινά σημεία μεταξύ τους. Τέτοιες ομάδες τις θεωρούμε όμοιες. Η προσέγγιση Cluster-Based Graph Formulation (CBGF) κατασκευάζει εναν γράφο που αναπαριστά αυτή την ομοιότητα μεταξύ ομάδων απο διαφορετικές ομαδοποιήσεις και διαμερίζει αυτό τον γράφο ώστε όμοιες ομάδες να περιέχονται στο ίδιο διαμερισμένο υποσύνολο.

Δοθέντος ενός cluster ensemble $\Pi = \{\pi^1, \pi^2, ..., \pi^r\}$, πρώτα ξαναγράφουμε το προηγούμενο σύνολο ως: $\Pi = \{c_1^1, ..., c_{K_1}^1, ..., c_1^r, ..., c_{K_r}^r\}$ με c_j^i να είναι το j-οστό cluster της i-οστής διαμέρισης π^i στο ensemble Π. Σημειώστε οτι το σύνολο των clusters σε όλες τις ομαδοποιήσεις είναι: $t = \sum_{l=1}^{r} K_l$. Η CBGF προσέγγιση κατασκευάζει εναν γράφο $G = (V, W)$, όπου:

- V είναι ένα σύνολο με t κορυφές, όπου κάθε μια αναπαριστά μια ομάδα
- W είναι ένας πίνακας, με $W(i,j)$ να είναι η ομοιότητα μεταξύ των ομάδων c_i, c_j η οποία υπολογίζεται με βάση την Jaccard ομοιότητα: $W(i,j) = \frac{|c_i \cap c_j|}{|c_i \cup c_j|}$, δηλαδή το πλήθος των κοινών σημείων στις δύο ομάδες προς το συνολικό πλήθος σημείων σε αυτές τις ομάδες.

Αποδεικνύεται ότι, η διαμέριση ενος τέτοιου γράφου, οδηγεί σε μια διαμέριση των ομάδων. Έπειτα, υπολογίζουμε μια τελική ομαδοποίηση σύμφωνα με τον παρακάτω τρόπο. Αρχικά, κάθε σύνολο απο ομάδες θεωρείται ως μια meta-ομάδα. Για κάθε

ομαδοποίηση στο ensemble, ένα σημείο θεωρείται ότι ανήκει σε μια meta-ομάδα, αν αυτό με την σειρά του περιέχει την ομάδα στην οποία το σημείο ανήκει. Σημειώστε ότι, ένα σημείο μπορεί να θεωρηθεί ότι ανήκει σε πολλές meta-ομάδες. Στην περίπτωση αυτή το σημείο καταχωρείται σε εκείνη την meta-ομάδα η οποία περιέχει τις περισσότερες ομάδες που περιέχουν αυτό το σημείο. Σε περίπτωση "ισοπαλίας" αποφασίζουμε τυχαία την καταχώρηση του σημείου σε μία απο τις meta-ομάδες που το "διεκδικούν".

Μια βασική προϋπόθεση του CBGF είναι να υπάρχει μια συσχέτιση μεταξύ των clusters σε διαφορετικές ομαδοποιήσεις. Σε περίπτωση που κάτι τέτοιο δεν ισχύει, αυτή η προσέγγιση θα αποτύχει να προσφέρει ικανοποιητικά αποτελέσματα.

2.4.3 Η προσέγγιση HBGF (Hybrid Bipartite Graph Formulation)

Η προσέγγιση Hybrid Bipartite Graph Formulation (HBGF), αναπαριστά τα πρότυπα και τις ομάδες του ensemble ταυτόχρονα ως κορυφές του γράφου που κατασκευάζει ως εξής: Δοθέντος ενός cluster ensemble $\Pi = \{\pi^1, \pi^2, ..., \pi^r\}$, η HBGF προσέγγιση κατασκευάζει εναν γράφο $G = (V, W)$, όπου:

- $V = V^C \cup V^I$, με V^C να περιέχει t κορυφές, το πλήθος των clusters σε όλο το ensemble, και V^I να περιέχει τις N κορυφές που αναπαριστούν η κάθε μία από ένα σημείο του συνόλου δεδομένων X.
- Ο πίνακας ακμών W ορίζεται ως εξής: Αν οι κορυφές i και j αναπαριστούν και οι δυο clusters ή και οι δύο σημεία, τότε $W(i, j) = 0$. Διαφορετικά αν το σημείο i ανήκει στο cluster j, τότε $W(i, j) = W(j, i) = 1$ και 0 διαφορετικά.

Αποδεικνύεται, οτι αν ένας αλγόριθμος διαμέρισης γράφων εφαρμοστεί σε έναν HBGF γράφο, τότε διαμερίζει ταυτόχρονα τις κορυφές-πρότυπα απο τις κορυφές-ομάδες. Η διαμέριση των κορυφών που αναπαριστούν τα σημεία, μπορεί να θεωρηθεί ως η τελική ομαδοποίηση των σημείων.

2.5 Spectral Clustering

Μια εναλλακτική προσέγγιση για ομαδοποίηση δεδομένων που άρχισε πρόσφατα να χρησιμοποιείται είναι το spectral clustering. Σε αυτή την μέθοδο χρησιμοποιούμε τα "μεγαλύτερα" ιδιοδιανύσματα* του πίνακα που βασίζεται στις αποστάσεις των σημείων που αποτελούν το σύνολο δεδομένων μας. Ο αλγόριθμος που ακολουθεί είναι των Ng et al [15]. Ο αλγόριθμος αυτός παίρνει σαν είσοδο έναν $N \times d$ πίνακα X με τα σημεία που θέλουμε να χωρίσουμε σε k ομάδες.

Είσοδος:	X είναι ένας $N \times d$ πίνακας των σημείων προς ομαδοποίηση k είναι το επιθυμητό πλήθος clusters
Έξοδος:	μια ομαδοποίηση των N σημείων σε k ομάδες

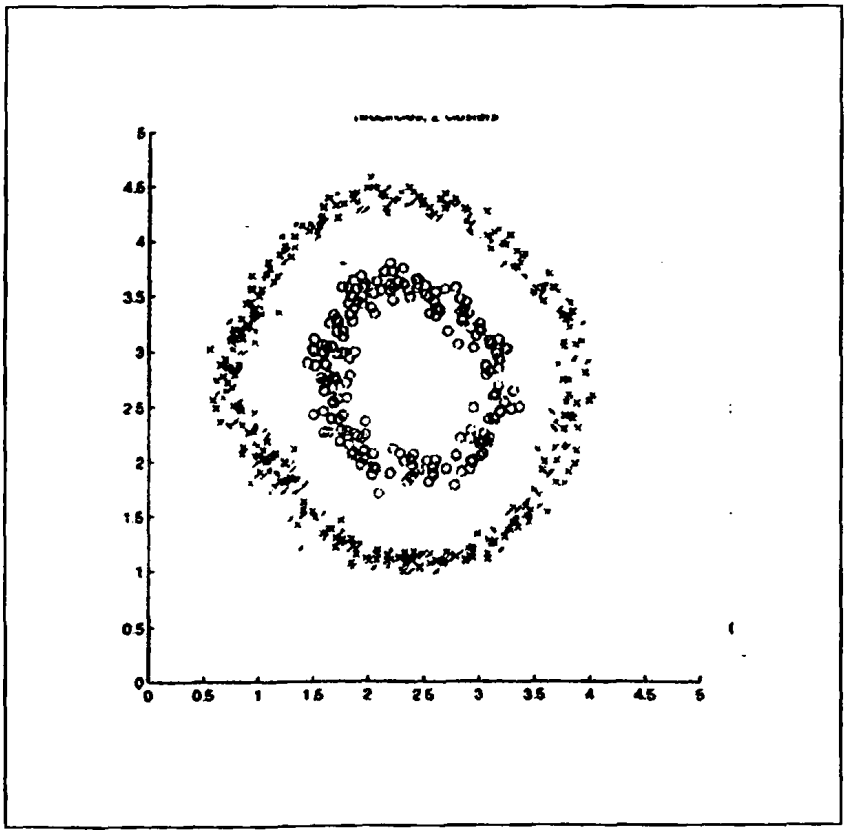
Αλγόριθμος:

1. Φτιάξε τον πίνακα συσχέτισης $A \in \mathbb{R}^{N \times N}$ που ορίζεται ως εξής: $A_{ij} = \exp\left(-\left\|x_i - x_j\right\|^2 / 2 \cdot \sigma^2\right)$, για $i \neq j$ και $A_{ii} = 0$
2. Ορίζουμε τον πίνακα D να είναι ο διαγώνιος πίνακας του οποίου τα (i,i) στοιχεία είναι τα άθροισμα των i-γραμμών του A, και φτιάχνουμε τον πίνακα $L = D^{-1/2} \cdot A \cdot D^{-1/2}$
3. βρες $u_1, u_2, ..., u_k$ τα k μεγαλύτερα ιδιοδιανύσματα του L και φτιάξε τον πίνακα $U = \left[u_1 u_2 ... u_k\right] \in \mathbb{R}^{N \times k}$
4. Φτιάξε τον Y απο τον U κανονικοποιώντας τις γραμμές του U ώστε να έχουν μοναδιαίο μήκος, δηλ. $Y_{ij} = U_{ij} \Big/ \left(\sum_j U_{ij}^2\right)^{1/2}$
5. Θεωρώντας κάθε γραμμή του Y ως ενα σημείο στον $\mathbb{R}^k$, τα ομαδοποιούμε σε k ομάδες μέσω του αλγόριθμου kmeans
6. Τέλος, το αρχικό σημείο x_i θα ανήκει στην ομάδα j αν και μόνον αν η γραμμή i του πίνακα Y ανήκει στην ομάδα j

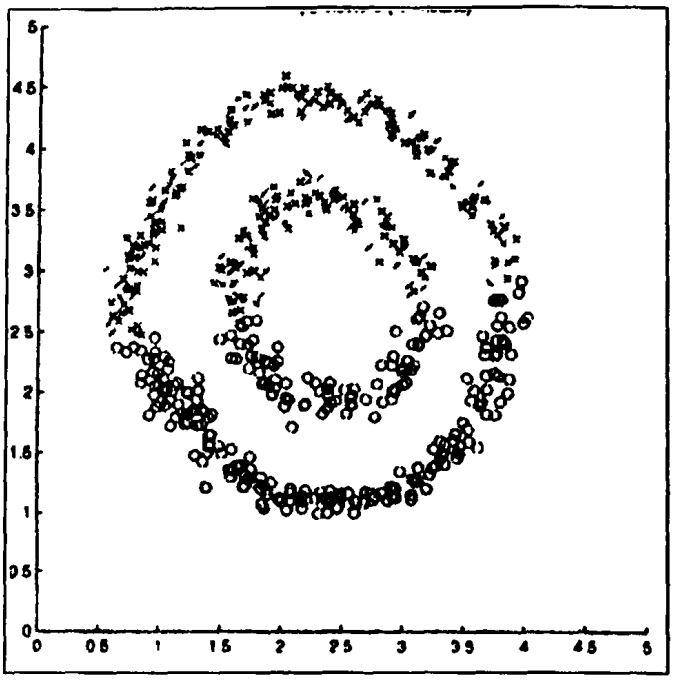
Σχήμα 2.5: Αλγόριθμος Spectral Clustering.

