ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανίχνευση Αλλαγών και Συσταδοποίηση σε πραγματικό χρόνο

Γεράσιμος Ε. Σκαράκης

ΕΠΙΒΛΕΠΟΝΤΕΣ: **Ευστάθιος Χατζηευθημιάδης,** Αναπληρωτής Καθηγητής ΕΚΠΑ
Βασίλειος Παπαταξιάρχης, Υποψήφιος Διδάκτωρ

ΑΘΗΝΑ

ΜΑΪΟΣ 2015

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανίχνευση Αλλαγών και Συσταδοποίηση σε πραγματικό χρόνο

Γεράσιμος Ε. Σκαράκης

Α.Μ.: 1115200900123

ΕΠΙΒΛΕΠΟΝΤΕΣ: **Ευστάθιος Χατζηευθημιάδης,** Αναπληρωτής Καθηγητής ΕΚΠΑ
Βασίλειος Παπαταξιάρχης, Υποψήφιος Διδάκτωρ

ΠΕΡΙΛΗΨΗ

Καθώς ο όγκος των πληροφοριών που είναι διαθέσιμες αυξάνεται με εκθετικό ρυθμό με την πάροδο του χρόνου, προκύπτει η ανάγκη εξαγωγής της χρήσιμης γνώσης που βρίσκεται συσσωρευμένη στην πληροφορία. Η μελέτη τεχνικών μέσω των οποίων επιτυγχάνεται αυτό, αποτελεί ένα σημαντικό πεδίο έρευνας.

Σκοπός της εργασίας είναι να εξετάσω τους τρόπους με τους οποίους μπορώ να βρω αλλαγές σε χρονοσειρές με αριθμητικά δεδομένα σε πραγματικό χρόνο, χρησιμοποιώντας αλγορίθμους ανίχνευσης αλλαγών(change detection) και συσταδοποίησης (clustering). Με πειραματικές μελέτες αναλύω τα θετικά και αρνητικά της κάθε υλοποίησης και καταλήγω στις ποιο αποτελεσματικές μετρικές για τον στόχο αυτό.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Εξόρυξη Δεδομένων

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: ανίχνευση αλλαγών, συσταδοποίηση, αλγόριθμος

ABSTRACT

While the volume of information that is available increases with an exponential rate over time, the need to export the useful knowledge that is accumulated in the information. The study of the techniques through with which this is achieved, is an important field of research.
The purpose of this project is to look at the ways in which I can find changes in datasets with numerical data, using change detection and clustering algorithms. With experimental studies I analyze the pros and cons of each algorithm and I conclude to the most effective metrics for this purpose.

SUBJECT AREA: Data Mining

KEYWORDS: change detection, clustering, algorithm

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ 10

1. ΕΙΣΑΓΩΓΗ 11

1.1. Είδη Δεδομένων 11

1.1.1. Στατικά Δεδομένα 11

1.1.2. Δεδομένα Πραγματικού χρόνου 11

1.2. Περιγραφή Αλγορίθμων 11

1.1.3. Ανίχνευσης αλλαγής 11

1.1.4. Συσταδοποίησης 12

1.3. Οργάνωση της εργασίας 12

2. ΑΛΓΟΡΙΘΜΟΙ ΑΝΙΧΝΕΥΣΗΣ ΑΛΛΑΓΩΝ 13

2.1. Step Detection 14

2.2. Online αλγόριθμοι ανίχνευσης αλλαγών 15

2.2.1. Cumulative Sum 16

2.2.2. Προσαρμοστικός Cumulative Sum 16

2.2.3. Shewhart Controller 17

3. ΑΛΓΟΡΙΘΜΟΙ ΣΥΣΤΑΔΟΠΟΙΗΣΗΣ 19

3.1. Στατικοί αλγόριθμοι συσταδοποίησης 20

3.1.1. Συσταδοποίηση με βάση την συνδεσιμότητα 20

3.1.2. Συσταδοποίηση με βάση το κεντροειδές 21

3.1.3. Συσταδοποίηση με βάση την κατανομή των δεδομένων 22

3.1.4. Συσταδοποίηση με βάση την διασπορά 24

3.2. On-line Αλγόριθμοι Συσταδοποίησης 25

3.3. Μετρικές αποστάσεων 25

3.3.1. Ευκλείδεια Απόσταση 26

3.3.2. Απόσταση Manhattan 26

3.3.3. Απόσταση Minkowsky 27

3.3.4. Απόσταση Mahalanobis 27

4. ΜΕΘΟΔΟΛΟΓΙΕΣ – ΜΕΤΡΙΚΕΣ ΣΥΣΤΑΔΟΠΟΙΗΣΗΣ 28

4.1. Cumulative Sum πολλαπλων καταστάσεων 29

4.2. Silhouette Coefficient 29

4.3. Local Outlier Factor 31

4.4. Αλγοριθμος ελέγχου απόστασης συστάδων 33

4.5. Λεπτομέρειες Υλοποίησης 34
5. ΠΕΙΡΑΜΑΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ 35
5.1. Σύνολα δεδομένων που χρησιμοποιήθηκαν 35
5.2. Πειράματα – Αποτελέσματα 35
5.2.1. Μετρικές – Σενάρια που δοκιμάστηκαν 35
5.2.2. Αποτελέσματα/Συγκριση Αλγορίθμων 36
5.2.2.1. Αλγόριθμοι ανίχνευσης αλλαγής 36
5.2.2.2. Αλγόριθμοι συσταδοποίησης 38
5.3. Χρόνοι Απόκρισης Συστήματος 45
5.4. Συμπεράσματα 47
5.4.1. CuSum 47
5.4.2. Προσαρμοστικός CuSum 47
5.4.4. CuSum Πολλαπλών καταστάσεων 48
5.4.5. Local Outlier Factor 49
5.4.6. Silhouette Coefficient 49
5.4.7. Αλγοριθμος ελέγχου απόστασης συστάδων 50
ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ 51
ΑΝΑΦΟΡΕΣ 52

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 5.1: Συγκριτικά αποτελέσματα χρόνων απόκρισης συστήματος μεταξύ των αλγορίθμων LOF, Silhouette Coefficient και του συνδιασμού αυτών ανά dataset…….....46

Σχήμα 5.2: Συγκριτικά αποτελέσματα χρόνων απόκρισης συστήματος μεταξύ των αλγορίθμων Προσαρμοστικού CuSum, ελέγχου απόστασης συτάδων, Cusum πολλαπλών και Shewhart Controller καταστάσεων ανά dataset.....46