* δηλαδή εκείνα τα ιδιοδιανύσματα που αντιστοιχούν στις μεγαλύτερες ιδιοτιμές

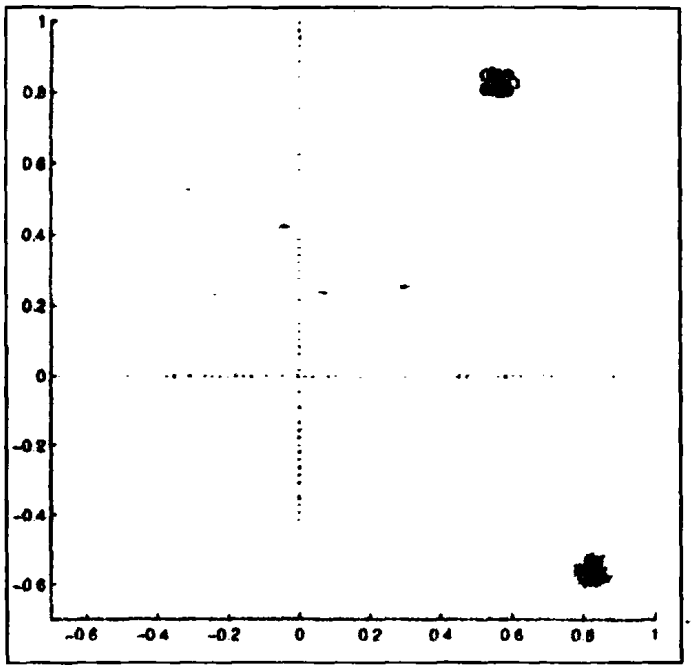
Στον παραπάνω αλγόριθμο η παράμετρος σ καθορίζει το πόσο γρήγορα ελαττώνεται η ποσότητα A_{ij} ανάλογα με την απόσταση των σημείων x_i, x_j. Σημειώνουμε ότι, ο παραπάνω αλγόριθμος είναι μόνο ένας απο πολλούς που υπάρχουν στην βιβλιογραφία. Μια σύνοψη ορισμένων εξ΄αυτών, παρατίθεται στην εργασία του Weiss [18]. Με μια πρώτη ματιά ο παραπάνω αλγόριθμος δεν φαίνεται να έχει κάποια ιδιαιτερότητα, καθώς κάποιος μπορεί να αναρωτηθεί ότι εφόσον εκτελούμε τον kmeans στο βήμα 5, γιατί να μην τον εκτελέσουμε απ΄ευθείας στα αρχικά δεδομένα μας. Στο παρακάτω Σχήμα 2.2 δίνουμε ένα παράδειγμα. Σε αυτό το σχήμα, όπου υποθέτουμε δυο ομάδες, ο αλγόριθμος kmeans βρίσκει την μη-ικανοποιητική λύση του Σχήματος 2.3. Μόλις όμως προβάλουμε τα δεδομένα μας στον χώρο $\mathbb{R}^k$ (k=2), σχηματίζονται ευδιάκριτα οι δυο ομάδες, τις οποίες και ο αλγόριθμος ανακαλύπτει, όπως φαίνεται στο Σχήμα 2.4. Σημειώνουμε, ότι οι ομάδες στο Σχήμα 2.4 σχηματίζουν γωνία 90° σε σχέση με την αρχή των αξόνων.



Σχήμα 2.6: Ένα παράδειγμα συνόλου δεδομένων προς ομαδοποίηση.



Σχήμα 2.7: Η μη-ικανοποιητική λύση του αλγόριθμου kmeans για το σύνολο δεδομένων του Σχήματος 2.6.



Σχήμα 2.8: Αποτέλεσμα ομαδοποίησης με spectral clustering για το σύνολο δεδομένων του Σχήματος 2.6.

Για την κατανόηση του αλγόριθμου spectral clustering, θα βοηθήσει να αναλύσουμε την συμπεριφορά του στην "ιδανική" περίπτωση, όπου σημεία διαφορετικών ομάδων απέχουν μεταξύ τους άπειρη απόσταση. Για την ανάλυση που ακολουθεί, υποθέτουμε οτι έχουμε $k = 3$ ομάδες, και ότι αυτές οι τρεις ομάδες έχουν

η κάθε μια N_1, N_2, N_3 σημεία και τα αντίστοιχα σύνολα συμβολίζονται με X_1, X_2, X_3, δηλαδή:

$$N = N_1 + N_2 + N_3$$
$$X = X_1 \cup X_2 \cup X_3$$

Επίσης, υποθέτουμε οτι τα πρότυπα του συνόλου $X = \{x_1, x_2, ..., x_N\}$, είναι διατεταγμένα έτσι ώστε τα πρώτα N_1 στοιχεία να ανήκουν στο σύνολο X_1, τα επόμενα N_2 πρότυπα να ανήκουν στο σύνολο X_2, κ.ό.κ. Επίσης, θα χρησιμοποιήσουμε τον συμβολισμό $j \in X_i$ αντί για τον συμβολισμό $x_j \in X_i$.

Μετακινώντας τις ομάδες σε "άπειρη" απόσταση μεταξύ τους, μηδενίζουμε εκείνα τα στοιχεία A_{ij} του πίνακα ομοιότητας A που αντιστοιχούν σε πρότυπα x_i, x_j που δεν ανήκουν στην ίδια ομάδα. Πιο συγκεκριμένα, ορίζουμε τον πίνακα ομοιότητας $\hat{A}_{ij}$, ως εξής: $\hat{A}_{ij} = 0$, αν τα αντίστοιχα πρότυπα x_i, x_j ανήκουν σε διαφορετική ομάδα, και $\hat{A}_{ij} = A_{ij}$ διαφορετικά. Επίσης, ορίζουμε τους πίνακες $\hat{L}, \hat{D}, \hat{U}$ και $\hat{Y}$ όπως και παραπάνω στον αλγόριθμο spectral clustering, με βάση τον πίνακα $\hat{A}$ που ορίσαμε προηγουμένως. Σημειώστε, οτι οι πίνακες $\hat{A}$ και $\hat{L}$ θα έχουν μη-μηδενικά στοιχεία μόνο σε περιοχές γύρω απο την κύρια διαγώνιο, δηλαδή:

$$\hat{A} = \begin{pmatrix} A^{(11)} & 0 & 0 \\ 0 & A^{(22)} & 0 \\ 0 & 0 & A^{(33)} \end{pmatrix}$$

$$\hat{L} = \begin{pmatrix} \hat{L}^{(11)} & 0 & 0 \\ 0 & \hat{L}^{(22)} & 0 \\ 0 & 0 & \hat{L}^{(33)} \end{pmatrix}$$

όπου, με $A^{(ii)}$ και $\hat{L}^{(ii)}$ συμβολίζουμε το κομμάτι εκείνο των πινάκων A και $\hat{L}$ αντίστοιχα που αναφέρονται στα ζεύγη προτύπων που ανήκουν στην ομάδα i, και ισχύουν:

$$\hat{L}^{(ii)} = \left(\hat{D}^{(ii)}\right)^{-1/2} \cdot A^{(ii)} \cdot \left(\hat{D}^{(ii)}\right)^{-1/2}$$

και

$$\hat{A}^{(ii)} = A^{(ii)} \in \mathbb{R}^{N_i \times N_i}$$

είναι ο πίνακας ομοιοτήτων των σημείων του συνόλου X_i.

Τώρα, για να υπολογίσουμε τον πίνακα $\hat{U}$, πρέπει να βρούμε τις $k=3$ πρώτες (δηλαδή μεγαλύτερες) ιδιοτιμές του πίνακα $\hat{L}$. Επειδή, ο $\hat{L}$ είναι block-diagonal, αποδεικνύεται ότι τα ιδιοδιανύσματα και οι ιδιοτιμές του $\hat{L}$ είναι η ένωση των ιδιοδιανυσμάτων και των ιδιοτιμών των αντίστοιχων πινάκων $\tilde{L}^{(11)}$, $\tilde{L}^{(22)}$ και $\tilde{L}^{(33)}$ απο τους οποίους αποτελείται, με την προσθήκη του κατάλληλου πλήθους μηδενικών. Απο την θεωρία της γραμμικής άλγεβρας, επίσης γνωρίζουμε ότι οι πίνακες της μορφής $\tilde{L}^{(ii)}$ έχουν μία ιδιοτιμή με τιμή την μονάδα και οτι επειδή $A_{jk}^{(ii)} > 0 \left(j \neq k\right)$, η επόμενη ιδιοτιμή θα είναι αυστηρά μικρότερη της μονάδας. Έτσι, οι τρεις μεγαλύτερες ιδιοτιμές του πίνακα $\hat{L}$ είναι μονάδα – μία για κάθε υποπίνακα $\tilde{L}^{(ii)}, i=1,2,3$. Οπότε, αν με $u_1^{(1)}, u_1^{(2)}, u_1^{(2)}$ συμβολίσουμε τα αντίστοιχα μεγαλύτερα ιδιοδιανύσματα των πινάκων $\tilde{L}^{(11)}$, $\tilde{L}^{(22)}$ και $\tilde{L}^{(33)}$, και τα τοποθετήσουμε κατάλληλα στον πίνακα $\hat{U}$, θα πάρουμε:

$$\hat{U} = \begin{bmatrix} u_1^{(1)} & \vec{0} & \vec{0} \\ \vec{0} & u_1^{(2)} & \vec{0} \\ \vec{0} & \vec{0} & u_1^{(3)} \end{bmatrix} \in \mathbb{R}^{N \times 3}$$

Τα N σημεία του πίνακα $\hat{U}$, πρέπει τώρα να ομαδοποιηθούνε σε 3 ομάδες. Η λύση αυτής της ομαδοποίησης είναι προφανής: έχουμε N_1 σημεία που έχουν μη-μηδεδινή τιμή μόνο στην πρώτη διάσταση, N_2 σημεία με μη-μηδενική τιμή μόνο στην δεύτερη διάσταση, κ.ό.κ. Άρα, η ομαδοποίηση που παίρνουμε είναι και η πραγματική.

Για την γενική περίπτωση που τα κέντρα των ομάδων δεν απέχουν άπειρη απόσταση μεταξύ τους, και θέλουμε να τα ομαδοποιήσουμε σε k (αντί για τρεις όπως προηγουμένως) ομάδες, μπορεί να δειχτεί με την βοήθεια της θεωρίας διαταραχής πινάκων, οτι η μορφή του πίνακα $\hat{U}$ στην οποία καταλήγουμε, είναι παρόμοια με αυτή που καταλήξαμε κατα την ανάλυση της ιδανικής περίπτωσης, όπου τα σημεία διαφορετικών ομάδων απέχουν άπειρη απόσταση μεταξύ τους. Σε αυτό τον πίνακα

$\hat{U}$, με κατάλληλες εναλλαγές των στηλών του μπορούμε να τον φέρουμε σε μια μορφή όπου όλες οι μεγαλύτερες μη-μηδενικές τιμές εμφανίζονται σε μια μόνο πάντα στήλη για κάθε σημείο, ανάλογα σε ποια απο τις k ομάδες ανήκει και στις υπόλοιπες στήλες να έχει τιμές πολύ μικρές. Ομαδοποιώντας τις γραμμές αυτού του πίνακα με τον αλγόριθμο kmeans μπορούμε να διαχωρίσουμε τα δεδομένα σε k ομάδες.

ΚΕΦΑΛΑΙΟ 3. ΜΕΘΟΔΟΛΟΓΙΕΣ ΠΟΥ ΥΛΟΠΟΙΗΘΗΚΑΝ

Σε αυτό το Κεφάλαιο θα περιγράψουμε πιο αναλυτικά ποιους απο τους αλγορίθμους του δεύτερου Κεφαλαίου υλοποιήσαμε και μέσα σε ποια μεθοδολογία πειραματιστήκαμε. Στο τέλος, θα περιγράψουμε και έναν νέο αλγόριθμο που βασίζεται στον γνωστό αλγόριθμο των k-μέσων (kmeans) και θα παραθέσουμε τα συγκριτικά αποτελέσματα που προέκυψαν για όλους τους αλγορίθμους και για όλα τα σύνολα δεδομένων με τα οποία πειραματιστήκαμε.

3.1 Γενικά

Στο προηγούμενο Κεφάλαιο, περιγράψαμε τα cluster ensembles. Στο παρόν Κεφάλαιο θα επικεντρωθούμε σε εκείνους τους αλγορίθμους που υλοποιήσαμε για να πειραματιστούμε. Θα περιγράψουμε τόσο τους αλγόριθμους που χρησιμοποιήσαμε στο πρώτο βήμα (ensemble constructor), όσο και αυτούς που χρησιμοποιήσαμε στο δεύτερο βήμα (consensus function).

Στα παρακάτω θεωρούμε οτι το σύνολο δεδομένων προς ομαδοποίηση συμβολίζεται με:

$$X = \{x_1, ..., x_N\}, x_i \in \mathbb{R}^d, i = 1, ..., N$$

δηλαδή έχουμε ένα σύνολο N προτύπων διάστασης d. Τα δεδομένα αυτά σχηματίζουν k_{true} ομάδες, και το πλήθος των ομάδων μας είναι γνωστό σε κάθε περίπτωση.