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Single-linkage σε συστάδες βασισμένες στην πυκνότητα. Εξάγονται 20 συστάδες, οι περισσότερες από τις οποίες περιέχουν μόνο ένα στοιχείο, δεδομένου ότι αυτή η ομαδοποίηση δεν έχει την έννοια του «θορύβου». 21
Εικόνα 2:K-means συσταδοποίηση 22
Εικόνα 3:Σε δεδομένα με κατανομή Gauss, ο EM λειτουργεί καλά, δεδομένου ότι χρησιμοποιεί Gaussians μοντέλα ομαδοποίησης 23
Εικόνα 4:Συστάδες βασισμένες στην διασπορά δεν μπορούν να μοντελοποιηθούν χρησιμοποιώντας Gaussian κατανομές.....23
Εικόνα 5:Ομαδοποίηση βασισμένη στην διασπορά με DBSCAN.....24
Εικόνα 6:LOF.....31
Εικόνα 7:Αποτέλεσμα CuSum Πολλαπλών καταστάσεων – METORQUE Dataset. (δεδομένα-χρόνος).....39
Εικόνα 8:Αποτέλεσμα CuSum Πολλαπλών καταστάσεων – METORQUE Dataset. (δεδομένα-συστάδες).....39
Εικόνα 9:Αποτέλεσμα Local Outlier Factor – METORQUE Dataset. (δεδομένα-χρόνος).....40
Εικόνα 10: Αποτέλεσμα Local Outlier Factor – METORQUE Dataset. (δεδομένα-συστάδες).....40
Εικόνα 11:Αποτέλεσμα Silhouette Coefficient – METORQUE Dataset. (δεδομένα-χρόνος).....41
Εικόνα 12:Αποτέλεσμα Silhouette Coefficient – METORQUE Dataset. (δεδομένα-συστάδες).....41
Εικόνα 13: Αποτέλεσμα Silhouette Coefficient+LOF – METORQUE Dataset. (δεδομένα-χρόνος).....42
Εικόνα 14: Αποτέλεσμα Silhouette Coefficient+LOF – METORQUE Dataset. (δεδομένα-συστάδες).....42
Εικόνα 15:Αποτέλεσμα Αλγόριθμος ελέγχου απόστασης – METORQUE Dataset. (δεδομένα-χρόνος).....43
Εικόνα 16: Αποτέλεσμα Αλγόριθμος ελέγχου απόστασης – METORQUE Dataset. (δεδομένα-συστάδες).....43
Εικόνα 17:Αποτέλεσμα Αλγόριθμος ελέγχου απόστασης(λιγότερο αυστηρός) – METORQUE Dataset. (δεδομένα-χρόνος) 44
Εικόνα 18: Αποτέλεσμα Αλγόριθμος ελέγχου απόστασης(λιγότερο αυστηρός) – METORQUE Dataset. (δεδομένα-συστάδες).....44

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1:Αποτελέσματα ανίχνευσης αλλαγών προσαρμοστικού CuSum 36
Πίνακας 2:Αποτελέσματα ανίχνευσης αλλαγών Shewhart Controller 37
Πίνακας 3:Αποτελέσματα συσταδοποίησης 38
Πίνακας 4:Χρόνοι Απόκρισης αλγορίθμων ανίχνευσης αλλαγών 45
Πίνακας 5:Χρόνοι Απόκρισης αλγορίθμων συσταδοποίησης.....45

ΠΡΟΛΟΓΟΣ

Η παρούσα πτυχιακή εργασία πραγματοποιήθηκε στο Εθνικό Καποδιστρια΄κο Πανεπιστήμιο Αθηνών, στο τμήμα Πληροφορικής και Τηλεπικοινωνιών. Στόχος αυτής της πτυχιακής είναι η μελέτη των τρόπων έβρεσης αλλαγών σε χρονοσειρές με αριθμητικά δεδομένα.

Θέλω ευχαριστήσω τον καθηγητή μου κ. Ευστάθιο Χατζηευθημιάδη που μου έδωσε την ευκαιρία να ασχοληθώ με ένα τόσο ενδιαφέρον .θέμα.
Επίσης θα ήθέλα να ευχαριστήσω τον επιβλέποντα υποψήφιο διδάκτορ Παπαταξιάρχη Βασίλειο, ο οποίος με βοήθησε πάρα πολύ ώστε να ολοκληρωθεί αυτή η εργασία. Τον ευχαριστώ πολύ για όλα όσα μου δίδαξε, για το επιστημονικό υλικό που μου προσέφερε, τις συμβουλές του και τις ώρες που μου αφιέρωσε.

1. ΕΙΣΑΓΩΓΗ

1.1. Είδη Δεδομένων

Η επεξεργασία δεδομένων από έναν προγραματιστή μπορεί να γίνει με δύο τρόπους, ανάλογα με τις απαιτήσεις.

1.1.1. Στατικά Δεδομένα

Στην περίπτωση που υπάρχει ένα σύνολο δεδομένων συγκεκριμένουν μεγέθους γνωστού εκ των προτέρων, ο χρήστης μπορεί να χρησιμοποιήσει κλασικούς αλγορίθμους επεξεργασίας στατικών δεδομένων, οι οποίοι πέρνουν υπ' όψη ολόκληρο τον όγκο των δεδομένων και βγάζουν ακριβή συμπεράσματα.

1.1.2. Δεδομένα Πραγματικού χρόνου

Όταν το σύνολο των δεδομένων δεν είναι στατικό, αλλά αυξάνεται με την πάροδο του χρόνου[1], τότε πρέπει να χρησιμοποιήσουμε αλγορίθμους που δεν τερματίζουν, οι οποίοι επεξεργάζονται τα δεδομένα αυτά σειριακά έτσι ώστε να μπορούν να δέχονται καινούργια δεδομένα που έρχονται σε τακτά χρονικά διαστήματα και να βγάζουν κατάλληλα αποτελέσματα.

1.2. Περιγραφή Αλγορίθμων

Οι αλγόριθμοι που εξετάζονται σε αυτήν την εργασία δέχονται δεδομένα πραγματικού χρόνου και ονομάζονται σειριακοί ή online αλγόριθμοι. Ανάλογα με τον τρόπο που λειτουργούν και τα αποτελέσματα που εξάγουν μπορούν να κατηγοριοποιήθουν σε Αλγορίθμους Ανιχνευσης Αλλαγής (change detection algorithms) και Αλγορίθμους Συσταδοποίησης (clustering algorithms).

1.1.3. Ανίχνευσης αλλαγής

Οι αλγόριθμοι ανίχνευσης αλλαγής αυτοί έχουν στόχο να εντοπίζουν αν παρουσιάστηκε μια ή περισότερες αλλαγές, αλλά και να αναγνωρίζουν τις χρονικές στιγμές αυτών των αλλαγών.
Εφαρμογές τέτειων αλγορίθμων μπορεί να ασχολούνται με αλλαγές στην μέση τιμή, διακύμανση, συσχέτιση ή και φασματική πυκνότητα της διαδικασίας.

Αλγόριθμοι ανίχνευσης αλλαγής χρησιμοποιούνται συχνά για τον έλεγχο ποιότητας προϊόντων σε βιομηχανίες, για φιλτράρισμα ανεπιθύμητων μηνυμάτων, για ιατρικές διαγνώσεις και για ανίχνευση παραβιάσεων.

1.1.4. Συσταδοποίησης

Η συσταδοποίηση έχει ως στόχο τα αντικείμενα μέσα σε μια ομάδα να είναι όμοια ή να σχετίζονται μεταξύ τους με βάση κάποια χαρακτηριστικά, τα οποία καθορίζονται από τον εκάστοτε αλγόριθμο, και να είναι διαφορετικά ή μη σχετιζόμενα με τα αντικείμενα των άλλων ομάδων. Οι τρόποι που γίνεται αυτή η ομαδοποίηση διαφέρει από μετρική σε μετρική, ανάλογα με το είδος των ομοιοτήτων που ελέγχει η κάθε μια.

Τόσο η συσταδοποίηση, όσο και η on-line συσταδοποίηση είναι από τις πιο διαδεδομένες τεχνικές εξόρυξης δεδομένων που χρησιμοποιούνται στη στην επιστήμη των υπολογιστών, στην ιατρική, στην ψυχολογία, στην βιολογία, στη στατιστική και σε αρκετούς άλλους τομείς. Για παράδειγμα, η συσταδοποίηση μπορεί να χρησιμοποιηθεί στον τομέα του μάρκετινγκ για να βρεθούν ομάδες πελατών με παρόμοιες συμπεριφορές. Επίσης, στον ίδιο κλάδο η online συσταδοποίηση μπορεί να χρησιμοποιηθει για προτείνει στον πελάτη σε πραγματικό χρόνο προϊόντα που μπορούν να τον ενδιαφέρουν, αφού έχοντας λάβει υπ' όψη προϊγούμενες προτιμήσεις του. Αυτό το παράδειγμα παρατηρείται και στο διαδίκτυο σε γνωστές ιστοσελίδες όπως το www.youtube.com με τα προτινόμενα βίντεο.

1.3. Οργάνωση της εργασίας

Η παρούσα εργασία εστιάζεται στην ανάπτυξη ενός πλαισίου Η οργάνωση της εργασίας έχει ως εξής.

Στο κεφάλαιο 2 αναλύεται το θέμα της ανίχνευσης αλλαγών και περιγράφονται ο πιο γνωστός αλγόριθμος on-line ανίχνευσης αλλαγών καθώς και μια τροποποίηση του που έχει το χαρακτηριστικό ότι προσαρμόζεται στις αλλαγές και βγάζει περισσότερα συμπεράσματα. Στο κεφάλαιο 3 περιγράφονται γνωστοί αλγόριθμοι στατικής και on-line συσταδοποίησης, ενώ στο κεφάλαιο 4 παρουσιάζονται οι μετρικές συσταδοποίησης που υλοποιούνται στην εργασία αυτή και περιγράφονται αναλυτικά.

Η παρούσα εργασία ολοκληρώνεται με το κεφάλαιο 5 στο οποίο δίνονται συμπεράσματα της όλης μελέτης και κάποια για μελέτη θέματα που αφορούν στο συγκεκριμένο εξεταζόμενο πεδίο έρευνας.

2. ΑΛΓΟΡΙΘΜΟΙ ΑΝΙΧΝΕΥΣΗΣ ΑΛΛΑΓΩΝ

Στην στατιστική ανάλυση, η ανίχνευση αλλαγών ή ανίχνευση σημείου αλλαγής προσπαθεί να εντοπίσει τα χρονικά σημεία όπου η κατανομή πιθανότητας μιας στοχαστικής διαδικασίας ή χρονοσειράς αλλάζει. Σε γενικές γραμμές το πρόβλημα αφορά τόσο την ανίχνευση του αν έχει συμβεί ή όχι μια ή περισσότερες τέτοιες αλλαγές, και τον εντοπισμό των χρόνου εμφάνισης αυτών των αλλαγών.

Συγκεκριμένες εφαρμογές, όπως step detection, ασχολούνται με τις αλλαγές στην μέση τιμή, την διακύμανση, την συσχέτιση, ή την φασματική πυκνότητα της διαδικασίας. Γενικότερα στην τεχνική αυτή ανοίκει και η ανίχνευση ανώμαλων συμπεριφορών: ανίχνευση ανωμαλιών[2].

2.1. Step Detection

Step Detection (ανίχνευση βήματος) [γνωστή και ως step smoothing (λείανση βήματος), step filtering(φιλτράρισμα βήματος), shift detection (ανίχνευση μετατόπισης), jump detection (ανίχνευση πηδήματος) ή edge detection (ανίχνευση άκρης)] αποτελεί την διαδικασία εντοπισμού ξαφνικών αλλαγών (βημάτων, μετατοπίσεων, πηδημάτων) σε μια χρονοσειρά ή κάποιο σήμα. Συχνά, το βήμα αυτό είναι μικρό και οι χρονοσειρές έχουν αλλοιωθεί εξαιτίας κάποιου είδους θορύβου, και αυτό καθιστά το πρόβλημα δύσκολο, διότι το βήμα μπορεί να κρυφτεί λόγω του θορύβου.

Το πρόβλημα ανίχνευσης βήματος εμφανίζεται σε μεγάλο αριθμό επιστημονικών και μηχανικών πλαισίων, μερικά από τα οποία αποτελούν την γενετική, την βιοφυσική και την επεξεργασία εικόνων.

Οι περισσότεροι offline αλγόριθμοι για στην ανίχνευση βήματος ψηφιακών δεδομένων μπορούν να κατηγοριοποιηθούν ως top-down, bottom-up, sliding window, ή global μέθοδοι.

Top-down μέθοδος

Αυτοί οι αλγόριθμοι ξεκινούν με την παραδοχή ότι δεν υπάρχουν βήματα και εισάγουν πιθανά βήματα ένα την φορά, τα οποία ελέγχουν για να βρουν αυτό το οποίο ελαχιστοποιεί κάποια κριτήρια.