Όλες οι παρακάτω μέθοδοι στηρίζονται στην μέθοδο των cluster ensembles. Στο πρώτο βήμα (ensemble constructor) σκοπός μας είναι να εξάγουμε εναν $N \times N$ πίνακα ομοιότητας, που για κάθε ζεύγος προτύπων έχει μια τιμή ομοιότητας των σημείων. Αυτή την ομοιότητα μπορούμε να την δούμε ως την πεποίθηση οτι ενα ζεύγος προτύπων ανήκει στην ίδια ομάδα. Προφανώς, ο πίνακας αυτός είναι

συμμετρικός. Ο πίνακας αυτός αποτελεί την έξοδο του πρώτου σταδίου και με την βοήθεια αυτού ομαδοποιούνται τα δεδομένα στο δεύτερο στάδιο (consensus function).

Επίσης, να σημειώσουμε οτι δεν είναι απαραίτητο στον ensemble constructor να ομαδοποιούμε r-φορές σε k_{true} ομάδες. Μάλιστα, μας συμφέρει να ομαδοποιούμε σε k_{max} ($> k_{true}$) ομάδες, ώστε να πάρουμε μια καλύτερη εκτίμηση για τις αποστάσεις που θα υπολογιστούν μετά μέσω του πίνακα ομοιότητας. Όλες οι μέθοδοι που θα περιγράψουμε παρακάτω υλοποιήθηκαν στο προγραμματιστικό περιβάλλον της Matlab.

3.2 EM και Agglomerative Clustering

Σε αυτή τη μέθοδο, πάντα σύμφωνα με τα όσα αναφέραμε για τα cluster ensembles που περιγράψαμε στο δεύτερο Κεφάλαιο, εκτελούμε στο πρώτο βήμα r τυχαίες προβολές και έπειτα ομαδοποιούμε τα μειωμένης διάστασης δεδομένα με τον αλγόριθμο EM για την εκπαίδευση μικτών κανονικών κατανομών. Σε κάθε μια απο τις r επαναλήψεις, αφού προβληθούν τα δεδομένα στην μικρότερη διάσταση, ομαδοποιούνται μέσω του EM και προκύπτει μια ομαδοποίηση των δεδομένων μας σε k_{max} ομάδες, που είναι μια υπερεκτίμηση του πραγματικού αριθμού των ομάδων που υπάρχουν στο σύνολο δεδομένων. Να σημειώσουμε οτι σε όλες τις περιπτώσεις, το πραγματικό πλήθος ομάδων k_{true} θεωρείται γνωστό.

Κάθε εκτέλεση του EM μας δίνει ενα μικτό μοντέλο θ που περιγράφει k_{max} γκαουσιανές κατανομές στην προβληθείσα διάσταση d'. Για το σημείο x_i η πιθανότητα να έχει παραχθεί απο την l-οστή κατανομή, δίνεται από: $P(l|i,\theta), l = 1,...,k_{max}$. Ορίζουμε επίσης $P^{\theta}_{i,j}$ να είναι η πιθανότητα τα σημεία x_i και x_j να έχουν παραχθεί απο την ίδια κατανομή με βάση το μοντέλο θ, η οποία μπορεί να υπολογιστεί με τον παρακάτω τρόπο:

$$P^{\theta}_{i,j} = \sum_{l}^{k_{max}} P(l|i,\theta) \times P(l|j,\theta)$$

Έτσι, σε κάθε βήμα m που εκτελείται ο EM κατασκευάζουμε εναν $N \times N$ πίνακα A^m που δίνει τις παραπάνω πιθανότητες για όλα τα ζεύγη σημείων. Προφανώς ο πίνακας A^m που κατασκευάζεται στο m-οστό βήμα είναι συμμετρικός.

Τελικά, προκύπτει ένας πίνακας A ομοιότητας των σημείων, που είναι ο μέσος όρος των πινάκων A^m, για m=1,...,r.

$$A = \frac{1}{r}\sum_{m=1}^{r} A^m$$

Ο πίνακας ομοιότητας A δίνει την συνολική εκτίμηση δυο σημεία x_i, x_j να ανήκουν στην ίδια ομάδα με βάση τα αποτελέσματα των r ανεξάρτητων εκτελέσεων του EM στα προβληθέντα δεδομένα μας.

Στο δεύτερο στάδιο (consensus function), σκοπό μας είναι να εκμεταλλευτούμε την πληροφορία απο το πρώτο βήμα (πίνακας ομοιότητας A) και να ομαδοποιήσουμε τα δεδομένα μας σε k_{true} ομάδες. Για αυτό τον σκοπό χρησιμοποιούμε την διαδικασία του agglomerative clustering, που περιγράφεται στο παρακάτω Σχήμα.

Είσοδος: A είναι ένας $N \times N$ πίνακας ομοιότητας

k_{true} είναι το επιθυμητό πλήθος ομάδων

Έξοδος: μια ομαδοποίηση των N σημείων σε k_{true} ομάδες

Αλγόριθμος:

$l = N$

$c_i = \{X_i\}$, i = 1, …, N // δηλαδή αρχικά έχουμε N ομάδες

Repeat

- βρες το πιο **όμοιο** ζεύγος ομάδων i,j έστω c_i, c_j
- συνένωσε τις ομάδες c_i, c_j
- $l = l - 1$

Until $l \le k_{true}$

Σχήμα 3.1: Agglomerative Clustering Αλγόριθμος.

Σε αυτό τον αλγόριθμο, ξεκινάμε απο N ομάδες, δηλαδή κάθε πρότυπο ανήκει σε μια ξεχωριστή ομάδα. Σε κάθε βήμα, βρίσκουμε τις δύο ποιο όμοιες ομάδες προτύπων και τις συνενώνουμε σε μία ομάδα μειώνοντας έτσι το πλήθος των ομάδων κατα ένα, μέχρι ενα επιθυμητό πλήθος ομάδων k_{true}. Για να δουλέψει ο παραπάνω αλγόριθμος, πρέπει να οριστεί η ομοιότητα μεταξύ δυο ομάδων c_i, c_j. Η ομοιότητα ορίζεται ως η μικρότερη ομοιότητα μεταξύ σημείων σε διαφορετικές ομάδες:

$$sim(c_i, c_j) = \min_{x_i \in c_i, x_j \in c_j} A_{ij}$$

3.3 kmeans και spectral clustering

Σε αυτή τη μέθοδο ομαδοποίησης δεδομένων υψηλής διάστασης στο πρώτο στάδιο (ensemble constructor) κάνουμε προβολή των δεδομένων με τη μέθοδο των τυχαίων προβολών και έπειτα ομαδοποιούνται με τον αλγόριθμο ομαδοποίησης kmeans. Σε κάθε ομαδοποίηση με kmeans τα N δεδομένα μας χωρίζονται σε k_{max} ομάδες. Η έξοδος σε κάθε επανάληψη $m = 1,...,r$ είναι ένας $N \times N$ πίνακας A^m με τιμή 1 για εκείνα τα ζεύγη σημείων που ανήκουν στην ίδια ομάδα και 0 για εκείνα τα ζεύγη σημείων που ανήκουν σε διαφορετική ομάδα. Έτσι, αφού κάνουμε r επαναλήψεις, παίρνουμε απο το πρώτο βήμα εναν πίνακα ομοιοτήτων A ($A = \frac{1}{r}\sum_{m=1}^{r} A^m$), η οποία για κάθε ζεύγος δίνει έναν αριθμό μεταξύ 0 και 1 που είναι το ποσοστό των κοινών συνυπάρξεων κάθε ζεύγους σημείων στην ίδια ομάδα. Έτσι, όσο πιο μεγάλο είναι αυτό το ποσοστό, τόσο μεγαλύτερη πιθανότητα έχουν δυο σημεία τελικά να ανήκουν στην ίδια τελική ομαδοποίηση στην ίδια ομάδα.

Στο δεύτερο στάδιο (consensus function), ο πίνακας A που υπολογίσαμε στο πρώτο στάδιο χρησιμοποιείται από τον αλγόριθμο spectral clustering, τον οποίο περιγράψαμε σε προηγούμενο Κεφάλαιο, για να ομαδοποιήσουμε τα δεδομένα σε k_{true} ομάδες.

Εκτός απο τον γνωστό αλγόριθμο kmeans, υλοποιήσαμε και πειραματιστήκαμε με άλλες δυο παραλλαγές του, που υπάρχουν στην βιβλιογραφία [14]. Επίσης, μελετήσαμε εναν εναλλακτικό ορισμό της ομοιότητας, που προτείνουμε στην Ενότητα 3.4.

3.3.1 Ο Αλγόριθμος Ομαδοποίησης global kmeans

Ο αλγόριθμος global kmeans [14] είναι μια παραλλαγή του αλγόριθμου kmeans. Είναι ενας ντετερμινιστικός μηχανισμός αναζήτησης που λειτουργεί επαναληπτικά. Λύνουμε διαδοχικά το πρόβλημα της ομαδοποίησης από k=1 μέχρι ένα μέγιστο επιτρεπτό πλήθος ομάδων k_{max}.

Έτσι, ξεκινάμε λύνοντας το πρόβλημα της ομαδοποίησης για k=1, που έχει την προφανή λύση που είναι το κέντρο των σημείων του συνόλου δεδομένων X. Έστω, ότι έχουμε λύσει το πρόβλημα για k=M-1 ομάδες και πάμε να λύσουμε το πρόβλημα για k=M ομάδες. Για να το κάνουμε αυτό, κάνουμε N (όπου N το πλήθος των σημείων μας) επανεκτελέσεις του kmeans αλγορίθμου ομαδοποίησης με M κέντρα, όπου τα M-1 κέντρα αρχικά τοποθετούνται στη λύση του προηγούμενου βήματος και το M-οστό κέντρο αρχικοποιείται διαδοχικά σε όλα τα σημεία του συνόλου δεδομένων μας. Έτσι, παίρνουμε N λύσεις για M ομάδες και κρατάμε την καλύτερη ως ενδιάμεση λύση για k=M μέχρι να φτάσουμε στην τελική λύση $k=k_{max}$, μέσω της επαναληπτικής αυτής διαδικασίας.

Η έξοδος του αλγορίθμου είναι ενας $N \times N$ πίνακας ομοιότητας με 1 για τα ζεύγη σημείων που ανήκουν στην ίδια ομάδα και 0 σε διαφορετική περίπτωση, όπως ακριβώς και στην περίπτωση του απλού αλγορίθμου kmeans. Στην εργασία [14], δείχνεται πειραματικά οτι αυτή η ολική αντιμετώπιση του προβλήματος είναι καλύτερη απο την κλασσική προσέγγιση μέσω του kmeans και ότι δεν έχει τα μειονεκτήματά του, όπως η εξάρτηση απο τις αρχικές τιμές των κέντρων και έτσι δίνει μια λύση που είναι καλύτερη απο την τοπική αναζήτηση που κάνει ο kmeans.

3.3.2 Ο Αλγόριθμος Ομαδοποίησης fast global kmeans

Βασισμένοι στην γενική ιδέα του αλγόριθμου global kmeans, στο [14], προτείνεται και μια παραλλαγή αυτού που βελτιώνει τον χρόνο εκτέλεσής του, χωρίς να στοιχίζει ιδιαίτερα σε απόδοση, που είναι ο αλγόριθμος fast global kmeans. Η διαφορά του απο τον αλγόριθμο global kmeans είναι στον τρόπο που υπολογίζεται η λύση στο k-οστό βήμα απο την λύση που έχουμε στο (k-1)-οστό βήμα.

Έστω, η λύση που έχουμε απο το (k-1)-οστό βήμα (δηλαδή τα k-1 κέντρα) είναι η:

$$m_1^*(k-1), m_2^*(k-1), ..., m_{k-1}^*(k-1)$$

Τότε, στην περίπτωση του αλγόριθμου global kmeans, θα έπρεπε να εξετάσουμε τις N λύσεις που παίρνουμε ξεκινώντας απο τις εξής αρχικές καταστάσεις:

$$\left(m_1^*(k-1), m_2^*(k-1), ..., m_{k-1}^*(k-1), x_i\right), i = 1, ..., N$$

και ξεκινώντας απο αυτές να εκτελέσουμε τον απλό αλγόριθμο kmeans και να κρατήσουμε την καλύτερη απο τις N λύσεις.