Bottom-up μέθοδος

Εδώ οι αλγόριθμοι λειτουργούν με την ακριβώς αντίθετη λογική από τους top-down. Αρχικά θεωρούν ότι υπάρχει βήμα μεταξύ όλων των δειγμάτων ενός ψηφιακού σήματος, και στην συνέχεια συγχωνεύουν βήματα σύμφωνα με κάποια κριτήρια με τα οποία ελέγχονται όλα τα βήματα.

Sliding window μέθοδος

Σε αυτή την μέθοδο εξετάζεται το σήμα μέσω ενός παραθύρου. Οι αλγόριθμοι ψάχνουν για στοιχεία βήματος που εμφανίζονται εντός του παραθύρου, το οποίο μετά από κάθε ολοκληρωμένο έλεγχο μετατοπίζεται(σείρεται) κατα μία θέση κάθε φορά, όπου και ξαναγίνεται έλεγχος μέσω αυτού. Την μέθοδο αυτή ακολουθούν αρκετά φίλτρα που έχουν στόχο την μείωση του θορύβου διατηρώντας όμως τα απότομα βήματα του σήματος.

Global μέθοδος

Οι αλγόριθμοι αυτοί εξετάζουν το σήμα με μια κίνηση και προσπαθούν να εντοπίσουν βήματα στο σήμα με κάποιο είδος διαδικασίας βελτιστοποίησης.

2.2. Online αλγόριθμοι ανίχνευσης αλλαγών

Όταν η ανίχνευση βήματος πρέπει να εκτελείται με το που φτάνουν τα δεδομένα, τότε βρισκόμαστε στην περίπτωση της σειριακής ανάλυσης.

Στην σειριακή ανάλυση, κάθε αλγόριθμος παρουσιάζει 3 αρνητικά χαρακτηριστηκά αντιστρόφως ανάλογα μεταξύ τους:

- Ποσοστό αρνητικών συναγερμών(false alarm)
- Ποσοστό αποτυχίας εντοπισμού αλλαγής(misdetection)
- Καθυστέρηση ανίχνευσης

Ο κάθε αλγόριθμος στοχεύει στο να ελαχιστοποιήσει αυτά τα χαρακτηρηστικα χωρίς όμως να μπορεί να εξαλείψει και τα 3.

Οι περισσότεροι online αλγόριθμοι ανίχνευσης αλλαγών είναι και ταυτόχρονα αλγόριθμοι συσταδοποίησης, καθώς κάθε αλγόριθμος συσταδοποίησης μπορεί να τροποποιηθεί για ανίχνευση αλλαγών.

Ωστόσω υπάρχουν και αυθεντικοί αλγόριθμοι online ανίχνευσης αλλαγών, με πιο δημοφιλή τον CuSum. Επίσης πολύ συχνα χρησιμοποιούνται παραλλαγές του CuSum, ο οποίος τροποποιείται για να καλύψει τις ανάγκες του κάθε χρήστη. Μια τέτοια παραλλαγή υλοποιήθηκε στα πλαίσια αυτής της εργασίας και αναλύεται παρακάτω.

2.2.1. Cumulative Sum

Ο αλγόριθμος Cumulative Sum ή αλλιώς Cusum είναι ένας σειριακός αλγόριθμος ελέγχου αλλαγής κατάστασης, και συγκεκριμένα ο ποιο γνωστός του είδους.
Όπως λέει και το όνομά του, ο Cusum υπολογίζει το συσωρευτικό άθροισμα των στοιχείων μιας διαδικασίας και αυτό είναι που τον κάνει σειριακό. Τα στοιχεία X_n αναθέτονται βάρη ω_n και αθροίζονται ως εξής:

$$S_0=0$$
$$S_{n+1}=\max(0,S_n + x_n - \omega_n)$$

Όταν η τιμή του S ξεπεράσει ένα κατώφλι h το οποίο έχει τεθεί από τον χρήστη, εντοπίζεται αλλαγή. Σε περίπτωση που θέλουμε να εξετάσουμε και για αρνητικές τιμές, θα πρέπει να υπολογιστεί και ένα δεύτερο άθροισμα[3]:

$$S^2_{n+1}=\min(0,S_n - x_n + \omega_n)$$

Όπου το κατώφλι εδώ θα έχει αρνητική τιμή. Τα βάρη ω_n αναθέτονται από τον χρήστη και αποτελούν την κανονική κατάσταση την οποία εξετάζουμε.

2.2.2. Προσαρμοστικός Cumulative Sum

Ο κλασικός αλγόριθμος Cusum έχει το μειονέκτημα ότι δεν μπορεί να προσαρμοστεί σε αλλαγές. Ένα δείγμα μπορεί πολυ εύκολα να έχει καταστάσεις(συστάδες) με διαφορετική διακύμανση. Αυτό σημαίνει ότι χρειάζεται διαφορετικό βάρος ω για τον έλεγχο της κάθε κατάστασης, καθώς εαν το βάρος είναι κατάλληλο για την συστάδα με την μικρότερη διακύμανση, τότε θα παρουσιάζεται αρνητικός συναγερμός, ενώ αν είναι κατάλληλο για την άλλη συστάδα τότε παρουσιάζει αποτυχία στον εντοπισμό κάποιων αλλαγών.

Για αυτό τον λόγο υλοποιήθηκε αλγόριθμος με μεταβλητό ω_n. Για την υλοποίησή του χρησιμοποιήθηκε η μέθοδος του σειρόμενου παραθύρου (Sliding window). Δεν δίνεται από τον χρήστη η αρχική κατάσταση, αλλά, μέσω μιας αρχικής φάσης εκπαίδευσης, ο αλγόριθμος υπολογίζει το ω_n ως την μέση τιμή των τελευτέων ν στοιχείων, με ν σταθερός αριθμός που δίνεται στην αρχή.

Στην συνέχεια το συσωρευτικό άθροισμα υπολογίζεται με τον ίδιο ακριβώς τρόπο και με κάθε νέο στοιχείο, το ω υπολογίζεται εκ νέου. Με αυτόν τον τρόπο, εάν παρατήρηθει συνεχής απόκλιση από μια «σωστή» κατάσταση, ο αλγόριθμος θα αρχίσει να προσαρμόζεται σε αυτή τη νέα κατασταση, θεωρώντας πλέον αυτή ως σωστή.

Ωστόσω για τις ανάγκες μερικών συνόλων δεδομένων μπορεί να χρειαστεί να επηρεάσουμε την αυστηρότητα του αλγορίθμου. Αυτό μπορούμε να το κάνουμε, εκτός με το να μεταβάλουμε το κατώφλι h, αυξάνοντας ή μειώνοντας το βάρος ω κάθε φορά που υπολογίζεται εκ νέου.

2.2.3. Shewhart Controller

Ένας ακόμη αρκετά γνωστός αλγόριθμος ανίχνευσης αλλαγών είναι ο Shewhart Controller, του οποίου όμως η λειτουργία είναι εντελώς διαφορετική από τον CuSum. Ο CuSum ανιχνεύει απότομες αλλαγές(βήματα) λαμβάνοντας υπόψη μόνο τα στοιχεία που εμφανίστηκαν μετά το τελευταίο βήμα, δηλαδή ελέγχει αν υπάρχει απόκλιση από την τωρινή κατάσταση (συσταδα). Αντίθετα ο Shewhart Controller λαμβάνει υπόψη ολόκληρο το σύνολο δεδομένων, από το πρώτο στοιχείο του μέχρι το τελευτέο.

Στον Shewhart Controller η κανονικότητα ενός στοιχείου X_n καθορίζεται από δύο όρια: το Άνω Όριο Ελέγχου (Upper Control Limit/UCL) και το Κάτω Όριο Ελέγχου (Lower Control Limit/LCL). Αυτά τα όρια ελέγχου υπολογίζονται ως εξής:

- $UCL = \bar{\mathbf{x}}_n + a*\sigma_n$
- $LCL = \bar{\mathbf{x}}_n - a*\sigma_n$

Όπου $\bar{\mathbf{x}}_n$ η μέση τιμή του συνόλου δεδομένων, σ_n η τυπική απόκλιση και a μία σταθερά που καθορίζεται από τον χρήστη (συνήθως 2 ή 3).

Κάθε φορά που εμφανίζεται καινούργιο στοιχείο x_n, ο αλγόριθμος ελέγχει αν ξεπερνάει ένα από τα δύο όρια. Αν $x_n > UCL_n$ τότε ο αλγόριθμος επιστρέφει 1, αν $x_n < UCL_n$ επιστρέφει -1, αλλιώς επιστρέφει 0, το οποίο σημαίνει ότι δεν υπάρχει αλλαγή.

Τέλος υπολογίζεται η καινούργια μέση τιμή και διακύμανση:

- $\overline{\mathbf{X}}_{\mathbf{n}} = \overline{\mathbf{X}}_{\mathbf{n-1}} + \frac{\mathbf{X_n} + \overline{\mathbf{X}}_{\mathbf{n-1}}}{\mathbf{n}}$
- $\sigma^2 = \frac{(\mathbf{n-1}) * \boldsymbol{\sigma}^{\mathbf{2}}_{\mathbf{n-1}} + (\mathbf{X_n} - \overline{\mathbf{x}}_{\mathbf{n}}) * (\mathbf{X_n} - \overline{\mathbf{X}}_{\mathbf{n-1}})}{\mathbf{n}}$

3. ΑΛΓΟΡΙΘΜΟΙ ΣΥΣΤΑΔΟΠΟΙΗΣΗΣ

Οι αλγόριθμοι συσταδοποίησης είναι οι μέθοδοι που ακολουθούνται ώστε να επιτυγχάνεται η συσταδοποίηση ενός συνόλου δεδομένων. Ανάλογα με το σύνολο δεδομένων κάποιοι αλγόριθμοι συσταδοποίησης λειτουργούν καλύτερα από κάποιους άλλους. Έτσι οι αλγόριθμοι αυτοί χωρίζονται σε κατηγορίες ανάλογα με την μεθοδολογία που ακολουθείται. Επιπλέον μπορούμε ακόμα να τους διαχωρίσουμε και σε στατικούς (offline) και online όπου η κύρια διαφορά τους είναι ότι οι στατικοί λειτουργούν όταν το σύνολο των δεδομένων είναι γνωστό εκ των προτέρων ενώ οι online μπορούν να δουλεύουν και χωρίς αυτό, αλλά εκτελώντας συσταδοποίηση στοιχείο-στοιχείο (π.χ., ροές δεδομένων).

Αλγόριθμοι ομαδοποίησης μπορούν να κατηγοριοποιηθούν ανάλογα με το μοντέλο διασποράς τους. Η ακόλουθη επισκόπηση θα απαριθμήσει μόνο τα πιο εξέχοντα παραδείγματα αλγορίθμων ομαδοποίησης, καθώς υπάρχουν πιθανώς πάνω από 100 αλγόριθμοι ομαδοποίησης που έχουν δημοσιευθεί. Δεν παρέχουν όλοι τα μοντέλα τους για τις ομάδες τους και επομένως δεν μπορούν εύκολα να κατηγοριοποιηθούν.

Δεν υπάρχει αντικειμενικά "σωστός" αλγόριθμος, αλλά όπως έχει ειπωθεί, «ομαδοποίηση είναι στο μάτι του θεατή.»[4] Ο καταλληλότερος αλγόριθμος για ένα συγκεκριμένο πρόβλημα πρέπει συχνά να επιλεγεί πειραματικά, εκτός αν υπάρχει μαθηματικός λόγος να προτιμηθεί ένα μοντέλο διασποράς πάνω από ένα άλλο. Θα πρέπει να σημειωθεί ότι ένας αλγόριθμος που έχει σχεδιαστεί για ένα είδος μοντέλου δεν έχει καμία πιθανότητα σε ένα σύνολο στοιχείων που περιέχει ένα ριζικά διαφορετικό είδος μοντέλου[4]. Για παράδειγμα, τα k-means δεν μπορούν να βρουν μη-κυρτές συστάδες[4].

Στη συνέχεια παρουσιάζονται οι στατικοί και οι online αλγόριθμοι συσταδοποίησης, οι κατηγορίες τους, καθώς και κάποιοι γνωστοί αλγόριθμοι για την κάθε κατηγορία.