Αντίθετα, στον αλγόριθμο fast global kmeans, υπολογίζουμε για κάθε ένα από τα N υποψήφια νέα κέντρα ένα άνω όριο του σφάλματος ομαδοποίησης:

$$E_i \le E - b_i, i = 1,...,N$$

όπου E είναι το τετραγωνικό σφάλμα της λύσεως στο (k-1)-οστό βήμα και το b_i δίνεται από:

$$b_i = \sum_{j=1}^{N} \max\left(d_{k-1}^j - \left\|x_i - x_j\right\|^2, 0\right)$$

που είναι η βελτίωση που θα έχουμε στο σφάλμα αν στη λύση του (k-1)-οστού βήματος προσθέσουμε ως k-οστό κέντρο το σημείο x_i. Η ποσότητα d_{k-1}^j είναι η τετραγωνική απόσταση του σημείου x_j απο το κέντρο στο οποίο ανήκει στην (k-1)-οστή λύση. Έτσι, η παραπάνω ποσότητα υπολογίζει την βελτίωση του τετραγωνικού σφάλματος για την παραπάνω ορισμένη κατάσταση, με τα k κέντρα να έχουν τιμές:

$$\left(m_1^*(k-1), m_2^*(k-1),..., m_{k-1}^*(k-1), x_i\right).$$

Επειδή, σε κάθε βήμα του απλού αλγόριθμου kmeans, το σφάλμα μειώνεται, η παραπάνω ποσότητα E_i αποτελεί ενα άνω όριο στο σφάλμα που θα πάρουμε εκτελώντας τον αλγόριθμο kmeans ξεκινώντας απο την αρχική κατάσταση

$$\left(m_1^*(k-1), m_2^*(k-1),..., m_{k-1}^*(k-1), x_i\right)$$

για το βήμα k.

Έχοντας, υπολογίσει τις τιμές $E_i\left(i = 1,...,N\right)$, ο αλγόριθμος fast global kmeans τρέχει τον απλό αλγόριθμο kmeans μόνο για την αρχική κατάσταση:

$$\left(m_1^*(k-1), m_2^*(k-1),..., m_{k-1}^*(k-1), x_j\right)$$

με:

$$j = \arg\max_i b_i$$

δηλαδή με νέο κέντρο το σημείο x_j που μεγιστοποιεί το $b_i(i=1,...,N)$, δηλαδή που ελαχιστοποιεί το σφάλμα $E_i(i=1,...,N)$.

3.4 Ένας εναλλακτικός ορισμός της ομοιότητας

Εκμεταλλευόμενοι την επαναληπτική λειτουργία του αλγόριθμου global kmeans, που υπολογίζει λύσεις για όλα τα ενδιάμεσα προβλήματα ομαδοποίησης με τιμές για τον αριθμό των ομάδων απο k=1 μέχρι $k=k_{max}$, μελετήσαμε εναν εναλλακτικό ορισμό της απόστασης δυο σημείων. Σε αυτή την παραλλαγή, στο τέλος του επαναληπτικού αλγορίθμου global kmeans, αντί να παίρνουμε σαν έξοδο εναν $N \times N$ πίνακα με 1 και 0 που να δηλώνουν, αντίστοιχα, αν ενα ζεύγος σημείων ανήκει ή δεν ανήκει στην ίδια ομάδα, παίρνουμε εναν $N \times N$ πίνακα με δυνατές τιμές για τα στοιχεία του, τις εξής:

$$0, \frac{2}{K}, \frac{3}{K}, ..., \frac{K}{K}(=1)$$

όπου K είναι το μέγιστο πλήθος ομάδων που επιτρέπουμε στον αλγόριθμο global kmeans. Έτσι, αν ενα ζεύγος σημείων έχει την τιμή ομοιότητας $\frac{i}{K}, i=1,...,K$ τότε αυτό σημαίνει ότι τα σημεία αυτά είχαν ομαδοποιηθεί για τελευταία φορά στην ίδια ομάδα κατά την i-οστη (απο τις K) επαναλήψεις του global kmeans αλγορίθμου. Αν έχει τιμή ομοιότητας 0, τότε τα σημεία αυτά βρίσκονταν μαζί μόνον στην πρώτη επανάληψη όπου όλα τα σημεία ανήκαν στην ίδια ομάδα.

Αυτό μας δίνει μια καλύτερη εικόνα της ομοιότητας μεταξύ δυο σημείων, καθώς επιτρέπει και ενδιάμεσες τιμές ομοιότητας σημείων επιπλέον των ακραίων τιμών 0 και 1. Όπως θα δούμε και στο επόμενο Κεφάλαιο, πειραματικά αυτή η παραλλαγή είχε τουλάχιστον ελαφρώς καλύτερα αποτελέσματα, και σε αρκετές περιπτώσεις πολύ καλύτερα αποτελέσματα από τις άλλες μεθόδους.

3.5 Ομαδοποίηση δεδομένων υψηλής διάστασης χωρίς μείωση διάστασης

Για να μπορέσουμε να εκτιμήσουμε κατα πόσο επηρεάζεται η ποιότητα των ομαδοποιήσεων από την μείωση διάστασης που κάνουμε μέσω τυχαίων προβολών, κάναμε πειράματα με όλες τις παραπάνω μεθοδολογίες, χωρίς ωστόσο να προβάλουμε τα δεδομένα μας σε μικρότερη διάσταση. Έτσι μπορούμε να εκτιμήσουμε την όποια απώλεια πληροφορίας μέσω της διαδικασίας της τυχαίας προβολής σε μικρότερη διάσταση.

Αυτό, μας δίνει μια αίσθηση του πόσο μας κοστίζει σε ποιότητα το αποτέλεσμα της ομαδοποίησης όταν επιλέγουμε να προβάλουμε τα δεδομένα μας με τυχαίες προβολές. Επίσης, μας δίνει μια εκτίμηση για το ποια μέθοδος επηρεάζεται περισσότερο ή λιγότερο απο την διαδικασία της προβολής σε σχέση με τις υπόλοιπες.

3.6 Εκτίμηση της ποιότητας των τελικών ομαδοποιήσεων

Ένα επίσης πολύ ενδιαφέρον και σημαντικό ζήτημα στο πρόβλημα ομαδοποίησης είναι το πόσο καλύτερη είναι μια ομαδοποίηση από μια άλλη. Επίσης, ενα ζήτημα είναι το πλήθος των ομάδων που θα επιλέξουμε για να ομαδοποιήσουμε τα δεδομένα δεδομένου ότι το πλήθος τους αποτελεί μια επιπλέον παράμετρο προς εκτίμηση.

Σε όλα τα παρακάτω πειράματα, ξέρουμε το πραγματικό πλήθος ομάδων και έτσι στο δεύτερο βήμα (consensus function) ομαδοποιούμε πάντα σε k_{true} ομάδες, το πραγματικό δηλαδή πλήθος ομάδων.

Οπότε για να συγκρίνουμε το αποτέλεσμα δυο ομαδοποιήσεων, θα πρέπει να ορίσουμε ενα κριτήριο για το πότε μια ομαδοποίηση θεωρείται καλύτερη απο μια άλλη, καθώς και κατά πόσο μια ομαδοποίηση πλησιάζει στην πραγματική ομαδοποίηση των δεδομένων μας, εφόσον γνωρίζουμε ποια δεδομένα ανήκουν σε ποια ομάδα.

3.6.1 Το κριτήριο NMI

Το πρώτο κριτήριο που χρησιμοποιούμε για την αξιολόγηση της ποιότητας της ομαδοποίησης είναι το NMI (Normalized Mutual Information Criterion [4]). Το NMI αντιμετωπίζοντας δυο διαμερίσεις των σημείων ως τυχαίες μεταβλητές, υπολογίζει την αμοιβαία πληροφορία που μοιράζονται και την κανονικοποιεί στο διάστημα [0,1]. Δηλαδή, αν L_1 είναι η πραγματική διαμέριση των σημείων μας σε k ομάδες και L_2 μια

ομαδοποίηση που έχουμε υπολογίσει εμείς με κάποιον τρόπο, τότε όσο πιο κοντά στην μονάδα είναι το $NMI(L_1, L_2)$ τόσο καλύτερα ταιριάζει ή πλησιάζει η υπολογισμένη ομαδοποίηση L_2 στην πραγματική ομαδοποίηση L_1.

Έστω, $n_h^{(a)}$ είναι το πλήθος σημείων στην ομάδα C_h με βάση μια διαμέριση L_a, και $n_l^{(b)}$ το πλήθος των σημείων στην ομάδα C_l με βάση μια άλλη διαμέριση L_b. Επίσης, έστω $n_{h,l}$ να είναι τα σημεία εκείνα που ανήκουν στην ομάδα h με βάση τη διαμέριση L_a και ταυτόχρονα ανήκουν στην ομάδα l με βάση την διαμέριση L_b. Τότε το NMI για δυο διαμερίσεις L_a, L_b υπολογίζεται ως εξής:

$$NMI(L_a, L_b) = \frac{I(L_a, L_b)}{\sqrt{H(L_a) \cdot H(L_b)}}$$

όπου $I(L_a, L_b)$ είναι η αμοιβαία πληροφορία των L_a, L_b και $H(L)$ να είναι η εντροπία της διαμέρισης L. Έτσι, παίρνουμε:

$$NMI(L_a, L_b) = \frac{\sum_{h=1}^{k^{(a)}} \sum_{l=1}^{k^{(b)}} n_{h,l} \cdot \log\left(\frac{n \cdot n_{h,l}}{n_h^{(a)} \cdot n_l^{(b)}}\right)}{\sqrt{\left(\sum_{h=1}^{k^{(a)}} n_h^{(a)} \cdot \log\left(\frac{n_h^{(a)}}{n}\right)\right)\left(\sum_{l=1}^{k^{(b)}} n_l^{(b)} \cdot \log\left(\frac{n_l^{(b)}}{n}\right)\right)}}$$

3.6.2 Σφάλμα ομαδοποίησης

Εκτός απο το κριτήριο NMI, υπάρχει ακόμα ένα κριτήριο, απο το οποίο μπορούμε να συμπεράνουμε πόσο καλή είναι μια ομαδοποίηση δεδομένων, δεδομένης της πραγματικής ομαδοποίησης των δεδομένων αυτών. Πρόκειται, για το σφάλμα ομαδοποίησης (clustering error), που είναι το ποσοστό των προτύπων που έχουν καταχωρηθεί εσφαλμένα σε άλλη ομάδα και υπολογίζεται ως ο λόγος του πλήθους των προτύπων που καταχωρήθηκαν σε λάθος ομάδα προς το συνολικό πλήθος των προτύπων που περιέχει το σύνολο δεδομένων.

$$clustering\ error = \frac{\#\,\text{προτύπων που καταχωρήθηκαν σε λάθος ομάδα}}{\#\,\text{προτύπων στο σύνολο δεδομένων}}$$

Επίσης, αντίστοιχα, μπορούμε να υπολογίσουμε και το ποσοστό επιτυχίας της μεθόδου, που προκύπτει αν αφαιρέσουμε απο την μονάδα το παραπάνω κλάσμα.

$$\text{κλάσμα επιτυχίας} = 1 - clustering\ error$$

ΚΕΦΑΛΑΙΟ 4. ΠΕΙΡΑΜΑΤΙΚΑ ΔΕΔΟΜΕΝΑ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

Σε αυτό το Κεφάλαιο θα περιγράψουμε τα δεδομένα, τόσο τεχνητά όσο και πραγματικά, που χρησιμοποιήσαμε για να αξιολογήσουμε τις τεχνικές ομαδοποίησησης που περιγράψαμε στα προηγούμενα. Τα κύρια χαρακτηριστικά των δεδομένων που μας ενδιαφέρουν είναι το πλήθος τους N, η διάστασή τους d, το πλήθος ομάδων που σχηματίζουν k_{true}, καθώς και, στην περίπτωση που είναι διαθέσιμο, το πραγματικό διάνυσμα ομαδοποίησης L ($N \times 1$), δηλαδή η πληροφορία για κάθε στοιχείο του συνόλου σε ποια απο τις k_{true} ομάδες ανήκει.

Επιπλέον, στα τεχνητά δεδομένα δίνουμε την εκκεντρότητα (eccentricity e) και τον βαθμό διαχωρισιμότητας (separation degree c) βάσει των οποίων δημιουργήθηκαν οι ομάδες. Η εκκεντρότητα μας δίνει ενα μέτρο του πόσο σφαιρικά είναι τα δεδομένα μιας ομάδας, ενώ ο βαθμός διαχωρισιμότητας μας δίνει μια πληροφορία του πόσο κοντά ή μακριά είναι δυο ομάδες. Διαχωρισιμότητα πάνω απο 1 σημαίνει ότι οι ομάδες είναι επαρκώς απομακρυσμένες, ώστε να μπορούν να διακριθούν.