3.1. Στατικοί αλγόριθμοι συσταδοποίησης

Οι στατικοί αλγόριθμοι συσταδοποίησης χωρίζονται σε αρκετές κατηγορίες οι οποίες είναι οι εξής:

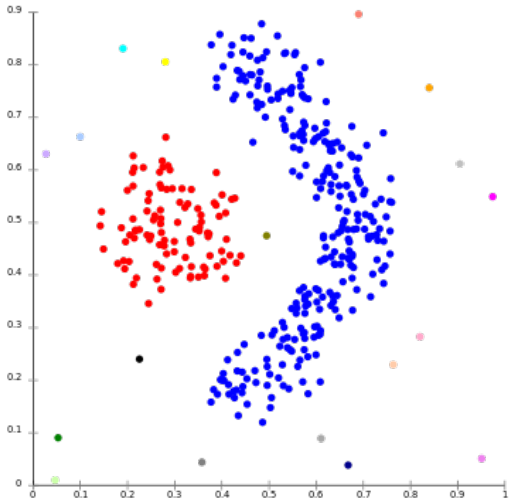
3.1.1. Συσταδοποίηση με βάση την συνδεσιμότητα

Συνδεσιμότητα με βάση την ομαδοποίηση, επίσης γνωστή ως ιεραρχική ομαδοποίηση , βασίζεται στην κεντρική ιδέα ότι τα αντικειμένα είναι πιο σχετικά με κοντινά αντικείμενα παρά με απομακρυσμένα. Αυτοί οι αλγόριθμοι συνδέουν "αντικείμενα" για να σχηματίσουν "συστάδες" με βάση την απόστασή τους.

Συνδεσιμότητα με βάση την ομαδοποίηση είναι μια ολόκληρη οικογένεια των μεθόδων που διαφέρουν από τον τρόπο που υπολογίζονται οι αποστάσεις. Εκτός από τη συνήθη επιλογή των λειτουργιών αποστάσεων, ο χρήστης πρέπει επίσης να αποφασίσει σχετικά με το κριτήριο σύνδεσης που θα χρησιμοποιήσει (ένα σύμπλεγμα αποτελείται από πολλά αντικείμενα και για αυτό υπάρχουν πολλοί υποψήφιοι για να υπολογίστει η απόσταση). Δημοφιλείς επιλογές είναι γνωστές ως single-linkage ομαδοποίηση (το ελάχιστο των αποστάσεων των αντικειμένων), complete linkage ομαδοποίηση (το μέγιστο των αποστάσεων των αντικειμένων) ή UPGMA, επίσης γνωστή και ως average linkage ομαδοποίηση(ο μέσος όρος των αποστάσεων).

Αυτές οι μέθοδοι δεν παράγουν μια μοναδική κατάτμηση του συνόλου των δεδομένων, αλλά μια ιεραρχία από την οποία ο χρήστης εξακολουθεί να πρέπει να επιλέξει τις κατάλληλες ομάδες. Δεν είναι πολύ ισχυρές έναντι των ακραίων τιμών, η οποίες είτε θα εμφανίζονται ως πρόσθετοι πόλοι ή ακόμη και θα προκαλούν την συγχώνευση άλλων συστάδων (γνωστό ως "chaining phenomenon", ιδίως με single-linkage ομαδοποίηση). Σε γενικές περιπτώσεις η πολυπλοκότητα είναι $\mathbf{O(n^3)}$, πράγμα που καθιστά τις μεθόδους αυτές πολύ αργές για μεγάλα σύνολα δεδομένων. Για ορισμένες ειδικές περιπτώσεις, υπάρχουν κάποιες βέλτιστες μεθόδοι της πολυπλοκότητας $\mathbf{O(n^2)}$: SLINK[5] για single-linkage και CLINK[6] για complete linkage ομαδοποίηση. Στην κοινότητα εξόρυξης δεδομένων οι μέθοδοι αυτές είναι αναγνωρισμένες ως θεωρητικά θεμέλια της συσταδοποίησης, αλλά συχνά

θεωρούνται ξεπερασμένές. Αποτέλεσαν όμως πηγή έμπνευσης για πολλές μεταγενέστερες μεθόδους όπως η συσταδοποίηση με βάση την πυκνότητα.



Εικόνα 1: Single-linkage σε συστάδες βασισμένες στην πυκνότητα. Εξάγονται 20 συστάδες, οι περισσότερες από τις οποίες περιέχουν μόνο ένα στοιχείο, δεδομένου ότι αυτή η ομαδοποίηση δεν έχει την έννοια του «θορύβου».

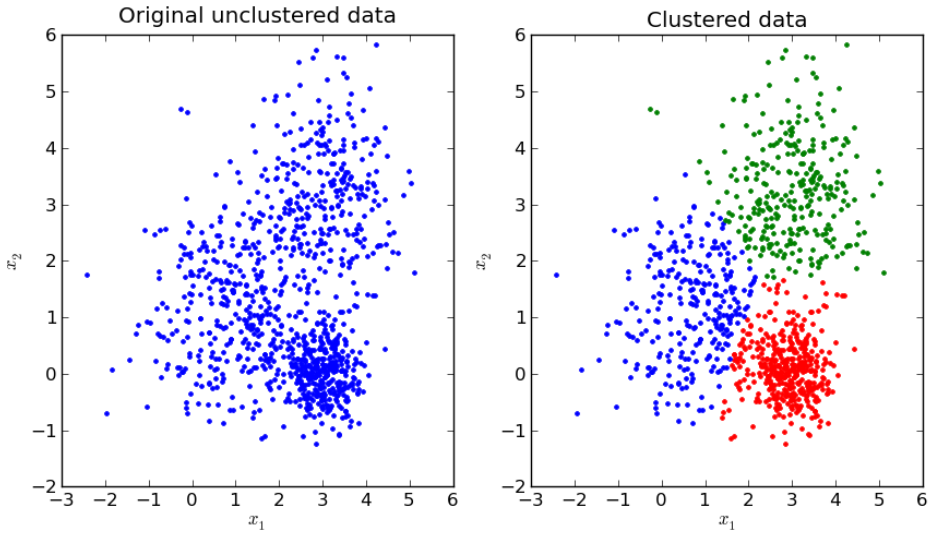
3.1.2. Συσταδοποίηση με βάση το κεντροειδές

Στην συσταδοποίησης με βάση το κεντροειδές, οι συστάδες εκπροσωπούνται από ένα κεντρικό σημείο, το οποίο όμως μπορεί να μην είναι μέλος του συνόλου δεδομένων. Όταν ο αριθμός των συστάδων είναι κάποιος σταθερός αριθμός k, η ομαδοποίηση k-means δίνει ένα επίσημο ορισμό ως ένα πρόβλημα βελτιστοποίησης: βρείτε τα k κέντρα των συστάδων και εκχωρήστε τα αντικείμενα στο πλησιέστερο κέντρο του συστάδων, έτσι ώστε να ελαχιστοποιούνται τα τετράγωνα των αποστάσεων από το σύμπλεγμα.