Σημειώνουμε οτι, για τα πειράματά μας, το πραγματικό πλήθος ομάδων k_{true} που σχηματίζουν τα δεδομένα μας (πραγματικά και τεχνητά) είναι γνωστό, αλλά δεν επηρεάζει την λειτουργία των αλγορίθμων μας αυτή η γνώση. Αυτή η πληροφορία χρησιμοποιείται μόνο για την αξιολόγηση των αποτελεσμάτων μας και την μεταξύ των μεθόδων μας σύγκριση.

4.1 Σκοπός των πειραμάτων

Όπως είπαμε, σκοπός αυτού του Κεφαλαίου είναι να αξιολογήσουμε τις διάφορες τεχνικές ομαδοποίησης που περιγράψαμε στο τρίτο Κεφάλαιο μέσα απο πειράματα

που κάναμε σε κάποια σύνολα δεδομένων, τόσο τεχνητά κατασκευασμένα όσο και πραγματικά.

Όλα τα πειράματα που θα περιγράψουμε παρακάτω, στηρίζονται στην φιλοσοφία των cluster ensembles, που αναπτύξαμε στο δεύτερο Κεφάλαιο. Θα χρησιμοποιήσουμε και θα συγκρίνουμε συνολικά τέσσερις διαφορετικές τεχνικές για το στάδιο των ensembles, ενώ θα δούμε και δυο εναλλακτικές προσεγγίσεις για την consensus function.

Όπως είπαμε, στο βήμα των ensembles, ο σκοπός μας είναι να πάρουμε εναν $N \times N$ πίνακα Α ομοιότητας των προτύπων μας, με βάση τον οποίο η consensus function στο δεύτερο στάδιο, θα ομαδοποιήσει τα πρότυπα. Οι τεχνικές που θα δούμε για το πρώτο βήμα, είναι ο αλγόριθμος ΕΜ για μικτές κατανομές, καθώς και άλλες τρεις μέθοδοι ομαδοποίησης που βασίζονται στον αλγόριθμο ομαδοποίησης kmeans. Αυτές είναι, καταρχήν ο κλασσικός αλγόριθμος ομαδοποίησης kmeans, καθως και οι global και fast global παραλλαγές αυτού [14] και τέλος θα αξιολογήσουμε τον εναλλακτικό ορισμό της ομοιότητας που προτείνουμε στην Ενότητα 3.4.

Επειδή, σε όλες τις περιπτώσεις έχουμε να ομαδοποιήσουμε δεδομένα υψηλής διάστασης, πριν εφαρμοστούν οι παραπάνω αλγόριθμοι ομαδοποίησης, τα δεδομένα προβάλονται σε μια μειωμένη διάσταση μέσω τυχαίων προβολών. Συγκεκριμμένα, τα στοιχεία p_{ij} των πινάκων προβολών P προκύπτουν απο την τυπική κανονική κατανομή.

$$p_{ij} \sim N(0,1), i.i.d$$

Αφού, λοιπόν απο το στάδιο των ensembles, υπολογιστεί ο πίνακας ομοιότητας Α, στο δεύτερο στάδιο της consensus function, εξετάζουμε δυο τεχνικές ομαδοποίησης, την agglomerative τεχνική και τον αλγόριθμο spectral clustering.

Σημειώνουμε, ότι το πλήθος των ομάδων που χρησιμοποιούμε στο στάδιο των ensembles, είναι πάντοτε μια υπερεκτίμηση του πραγματικού πλήθους ομάδων του συνόλου δεδομένων. Στις περισσότερες περιπτώσεις αν k_{true} είναι το πραγματικό πλήθος ομάδων, τότε στο πρώτο βήμα οι αλγόριθμοι ομαδοποίησης ομαδοποιούν τα δεδομένα σε $2 \times k_{true}$ ή $3 \times k_{true}$ ομάδες. Στο στάδιο της consensus function, τα πρότυπα ομαδοποιούνται πάντοτε στο πραγματικό πλήθος ομάδων k_{true} που σχηματίζουν.

Ως μέτρο αξιολόγησης των πειραμάτων που κάναμε, υπολογίζουμε τις τιμές του NMI και το ποσοστό επιτυχίας όπως τα περιγράψαμε στην Ενότητα 3.6. Πρέπει να τονίσουμε, πως τα όποια συγκριτικά αποτελέσματα των διαφόρων μεθόδων που χρησιμοποιήσαμε είναι ενδεικτικά και ότι κάθε μέθοδος μπορεί να είναι καλύτερη απο τις υπόλοιπες ανάλογα με το σύνολο δεδομένων που χρησιμοποιείται. Με άλλα λόγια, τα αποτελέσματα των παρακάτω πειραμάτων δεν μπορούν να αποτελέσουν απόδειξη οτι κάποια μέθοδος θα είναι η καλύτερη απο τις υπόλοιπες σε άλλα σύνολα δεδομένων.

4.2 Τα σύνολα δεδομένων

Παρακάτω δίνουμε τρεις Πίνακες με στοιχεία για τα σύνολα δεδομένων με τα οποία πειραματιστήκαμε. Πρώτα έχουμε τα τεχνητά δεδομένα:

α/α	N	d	k_{true}	e	c
1	500	10	10	5	2
2	500	10	10	5	1
3	500	10	10	5	0.5
4	500	10	10	5	0.4
5	500	10	10	5	0.3
6	500	10	10	5	0.2
7	500	10	10	5	0.1

Πίνακας 4.1: Τεχνητά δεδομένα και τα χαρακτηριστικά τους.

Τα παραπάνω τεχνητά δεδομένα κατασκευάστηκαν με βοήθεια της Matlab συνάρτησης mixgen [17] που είναι διαθέσιμη απο το διαδίκτυο. Τροποποιήσαμε, την παραπάνω ρουτίνα για να παίρνουμε ως έξοδο μαζί με τα παραγόμενα σημεία και ενα διάνυσμα με την πληροφορία για το ποια κατανομή έχει παράγει το κάθε σημείο. Τα παραπάνω δεδομένα χρησιμοποιήθηκαν για πειράματα που δεν περιέχουν προβολές και τα αποτελέσματά τους θα τα δούμε σε παρακάτω Ενότητα.

Στον αμέσως επόμενο Πίνακα, βλέπουμε τα πραγματικά βιολογικά δεδομένα που χρησιμοποιήσαμε στα πειράματά μας.

Σύνολο Δεδομένων	N	d	k_{true}
Δεδομένα Σήψης	71	4857	3
Δεδομένα λευχαιμίας	38	7129	2
Επιλεγμένα Δεδομένα λευχαιμίας	38	50	2
Δεδομένα Καρκίνου Παχέος Εντέρου	62	2000	2
Δεδομένα Κληρονομικού Καρκίνου του Μαστού	22	3226	3

Πίνακας 4.2: Πραγματικά Βιολογικά Δεδομένα και τα χαρακτηριστικά τους.

Το πρώτο σύνολο δεδομένων αφορά στην σήψη και είναι δεδομένα από 71 ασθενείς. Τα δεδομένα λευχαιμίας αφορούν σε μια μορφή λευχαιμίας που έχουν ληφθεί απο 38 άτομα με τους 27 να έχουν έναν από δυο δυνατούς τύπους λευχαιμίας και τους 11 να μην πάσχουν από λευχαιμία. Επίσης, τα επιλεγμένα δεδομένα λαυχαιμίας αφορούν το ίδιο σύνολο δεδομένων, με την διαφορά οτι έχουν επιλεχθεί τα 50 πιο κρίσιμα χαρακτηριστικά.

Τα δεδομένα για τον καρκίνο του παχέος εντέρου αφορούν 62 ασθενείς με τους 22 από αυτούς να πάσχουν από τον συγκεκριμμένο τύπο καρκίνου. Τέλος, τα δεδομένα για τον κληρονομικό καρκίνο του μαστού αφορούν 22 γυναίκες με τις 15 να έχουν έναν από δυο τύπους καρκίνου (8 και 7 αντίστοιχα) και 7 να μην έχουν την ασθένεια. Για όλα τα σύνολα βιολογικών δεδομένων, πλην αυτού της σήψης, μπορείτε να βρείτε στοιχεία στην ιστοσελίδα [15].

Στη συνέχεια, στον Πίνακα 4.3, παραθέτουμε στοιχεία για κάποι άλλα ευρέως χρησιμοποιημένα πειραματικά σύνολα δεδομένων, σχετικά υψηλής διάστασης, τα οποία έχουν χρησιμοποιηθεί στις εργασίες [12] και [13].

Σύνολο Δεδομένων	N	d	k_{true}
CHART	600	60	6
MFEAT (fourier)	2000	76	10
SATIMAGE	4435	36	6
SEGMENTATION	2310	19	7

Πίνακας 4.3: Πραγματικά Πειραματικά Δεδομένα και τα χαρακτηριστικά τους.

Το mfeat σύνολο δεδομένων αφορά χειρόγραφα ψηφία (0 έως 9) με 200 πρότυπα από κάθε κατηγορία. Τα χαρακτηριστικά προέρχονται απο

μετασχηματισμούς fourier. Το satimage αφορά σε δορυφορικές εικόνες, ενώ το segmentation είναι ενα σύνολο δεδομένων απο εικόνες εδάφους.

4.3 Πειράματα με Τεχνητά Δεδομένα

Σε αυτή την Ενότητα θα περιγράψουμε τα πειράματα που κάναμε σε τεχνητά δεδομένα που παρήχθησαν απο την ρουτίνα mixgen [16]. Θεωρούμε οτι θέλουμε να ομαδοποιήσουμε σε k_{true} ομάδες τα δεδομένα του Πίνακα 4.1. Στα παρακάτω πειράματα δεν κάνουμε μείωση διάστασης. Τα πειράματα που κάναμε είναι τα εξής τρία:

Πείραμα 1: Απ΄ευθείας εφαρμογή του spectral clustering στα δεδομένα μας. Το σ του πρώτου βήματος είναι μια παράμετρος που επηρεάζει άμεσα το αποτέλεσμα της ομαδοποίησης. Στον Πίνακα 4.4 δίνουμε τα συγκεντρωτικά αποτελέσματα για διάφορες τιμές του σ.

Πείραμα 2: Εξετάζουμε τον προτεινόμενο ορισμό της ομοιότητας. Συγκεκριμμένα, πρώτα εκτελούμε μια ομαδοποίηση σε $3 \times k_{true}$ ομάδες και ο πίνακας ομοιότητας A που προκύπτει χρησιμοποιείται στο βήμα 1 του αλγόριθμου spectral clustering. Οπότε, με βάση αυτό τον πίνακα ομοιότητας εκτελούμε τον αλγόριθμο spectral clustering, για να ομαδοποιήσουμε τελικά τα δεδομένα μας σε k_{true} ομάδες.

Πείραμα 3: Τέλος, στο τρίτο πείραμα με τεχνητά δεδομένα, εκτελούμε τον αλγόριθμο fast global kmeans για τα δεδομένα μας.

Τα αποτελέσματα φαίνονται στον παρακάτω Πίνακα 4.4. Για την αξιολόγηση των αποτελεσμάτων κάνουμε χρήση του NMI κριτηρίου που περιγράψαμε στην Ενότητα 3.5.

		c = 2	c = 1	c = 0.5	c = 0.4	c = 0.3	c = 0.2	c = 0.1
		NMI	NMI	NMI	NMI	NMI	NMI	NMI
Πείραμα 1	σ = 10	0.9821	0.9115	0.6069	0.5361	0.1471	0.1291	0.1062
	σ = 5	1.0000	0.9914	0.8823	0.8072	0.2064	0.1568	0.1244
	σ = 2	1.0000	0.9921	0.8904	0.8263	0.1718	0.1265	0.1047
	σ = 1.5	1.0000	1.0000	0.9029	0.8213	0.1704	0.1493	0.1227
	σ = 1	1.0000	1.0000	0.9398	0.8498	0.1832	0.1388	0.1229
	σ = 0.5	1.0000	1.0000	0.9577	0.9122	0.2284	0.2070	0.1070
Πείραμα 2	k = 2xk	0.9882	0.9882	0.9403	0.8859	0.2388	0.2138	0.1294
	k = 3xk	0.9882	0.9882	0.9340	0.8875	0.2411	0.1859	0.1285
Πείραμα 3		1	1	0.9359	0.8905	0.2304	0.2114	0.1345

Πίνακας 4.4:Αποτελέσματα ομαδοποιήσεων για τα τεχνητά δεδομένα του Πίνακα 4.1.

Παρατηρούμε γενικά ότι, κατα μέσο όρο τα αποτελέσματα του τρίτου πειράματος είναι ελαφρώς πιο ικανοποιητικά. Δηλαδή, ο fast global kmeans αλγόριθμος παράγει καλύτερα αποτελέσματα σε σχέση με τον αλγόριθμο spectral clustering, ανεξάρτητα με ποιον τρόπο ο τελευταίος παράγει τον πίνακα ομοιότητας Α. Αν συγκρίνουμε όμως τον αλγόριθμο spectral clustering με βάση τον πίνακα ομοιότητας που χρησιμοποιεί (Πειράματα 1 και 2), τότε θα δούμε οτι τα Πειράματα του τύπου 1 λειτουργούν καλύτερα μόνο στην περίπτωση που το σ (η παράμετρος του βήματος 1) γίνει αρκετά μικρή (0.5). Σε γενικές γραμμές όμως ο πίνακας ομοιότητας που παίρνουμε απο την τροποποιημένη έκδοση του kmeans οδηγεί σε ικανοποιητικές επιδόσεις. Όπως είναι λογικό, σε όλα τα παραπάνω πειράματα, όσο μικραίνει ο βαθμός διαχωρισιμότητας των δεδομένων, τόσο μειώνεται η επίδοση όλων των μεθόδων.

4.4 Πειράματα με Πραγματικά Δεδομένα υψηλής διάστασης

Σε αυτή την Ενότητα θα δώσουμε τα πειραματικά αποτελέσματα που έγιναν τόσο σε βιολογικά δεδομένα [16], όσο και σε ευρέως χρησιμοποιούμενα πραγματικά δεδομένα απο άλλες εφαρμογές [12], [13].

4.4.1 Πραγματικά Βιολογικά Δεδομένα υψηλής διάστασης

Τα βιολογικά δεδομένα με τα οποία πειραματιστήκαμε είναι αυτά που περιγράφονται στον Πίνακα 4.2. Συγκεκριμμένα για κάθε σύνολο δεδομένων τα πειράματα έγιναν ως εξής:

Θεωρούμε οτι η προβολή γίνεται σε διάσταση 20, ανεξάρτητα απο την αρχική διάσταση η οποία ποικίλει απο 2000 έως και 7129. Δημιουργούμε ένα ensemble από 200 τυχαίες προβολές και αντίστοιχες ομαδοποιήσεις. Στο τέλος μέσω της consensus function, παίρνουμε μια τελική ομαδοποίηση, που αξιολογείται με το NMI και το σφάλμα ταξινόμησης.

Για να έχουμε οσο το δυνατόν πιο αξιόπιστα αποτελέσματα, κάνουμε 50 πειράματα για κάθε σύνολο δεδομένων και στον πίνακα δίνουμε τον μέσο όρο και την τυπική απόκλιση των αποτελεσμάτων. Επίσης, πέρα απο το NMI δίνουμε και το ποσοστό επιτυχίας, που είναι το ποσοστό των προτύπων που ταξινομήθηκαν στη σωστή κατηγορία. Να σημειώσουμε, ότι αυτό διαφέρει από το NMI, καθώς το τελευταίο λαμβάνει υπόψιν του και το κατα πόσο αναμιγνύονται τα πρότυπα διαφορετικών ομάδων, ενώ το ποσοστό επιτυχίας είναι ένας δείκτης του πόσα προτύπα έχουν καταχωρηθεί στη σωστή ομάδα.

Για την κατασκευή του ensemble χρησιμοποιήσαμε τρεις διαφορετικές μεθοδολογίες. Τον αλγόριθμο global kmeans με την απλή και την τροποποιημένη συνάρτηση ομοιότητας (πειράματα M_1 και M_2, αντίστοιχα) και τον κλασικό αλγόριθμο ομαδοποίησης kmeans (πείραμα M_3). Λόγω του ότι τα δεδομένα είναι λίγα δεν χρειάστηκε να δοκιμάσουμε τον αλγόριθμο fast global kmeans. Ως consensus function χρησιμοποιήσαμε τη μέθοδο spectral clustering (Ενότητα 2.5). Η ομαδοποίηση στο τελευταίο βήμα της consensus function έγινε για διάφορες τιμές του πλήθους ομάδων k προκειμένου να εξετάσουμε αν μπορούμε να εκτιμήσουμε το πλήθος ομάδων αν και είναι γνωστό στα συγκεκριμένα δεδομένα.

Επίσης, για να μπορούμε να συγκρίνουμε ισότιμα, τους διάφορους αλγορίθμους ομαδοποίησης, οι πίνακες προβολής που χρησιμοποιήσαμε σε όλα τα βήματα ήταν οι ίδιοι, και έτσι η στοχαστικότητα που χαρακτηρίζει τις προβολές επηρεάζει τα πειράματα κατά όμοιο τρόπο.

Στους παρακάτω πίνακες δίνουμε τα αποτελέσματα των μετρήσεων για όλα τα σύνολα δεδομένων του Πίνακα 4.2. Δίνουμε τον μέσο όρο των μετρήσεων και σε παρένθεση την τυπική απόκλιση για 50 πειράματα. Το NMI είναι αυτό που

περιγράψαμε σε προηγούμενη Ενότητα και μετράει το πόσο καλά συμφωνεί η ομαδοποίηση των δεδομένων μας με την πραγματική ομαδοποίηση των δεδομένων που γνωρίζουμε. Τέλος, το succ είναι το ποσοστό επιτυχίας της ομαδοποίησης, που προέκυψε για κάθε πείραμα.

Για το σύνολο δεδομένων της σήψης, έχουμε τα εξής αποτελέσματα:

		k in Consensus Function						
		k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9
M_1	NMI	0.2144 (0.0562)	0.2137 (0.0379)	0.2645 (0.0408)	0.2800 (0.0420)	0.2885 (0.0496)	0.2869 (0.0430)	0.2927 (0.0399)
	succ	0.5040 (0.0580)	0.5544 (0.0419)	0.6702 (0.0474)	0.7001 (0.0365)	0.6965 (0.0397)	0.6962 (0.0375)	0.7098 (0.0333)
M_2	NMI	0.1842 (0.0304)	0.2189 (0.0564)	0.2676 (0.0531)	0.2339 (0.0448)	0.2492 (0.0474)	0.2781 (0.0410)	0.2904 (0.0367)
	succ	0.5539 (0.0482)	0.5803 (0.0704)	0.6308 (0.0600)	0.6282 (0.0639)	0.6661 (0.0441)	0.6926 (0.0385)	0.7018 (0.0377)
M_3	NMI	0.2256 (0.0504)	0.2266 (0.0426)	0.2585 (0.0466)	0.2804 (0.0468)	0.2790 (0.0410)	0.2974 (0.0483)	0.3028 (0.0362)
	succ	0.5271 (0.0547)	0.5692 (0.0469)	0.6739 (0.0440)	0.6959 (0.0330)	0.6888 (0.0350)	0.7066 (0.0403)	0.7142 (0.0331)

Πίνακας 4.5: Αποτελέσματα ομαδοποίησης στο σύνολο δεδομένων σήψης (N = 71, d = 4857, k_{true} = 3).

Απο τα παραπάνω αποτελέσματα, βλέπουμε οτι και οι τρεις τεχνικές ομαδοποίησης που χρησιμοποιήσαμε, δεν απέχουν πολύ μεταξύ τους, με την τρίτη μέθοδο όπου χρησιμοποιούμε τον απλό αλγόριθμο kmeans να υπερτερεί ελαφρά έναντι των άλλων. Επίσης, αν έπρεπε να προβλέψουμε πόσες ομάδες σχηματίζουν τα δεδομένα αυτά με βάση το NMI, τότε μια καλή πρόβλεψη θα ήταν τιμές του k μεταξύ 3 και 5, μιας και εκεί έχουμε μια καλή αύξηση του NMI σε σχέση με τα κέντρα ομάδων που εισάγουμε.

Να σημειώσουμε εδώ, ότι είναι αναμενόμενο η τιμή του NMI να αυξάνει όσο αυξάνουν οι ομάδες που δίνουμε, αλλά με βάση το αξίωμα του Occam (Occam's razor), πρέπει να επιλέξουμε να αυξήσουμε τις ομάδες μόνο αν κάτι τέτοιο δικαιολογείται απο μια μεγάλη σχετικά αύξηση στην επίδοση της ομαδοποίησης.

Το επόμενο σύνολο δεδομένων που πειραματιστήκαμε, είναι το σύνολο δεδομένων που αφορά ασθενείς με λευχαιμία.

		k in Consensus Function					
		k = 2	k = 3	k = 4	k = 5	k = 6	k = 7
M_1	NMI	0.3533 (0.1813)	0.4735 (0.0967)	0.5072 (0.0601)	0.4108 (0.0699)	0.3923 (0.0480)	0.3714 (0.0543)
	succ	0.7753 (0.1631)	0.9123 (0.0380)	0.9206 (0.0343)	0.8541 (0.0592)	0.8578 (0.0541)	0.8620 (0.0542)
M_2	NMI	0.3454 (0.2045)	0.4495 (0.1014)	0.4739 (0.0746)	0.4423 (0.0644)	0.4413 (0.0665)	0.4043 (0.0766)
	succ	0.7249 (0.1886)	0.8929 (0.0457)	0.8819 (0.0678)	0.8811 (0.0523)	0.9010 (0.0426)	0.8892 (0.0614)
M_3	NMI	0.3463 (0.2091)	0.4508 (0.0904)	0.5028 (0.0638)	0.4170 (0.0611)	0.3902 (0.0600)	0.3653 (0.0657)
	succ	0.7504 (0.1861)	0.9073 (0.0336)	0.9146 (0.0415)	0.8648 (0.0467)	0.8564 (0.0593)	0.8426 (0.0833)

Πίνακας 4.6: Αποτελέσματα ομαδοποίησης στο σύνολο δεδομένων της λευχαιμίας (N = 38, d = 7129, k_{true} = 2).

Η μέθοδος ομαδοποίησης που υπερτερεί σε όλες τις περιπτώσεις είναι η απο εμάς προτεινόμενη χρήση της τροποποιημένης απόστασης με βάση τον αλγόριθμο global kmeans. Επίσης, φαίνεται καθαρά, και σε αυτό συμφωνούν όλες οι μέθοδοι, ότι μια καλή επιλογή στο πλήθος των ομάδων είναι k=3.

Τώρα, θα δούμε τα αποτελέσματα ομαδοποίησης που πήραμε στο ίδιο σύνολο δεδομένων, αν ομαδοποιήσουμε τα δεδομένα με βάση μόνο τα 50 πιο σημαντικά χαρακτηριστικά τους, δηλαδή αφού έχουν υποστεί μια προεπεξεργασία.

		k in Consensus Function					
		k = 2	k = 3	k = 4	k = 5	k = 6	k = 7
M_1	NMI	0.1905 (0.0860)	0.2416 (0.0242)	0.2309 (0.0185)	0.2409 (0.0284)	0.2621 (0.0197)	0.2540 (0.0222)
	succ	0.7055 (0.0982)	0.7669 (0.0128)	0.7626 (0.0083)	0.7514 (0.0232)	0.7552 (0.0184)	0.7553 (0.0307)
M_2	NMI	0.2192 (0.1029)	0.2529 (0.0372)	0.2759 (0.0297)	0.2884 (0.0216)	0.2753 (0.0237)	0.2632 (0.0283)
	succ	0.7254 (0.0793)	0.7632 (0.0222)	0.7849 (0.0082)	0.7510 (0.0061)	0.7690 (0.0163)	0.7538 (0.0212)
M_3	NMI	0.1753 (0.1249)	0.2398 (0.0301)	0.2282 (0.0188)	0.2249 (0.0156)	0.2469 (0.0193)	0.2663 (0.0146)
	succ	0.6792 (0.1196)	0.7638 (0.0109)	0.7589 (0.0060)	0.7634 (0.0074)	0.7664 (0.0259)	0.7566 (0.0171)

Πίνακας 4.7: Αποτελέσματα ομαδοποίησης στα 50 επικρατέστερα χαρακτηριστικά του συνόλου δεδομένων λευχαιμίας N = 38, d = 50, k_{true} = 2).