Το πρόβλημα βελτιστοποίησης ίδια είναι γνωστό ότι είναι NP-hard , και ως εκ τούτου η κοινή προσέγγιση είναι μόνο προσεγγιστικές λύσεις. Μια ιδιαίτερα γνωστή κατά προσέγγιση μέθοδος είναι αλγόριθμος του Lloyd[7], που συχνά αναφέρεται ως " k-means αλγόριθμος ". Παρόλα αυτά μπορεί να βρει μόνο ένα τοπικό βέλτιστο , και συνήθως εκτελείται πολλές φορές με διαφορετικές τυχαίες αρχικοποιήσεις. Παραλλαγές του k-means περιλαμβάνουν συχνά τέτοιες βελτιστοποιήσεις όπως την επιλογή των καλύτερων από πολλαπλές εκτελέσεις, αλλά και τον περιορισμό των κεντροειδών σε μέλη του συνόλου

δεδομένων (k-medoids), επιλέγοντας τα αρχικά κέντρα λιγότερο τυχαία (k-means ++).

Οι περισσότεροι αλγόριθμοι τύπου k-means απαιτούν τον αριθμό των συστάδων k να προσδιοριστεί εκ των προτέρων, το οποίο θεωρείται ότι είναι ένα από τα μεγαλύτερα μειονεκτήματα αυτών των αλγορίθμων. Επιπλέον, οι αλγόριθμοι αυτοί προτιμούν συστάδες παρόμοιου μεγέθους, καθώς θα εισάγουν πάντα αντικείμενα στο κοντινότερο κεντροειδές. Αυτό συχνά οδηγεί σε εσφαλμένο προσδιορισμό των συνόρων μεταξύ των συστάδων (το οποίο δεν αποτελεί έκπληξη, καθώς οι αλγόριθμο βελτιστοποιούν τα κέντρα των συστάδων, και όχι τα σύνορα τους).



Εικόνα 2:K-means συσταδοποίηση

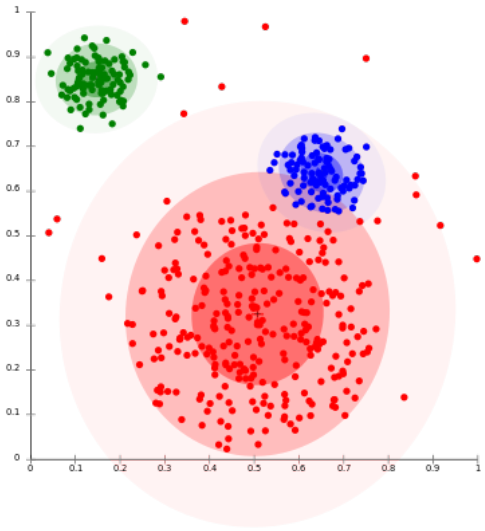
3.1.3. Συσταδοποίηση με βάση την κατανομή των δεδομένων

Το μοντέλο ομαδοποίησης με πιο στενή σχέση με την στατιστική βασίζεται σε μοντέλα κατανομής . Συστάδες μπορούν στη συνέχεια εύκολα να οριστούν ως αντικείμενα που πιθανότατα ανήκουν στην ίδια κατανομή. Μια βολική ιδιότητα αυτής της προσέγγισης είναι ότι αυτό μοιάζει με τον τρόπο με τον οποίο παράγονται τεχνητά σύνολα δεδομένων: με τυχαία δειγματοληψία αντικειμένων από μια κατανομή.

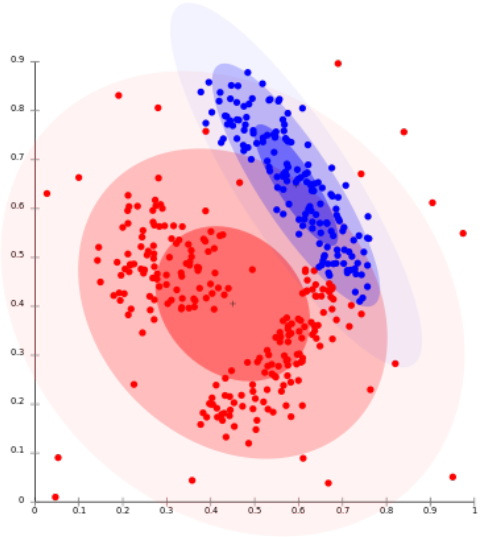
Ενώ η θεωρητική θεμελίωση των μεθόδων αυτών είναι εξαιρετική, υποφέρουν από ένα βασικό πρόβλημα που είναι γνωστό
ως υπερπροσαρμογή (overfitting), όπου το μοντέλο που προέκυψε συνήθως περιγράφει θόρυβο και όχι την βασική σχέση των δεδομένων, οπότε δεν θα

έχει καλή πρόβλεψη αφού συνήθως “υπερβάλει” για μικρές διακυμάνσεις των δεδομένων.

Μια εξέχουσα μέθοδος είναι γνωστή ως expectation-maximization algorithm [8] (ή αλλιώς EM-Clustering). Εδώ, το σύνολο δεδομένων συνήθως μοντελοποιείται με ένα σταθερό (για την αποφυγή υπερπροσαρμογής) αριθμό από Gaussian κατανομές που αρχικοποιούνται τυχαία και των οποίων οι παράμετροι έχουν βελτιστοποιηθεί για να ανταποκρίνονται καλύτερα στο σύνολο δεδομένων. Αυτό συγκλίνει σε ένα τοπικό βέλτιστο , έτσι πολλαπλές εκτελέσεις μπορούν να παράγουν διαφορετικά αποτελέσματα.