Φαίνεται οτι ο αλγόριθμος ομαδοποίησης global kmeans (η μέθοδος M_2) υπερτερεί των άλλων δυο. Και πάλι, όμως επιβεβαιώνεται οτι μια καλή επιλογή του πλήθους των ομάδων είναι ο αριθμός τρία.

Το προτελευταίο από τα σύνολα βιολογικών δεδομένων που αναλύσαμε, είναι αυτό που αφορά ασθενείς με καρκίνο παχέος εντέρου.

		k in Consensus Function							
		k = 2	k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9
M_1	NMI	0.0326 (0.0161)	0.2099 (0.0425)	0.2159 (0.0343)	0.1784 (0.0224)	0.1739 (0.0362)	0.2025 (0.0521)	0.2220 (0.0442)	0.2311 (0.0472)
	succ	0.5000 (0.0000)	0.7521 (0.0259)	0.7576 (0.0247)	0.7452 (0.0179)	0.7380 (0.0309)	0.7660 (0.0545)	0.7893 (0.0460)	0.8073 (0.0380)
M_2	NMI	0.0504 (0.0205)	0.0527 (0.0268)	0.1442 (0.0537)	0.1592 (0.0536)	0.2491 (0.0548)	0.2648 (0.0532)	0.2750 (0.0615)	0.2546 (0.0508)
	succ	0.5000 (0.0000)	0.5816 (0.0754)	0.6940 (0.0692)	0.7228 (0.0436)	0.8055 (0.0448)	0.8128 (0.0448)	0.8250 (0.0504)	0.8170 (0.0503)
M_3	NMI	0.0220 (0.0121)	0.1828 (0.0644)	0.1972 (0.0301)	0.1901 (0.0238)	0.1987 (0.0407)	0.2049 (0.0435)	0.2792 (0.0446)	0.2747 (0.0521)
	succ	0.5000 (0.0000)	0.7399 (0.0524)	0.7473 (0.0213)	0.7436 (0.0163)	0.7775 (0.0439)	0.7861 (0.0466)	0.8401 (0.0326)	0.8365 (0.0409)

Πίνακας 4.8: Αποτελέσματα ομαδοποίησης στο συνολο δεδομένων καρκίνου του παχαίους εντέρου (N = 62, d = 2000, k_{true} = 2).

Με βάση τα αποτελέσματα, οι ομάδες φαίνεται να είναι τουλάχιστον τρεις, αφού το NMI εκτοξεύεται από τιμές της τάξεως 0,003 για δύο ομάδες σε τιμές της τάξεως του 0,2 για τρείς ομάδες.

Το τελευταίο από τα βιολογικά σύνολα δεδομένων είναι αυτό του κληρονομικού καρκίνου του μαστού.

		k in Consensus Function						
		k = 3	k = 4	k = 5	k = 6	k = 7	k = 8	k = 9
M_1	NMI	0.1577 (0.0000)	0.2200 (0.0520)	0.2511 (0.0410)	0.3127 (0.0480)	0.3492 (0.0417)	0.3857 (0.0253)	0.4218 (0.0568)
	succ	0.4167 (0.0000)	0.5940 (0.0632)	0.6333 (0.0341)	0.6879 (0.0324)	0.7201 (0.0327)	0.7249 (0.0205)	0.7535 (0.0420)
M_2	NMI	0.1577 (0.0000)	0.1936 (0.0000)	0.2866 (0.0444)	0.3162 (0.0255)	0.3640 (0.0357)	0.3375 (0.0385)	0.3493 (0.0450)
	succ	0.4167 (0.0000)	0.4643 (0.0000)	0.5562 (0.0364)	0.6026 (0.0532)	0.7317 (0.0262)	0.7151 (0.0302)	0.7265 (0.0308)
M_3	NMI	0.2203 (0.0643)	0.2684 (0.0736)	0.3281 (0.0608)	0.3640 (0.0400)	0.3993 (0.0434)	0.4426 (0.0514)	0.4622 (0.0411)
	succ	0.5642 (0.0290)	0.6531 (0.0517)	0.6924 (0.0408)	0.7020 (0.0360)	0.7313 (0.0326)	0.7631 (0.0409)	0.7852 (0.0358)

Πίνακας 4.9: Αποτελέσματα ομαδοποίησης στο σύνολο δεδομένων του κληρονομικού καρκίνου του μαστού (N = 22, d = 3226, k_{true} = 3).

Ο απλός αλγόριθμος kmeans φαίνεται να λειτουργεί πιο αποδοτικά σε σχέση με τους άλλους δυο. Μια καλή εκτίμηση του πλήθους των ομάδων, με βάση τα αποτελέσματα των ομαδοποιήσεών μας είναι τρεις ή τέσσερις ομάδες.

4.4.2 Πειράματα με άλλα πραγματικά δεδομένα

Εδώ παραθέτουμε τα αποτελέσματα για τα σύνολα δεδομένων του Πίνακα 4.3. Τα ίδια σύνολα δεδομένων χρησιμοποίησαν οι Fern και Brodley στις εργασίες τους και μπορεί να τα βρεί κανείς στο διαδίκτυο στις τοποθεσίες [8] και [9].

Για τα πειράματά μας, χρησιμοποιούμε συνολικά τέσσερις τεχνικές ομαδοποίησης, που όλες τους στηρίζονται στην λογική των cluster ensembles, που έχουμε αναλύσει στο δεύτερο Κεφάλαιο. Στις πρώτες τρεις τεχνικές εφαρμόζουμε στο πρώτο βήμα, το ensemble constructor, τον αλγόριθμο kmeans σε μια απο τρείς παραλλαγές του – fast global kmeans, παραλλαγή του fast global kmeans και τον απλό kmeans (πειράματα M_1, M_2, M_3, αντίστοιχα)– ενώ στο δεύτερο βήμα, την consensus function, χρησιμοποιούμε τον πίνακα αποστάσεων που παράγουν οι πρώτοι αλγόριθμοι και κάνουμε ομαδοποίηση πάντα με τον αλγόριθμο spectral clustering, που περιγράψαμε επίσης στο δεύτερο Κεφάλαιο. Η τέταρτη τεχνική είναι να χρησιμοποιήσουμε τον αλγόριθμο ομαδοποίησης EM [6] στο ensemble constructor

και να κάνουμε agglomerative ομαδοποίηση στο consensus function, τεχνικές που εξηγήσαμε στην Ενότητα 3.2 (πείραμα M_4).

Επίσης, για να είμαστε αντικειμενικοί με τα αποτελέσματά μας, όπως και στην περίπτωση των βιολογικών δεδομένων, οι πίνακες προβολής είναι κοινοί για όλα τα πειράματα. Οι προβολές γίνονται σε διάσταση d_{low}, που δίνεται για κάθε σύνολο δεδομένων στους πίνακες που ακολουθούν. Η διάσταση προβολής είναι ίδια με αυτήν που είχαν επιλέξει οι Fern και Brodley, που σαν κριτήριο είχαν να διατηρείται η διακύμανση των στοιχείων κατά 80%, και υπολογίστηκε μέσω εφαρμογής του αλγορίθμου μείωσης διάστασης PCA.

Επίσης, η ομαδοποίηση στο πρώτο βήμα (ensemble constructor) γίνεται σε k_0 ομάδες, όπως έκαναν και οι Fern και Brodley, και τελικά στην consensus function κάνουμε ομαδοποίηση σε k_{true} ομάδες. Οι τιμές των παραπάνω παραμέτρων, για κάθε σύνολο δεδομένων, δίνονται στους πίνακες που ακολουθούν.

Στους πίνακες που ακολουθούν δίνουμε τα αποτελέσματα του NMI των ομαδοποιήσεων απο 10 πειράματα και σε παρένθεση την τυπική απόκλιση απο την μέση τιμή. Οι προβολές που κάναμε για κάθε cluster ensemble είναι 50.

Το πρώτο σύνολο δεδομένων που θα εξετάσουμε τα αποτελέσματα ομαδοποίησης του, είναι το CHART σύνολο δεδομένων.

	NMI	Succ
M_1	0.8247 (0.0066)	0.6659 (0.0011)
M_2	0.8341 (0.0081)	0.6697 (0.0155)
M_3	0.8313 (0.0092)	0.6708 (0.0172)
M_4	0.5003 (0.0083)	0.3331 (0.0016)

Πίνακας 4.10: Αποτελέσματα ομαδοποίησης CHART συνόλου δεδομένων ($N = 600$, $d = 60$, $d_{low} = 5$, $k_0 = 10$, $k_{true} = 6$).

Οι τρεις πρώτες μεθοδολογίες ομαδοποίησης που χρησιμοποιήσαμε είχαν τα καλύτερα αποτελέσματα, ενώ ο συνδυασμός του EM με τον agglomerative αλγόριθμο ομαδοποίησης, είχαν αρκετά πιο φτωχά αποτελέσματα σε σύγκριση με τους πρώτους.

Το αμέσως επόμενο σύνολο δεδομένων που θα εξετάσουμε είναι το MFEAT σύνολο χειρόγραφων χαρακτήρων.

($N = 600$, $d = 60$, $d_{low} = 5$, $k_0 = 10$, $k_{true} = 6$)

	NMI	Succ
M_1	0.6145 (0.0200)	0.6729 (0.0300)
M_2	0.6106 (0.0163)	0.6555 (0.0289)
M_3	0.6159 (0.0165)	0.6648 (0.0279)
M_4	0.5642 (0.0256)	0.5223 (0.0483)

Πίνακας 4.11: Αποτελέσματα ομαδοποίησης MFEAT συνόλου δεδομένων (N = 2000, d = 76, d_{low} = , k_0 = 15, k_{true} = 10).

Βλέπουμε και πάλι, οτι οι ομαδοποιήσεις που στηρίζονται στον kmeans δίνουν καλύτερα αποτελέσματα σε σχέση με τον EM, χωρίς κάποια απο τις τρεις πρώτες μεθόδους να υπερτερεί καθαρά έναντι των άλλων δυο.

Στο SATIMAGE σύνολο δεδομένων τα αποτελέσματα ομαδοποίησης, έχουν ως εξής:

	NMI	Succ
M_1	0.7465 (0.0338)	0.7646 (0.0344)
M_2	0.7351 (0.0171)	0.7566 (0.0375)
M_3	0.7234 (0.0341)	0.7458 (0.0276)
M_4	0.7301 (0.0527)	0.7357 (0.0607)

Πίνακας 4.12: Αποτελέσματα ομαδοποίησης SATIMAGE συνόλου δεδομένων (N = 4435, d = 36, d_{low} = 5, k_0 = 15, k_{true} = 6).

Στην περίπτωση αυτή, ο συνδυασμός του EM με τον agglomerative αλγόριθμο ομαδοποίησης συγκλίνει σε ποιότητα στα αποτελέσματα των άλλων τριών μεθόδων, αν και πάλι είναι ελαφρώς χειρότερα. Τα καλύτερα αποτελέσματα τα παίρνουμε κατα μέσο όρο με την παραλλαγμένη μεθοδολογία του kmeans (μέθοδο M1) που έχουμε προτείνει.

Τέλος, στο SEGMENTATION σύνολο δεδομένων τα αποτελέσματα ομαδοποίησης που πήραμε με όλες τις παραπάνω μεθοδολογίες, φαίνονται στον αμέσως επόμενο Πίνακα 4.13.

	NMI (st.d.)	Succ (st.d.)
M_1	0.5107 (0.0444)	0.5716 (0.0288)
M_2	0.5289 (0.0584)	0.5863 (0.0646)
M_3	0.5225 (0.0425)	0.5677 (0.0417)
M_4	0.5549 (0.1056)	0.3994 (0.0612)

Πίνακας 4.13: Αποτελέσματα ομαδοποίησης SEGMENTATION συνόλου δεδομένων ($N = 2310$, $d = 19$, $d_{low} = 5$, $k_0 = 15$, $k_{true} = 7$).

Βλέπουμε ότι, ο συνδυασμός του EM με τον agglomerative (μέθοδος M_4) για πρώτη φορά υπερτερεί έναντι των άλλων μεθόδων, αν και με ένα μικρό μόνο προβάδισμα.

4.4.3 Πειράματα με πραγματικά δεδομένα υψηλής διάστασης χωρίς προβολή

Τα τελευταία πειράματα που θα δούμε είναι πειράματα στα δεδομένα CHART, MFEAT και SEGMENTATION που περιγράψαμε στην Ενότητα 4.4.2. Η διαφορά εδώ είναι ότι δεν κάνουμε μείωση της διάστασης του προβλήματος. Συγκεκριμμένα θα δούμε ομαδοποιήσεις με τρεις διαφορετικούς τρόπους.

Στην πρώτη περίπτωση (πείραμα M_1) κάνουμε απευθείας εφαρμογή του αλγόριθμου spectral clustering στα δεδομένα, όπως περιγράφτηκε στην Ενότητα 2.5. Επειδή για τον ορισμό του πίνακα ομοιότητας A χρησιμοποιούμε την παράμετρο σ, εφαρμόζουμε τον αλγόριθμο για διάφορες τιμές του σ, που φαίνονται παρακάτω στους πίνακες. Στο δεύτερο πείραμα (M_2) χρησιμοποιούμε και πάλι τον αλγόριθμο spectral clustering, μόνο που τώρα ο πίνακας ομοιοτήτων A, προκύπτει μέσω του εναλλακτικού ορισμού της ομοιότητας που υπολογίζεται με την βοήθεια του αλγόριθμου ομαδοποίησης global kmeans, όπως τον περιγράψαμε στην Ενότητα 3.4. Στον υπολογισμό της ομοιότητας και στην εφαρμογή του αλγόριθμου global kmeans, χρησιμοποιούμε ως μέγιστο επιτρεπτό πλήθος κέντρων το διπλάσιο και τριπλάσιο, αντίστοιχα, του πραγματικού. Τέλος, στο πείραμα M_3, εκτελούμε μόνο τον αλγόριθμο global kmeans.

Στον παρακάτω Πίνακα 4.14, βλέπουμε τα αποτελέσματα ομαδοποίσης για το σύνολο δεδομένων CHART, 600 προτύπων, διάστασης 60 με 6 ομάδες.

Πείραμα		NMI	Succ
M_1	$\sigma = 10$	0.8615	0.6667
	$\sigma = 5$	0.4483	0.4900
	$\sigma = 2$	0.0579	0.2683
M_2	$k_{max} = 2 \times k_{true}$	0.8111	0.7333
	$k_{max} = 3 \times k_{true}$	0.7986	0.8067
M_3		0.7670	0.6450

Πίνακας 4.14: Ομαδοποίηση του συνόλου δεδομένων CHART χωρίς προβολή ($N = 600$, $d = 60$, $k_{true} = 6$).

Το καλύτερο αποτέλεσμα ομαδοποίησης το δίνει το πείραμα M_1 μαζί με το M_2, αν και στο M_2 το ποσοστό επιτυχίας είναι αρκετά βελτιωμένο σε σχέση με το M_1. Βλέπουμε ότι, το NMI πέφτει αρκετά οταν μειώνεται το σ του αλγόριθμου spectral clustering και ότι οι αλγόριθμοι που μειώνουν τη διάσταση στην προηγούμενη Ενότητα πετυχαίνουν καλά αποτελέσματα ομαδοποίησης.

Το επόμενο σύνολο δεδομένων προς εξέταση είναι το MFEAT, του οποίου τα αποτελέσματα ομαδοποίησης φαίνονται στον Πίνακα 4.15.

Πείραμα		NMI	Succ
M_1	$\sigma = 10$	0.0635	0.2145
	$\sigma = 5$	0.6177	0.6580
	$\sigma = 2$	0.6292	0.6585
	$\sigma = 1.5$	0.6370	0.7080
	$\sigma = 1$	0.6436	0.7120
	$\sigma = 0.5$	0.6677	0.7230
M_2	$k_{max} = 2 \times k_{true}$	0.6516	0.6870
	$k_{max} = 3 \times k_{true}$	0.6321	0.6675
M_3		0.6280	0.6640

Πίνακας 4.15: Ομαδοποίηση του συνόλου δεδομένων MFEAT χωρίς προβολή ($N = 2000$, $d = 76$, $k_{true} = 10$).

Βλέπουμε εδώ, οτι όλες οι προτεινόμενες μέθοδοι έχουν παρόμοια αποτελέσματα ομαδοποίησης, που είναι επίσης πολύ κοντά με τα αποτελέσματα ομαδοποίησης με χρήση πινάκων προβολής της Ενότητας 4.4.2. Τα αποτελέσματα για το τελευταίο σύνολο δεδομένων, το SEGMENTATION, δίνονται στον Πίνακα 4.16.

Πείραμα		NMI	Succ
M_1	$\sigma = 30$	0.0402	0.1498
M_2	$k_{max} = 2 \times k_{true}$	0.5608	0.5619
	$k_{max} = 3 \times k_{true}$	0.5632	0.5671
M_3		0.4880	0.5359

Πίνακας 4.16: Ομαδοποίηση του συνόλου δεδομένων SEGMENTATION χωρίς προβολή ($N = 2310$, $d = 19$, $k_{true} = 7$).

Εδώ πέρα και πάλι τα αποτελέσματα ομαδοποίησης είναι αρκετά όμοια μεταξύ τους, και πολύ κοντά στα αποτελέσματα ομαδοποίησης που είχαμε πάρει με χρήση τυχαίων προβολών στην προηγούμενη Ενότητα. Εξαίρεση αποτελεί το πείραμα M_1, για το οποίο δεν πετυχαίνουμε ικανοποιητικό αποτέλεσμα ομαδοποίησης, λόγω της μεγάλης τιμής του σ. Αλλά και με μικρότερη τιμή για την παράμετρο σ, ο αλγόριθμος spectral clustering δεν συνέκλινε σε λύση.

5

ΚΕΦΑΛΑΙΟ 5. ΣΥΜΠΕΡΑΣΜΑΤΑ – ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ

Κλείνοντας αυτή την εργασία θα θέλαμε να αναφερθούμε σε κάποια συμπεράσματα που βγαίνουν απο τις υλοποιήσεις και τα πειράματα που κάναμε. Καταρχήν, αυτό που πρέπει να πούμε είναι οτι οι τεχνικές που εξετάστηκαν είναι σχετικά καινούριες. Τόσο, οι τυχαίες προβολές, τα cluster ensembles, ο αλγόριθμος του spectral clustering, όσο και οι διαφορετικές εκδοχές του αλγόριθμου kmeans δεν έχουν ακόμα δοκιμαστεί αρκετά στην πράξη, πολύ περισσότερο οι συνδυασμοί αυτών.

Σε γενικές γραμμές φαίνεται πως οι τυχαίες προβολές λειτουργούν ικανοποιητικά για ενα μεγάλο φάσμα δεδομένων, όπου δύσκολα μπορεί κανείς να αντιπροτείνει εναν μοναδικό αλγόριθμο μείωσης διάστασης που να έχει πάντα καλά αποτελέσματα. Επίσης, το ότι οι τυχαίες προβολές κάθε φορά αποκαλύπτουν μια διαφορετική πτυχή των δεδομένων μας, έχει θετική επίδραση όταν χρησιμοποιείται για ομαδοποίηση δεδομένων υψηλής διάστασης με την μεθοδολογία των cluster ensembles.

Τέλος, ο αλγόριθμος του spectral clustering, οδηγεί σε καλές επιδόσεις ομαδοποίησης δεδομένων. Όταν μάλιστα για τον υπολογισμό της ομοιότητας των σημείων χρησιμοποιείται ο αλγόριθμος kmeans ή η παραλλαγή που προτείνουμε στην Ενότητα 3.4 τότε συνήθως παρατηρούμε μια επιπλέον βελτίωση.

Οπότε, συμπερασματικά μπορούμε να πούμε ότι οι τυχαίες προβολές έχουν πολύ καλή επίδοση στην μείωση διάστασης οσοδήποτε μεγάλων προβλημάτων και ότι σε συνδυασμό με τα cluster ensembles, μπορούν να δώσουν ικανοποιητικά αποτελέσματα ομαδοποιήσεων. Ειδικότερα, στην περίπτωση των cluster ensembles, η εφαρμογή του αλγόριθμου spectral clustering ως consensus function, οδηγεί σε καλές επιδόσεις. Το μεγάλο όφελος απο την χρήση των αλγορίθμων των τυχαίων προβολών σε συνδυασμό με τον (απλό ή global ή fast global) αλγόριθμο kmeans είναι η ταχεία και ποιοτική δημιουργία ενός ensemble.

Βιβλιογραφία

[1] E. Bingham, H. Mannila: "*Random Projection in dimensionality reduction: Applications to image and text data*", ACM, 2001.

[2] C. Bishop: "*Neural Networks for Pattern Recognition*", Clarendon Press, Oxford, 1995.

[3] R. Bellman: "*Adaptive Control Processes: A Guided Tour*", New Jersey, Princeton University Press, 1961.

[4] T. Cover, J. Thomas: "*Elements of Information Theory*", Wiley, 1991.

[5] S. Dasgupta, A. Gupta: "*An elementary proof of the Johnson-Lindenstrauss Lemma*", International Computer Science Institute, 1999.

[6] A. Dempster, N. Laird, D. Rubin : "*Maximum likelihood estimation from incomplete data via the EM algorithm*", Journal of the Royal Statistical Society, Series B, 39 (1), pp. 1-38, 1977.

[7] F. Dmitriy, D. Madigan: "*Experiments with Random Projections for Machine Learning*", ACM, Conference on Knowledge Discovery in Data, pp.517-522, 2003.

[8] X. Fern, C. Brodley: "*Random Projection for High Dimensional Data Clustering: A Cluster Ensemble Approach*", Proceedings of 20th International Conference on Machine learning, ICML2003.

[9] X. Fern, C. Brodley: "*Cluster Ensembles for High Dimensional Clustering: An Empirical Study*", Technical report, 2004.

[10] X. Fern, C. Brodley: "*Solving cluster ensemble problems by bipartite graph partitioning*", Proceedings of the Twenty First International Conference on Machine Learning, pp. 281-288, 2004.

[11] S. Dasgupta: "*Learning Mixtures of Gaussians*", Proceedings of the 40th Annual Symposium on Foundations of Computer Science, 1999.

[12] S. Hettich, S. D. Bay, "*The UCI KDD Archive", [http://kdd.ics.uci.edu]*, University of California, Irvine, Department of Information and Computer Science, 1999.

[13] D. J. Newman, S. Hettich, C. L. Blake, C. J. Merz, "*UCI Repository of machine learning databases",*

[http://www.ics.uci.edu/~mlearn/MLRepository.html]. CA, University of California, Irvine, Department of Information and Computer Science, 1998.

[14] A. Likas, N. Vlassis, J. Verbeek: "*The global k-means clustering algorithm*", Pattern Recognition, vol. 36, pp. 451-461, 2003.

[15] A. Ng, M. Jordan, Y. Weiss: "*On spectral clustering: Analysis and an algorithm*", Advances in Neural Information Processing Systems 14, pp. 849-856, 2002.

[16] C. Tang, A. Zhang, M. Ramanathan: "*Automatic Phenotype Mining from Gene Expression Profiles*", http://www.cse.buffalo.edu/DBGROUP/bioinformatics/supplementary/EPD/download.html, Bioinformatics (20), pp. 829-838, 2004.

[17] N. Vlassis web page: http://staff.science.uva.nl/~vlassis

[18] Y. Weiss: "*Segmentation using eigenvectors: A unifying view*", International Conference on Computer Vision, 1999.

ΒΙΟΓΡΑΦΙΚΟ

Ο Στέφανος – Κωνσταντίνος Πουρσαλίδης γεννήθηκε το 1980 στο Μόναχο της Γερμανίας. Αποφοίτησε το 1998 απο το Λύκειο της ίδιας πόλης. Εισήχθει την ίδια χρονιά στο προπτυχιακό πρόγραμμα σπουδών του τμήματος Πληροφορικής του Πανεπιστημίου Ιωαννίνων και έλαβε πτυχίο το 2003. Στο ίδιο τμήμα παρακολούθησε το μεταπτυχιακό πρόγραμμα σπουδών απο το οποίο αποφοίτησε τον Νοέμβριο του 2005.