Εικόνα 3:Σε δεδομένα με κατανομή Gauss, ο EM λειτουργεί καλά, δεδομένου ότι χρησιμοποιεί Gaussians μοντέλα ομαδοποίησης



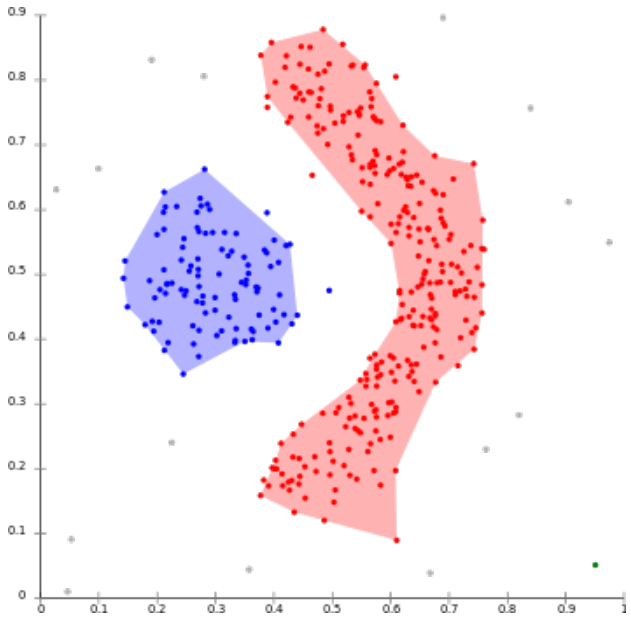
Εικόνα 4:Συστάδες βασισμένες στην διασπορά δεν μπορούν να μοντελοποιηθούν χρησιμοποιώντας Gaussian κατανομές

3.1.4. Συσταδοποίηση με βάση την διασπορά

Στην συσταδοποίηση με βάση την διασπορά[9] ,ως συστάδες ορίζονται οι περιοχές με υψηλότερη πυκνότητα απ' ότι το υπόλοιπο της ομάδας δεδομένων. Τα αντικείμενα σε αυτές τις αραιές περιοχές - που απαιτούνται για τον διαχωρισμό των συστάδων - συνήθως θεωρούνται ότι είναι θόρυβος και συνοριακά σημεία.

Η πιο δημοφιλής[10] μέθοδος της συσταδοποίηση με βάση την διασπορά είναι ο DBSCAN[11]. Σε αντίθεση με πολλές νεότερες μεθόδους, διαθέτει ένα καλά καθορισμένο πρότυπο ομαδοποίησης που ονομάζεται «density-reachability». Παρόμοια με την ομαδοποίηση με βάση την συνδεσιμότητα, βασίζεται σε σημεία σύνδεσης εντός ορισμένων ορίων αποστάσεων. Ωστόσο, συνδέει μόνο σημεία που ικανοποιούν ένα κριτήριο διασποράς, το οποίο στην αρχική παραλλαγή ορίζεται ως ένας ελάχιστος αριθμός άλλων αντικειμένων στο εσωτερικό αυτής της ακτίνας. Ένα σύμπλεγμα αποτελείται από όλα αντικείμενα τα οποία ικανοποιούν τα παραπάνω κριτήρια (τα οποία μπορούν να σχηματίσουν σύμπλεγμα αυθαίρετου σχήματος, σε αντίθεση με πολλές άλλες μεθόδους) συν όλα τα αντικείμενα που βρίσκονται εντός εμβέλειας από τα αντικείμενα αυτά. Μια άλλη ενδιαφέρουσα ιδιότητα του DBSCAN είναι ότι η πολυπλοκότητά του είναι αρκετά χαμηλή και ότι θα έχει ουσιαστικά τα ίδια αποτελέσματα σε κάθε εκτέλεση, και ως εκ τούτου δεν υπάρχει καμία ανάγκη να τρέξει πολλές φορές.

Ο OPTICS[12] είναι μια γενίκευση του DBSCAN που καταργεί την ανάγκη για επιλογής μιας κατάλληλης τιμής για την παράμετρο της ακτίνας Eps, και παράγει ένα ιεραρχικό αποτέλεσμα.Επίσης υπάρχει και ο DeLi-Clu [13] (Density-Link-Clustering) όπου συνδυάζει ιδέες από το single-linkage clustering και τον OPTICS εξαφανίζοντας τελείως την παράμετρο Eps και προσφέροντας βελτιώσεις επιδόσεων σε σύγκριση με τον OPTICS χρησιμοποιώντας ένα R-Tree ευρετήριο



Εικόνα 5:Ομαδοποίηση βασισμένη στην διασπορά με DBSCAN

3.2. On-line Αλγόριθμοι Συσταδοποίησης

Οι online αλγόριθμοι έχουν σαν στόχο τη συνεχή συσταδοποίηση δεδομένων και η διαφορά τους με τους στατικούς αλγόριθμους είνα ότι δεν ξέρουμε το σύνολο των δεδομένων εκ των προτέρων καθώς δέχονται δεδομένα σε πραγματικό χρόνο. Έτσι η συσταδοποίηση αλλάζει δυναμικά. Χωρίζονται σε “καθαρά” online αλλά και σε batch online τρόπους όπου η κύρια διαφορά τους είναι ότι οι online δουλεύουν στοιχείο-στοιχείο ενώ οι batch online συνήθως έχουν μία μνήμη όπου αποθηκεύουν τα στοιχεία που έρχονται και κάθε φορά που γεμίζει αυτή η μνήμη αδειάζουν τα στοιχεία της και δουλεύουν πάνω σε αυτά σαν στατικοί αλγόριθμοι.

Υπάρχουν αρκετές παραλλαγές γνωστών στατικών αλγορίθμων που τους επιτρέπουν να δουλέυουν online όπως είναι ο online k-means [14] όπου στην ουσία η ενημέρωση των κεντροειδών δεν γίνεται κάθε φορά στο τέλος του συνόλου δεδομένων αλλά ενημερώνεται το κεντροειδές κάθε φορά που ένα καινούργιο στοιχείο τοποθετηθεί στην συστάδα του.

Άλλος ένας online αλγόριθμος είναι ο ART [15] ο οποίος παίρνει σαν όρισμα μία παράμετρο που ονομάζεται vigilance και δημιουργεί συστάδες καθώς έρχονται τα στοιχεία με βάση αυτή. Έτσι όταν έρχεται ένα στοιχείο, καταχωρείται στην κοντινότερη συστάδα με βάση κάποια μετρική απόστασης εφόσον ικανοποιεί το κριτήριο της vigilance παραμέτρου, δηλαδή η απόσταση του από το κέντρο της συστάδας να είναι μικρότερη από αυτή. Ειδάλως, το στοιχείο αυτό δημιουργεί καινούργια συστάδα ακτίνας μήκους ίσης με το vigilance και κέντρο της συστάδας γίνεται το σημείο αυτό (novelty).

3.3. Μετρικές αποστάσεων

Όπως προαναφέρθηκε στο κεφάλαιο 1 (1.1.4.), στόχος της συσταδοποίησης είναι η ομαδοποίηση των αντικειμένων με βάση κάποιο μέτρο ομοιότητας. Ομοιότητα όμως είναι μια πολύ γενική έννοια, άρα πρώτα θα πρέπει να οριστεί τι είναι ομοιότητα και τι μη ομοιότητα.

Ως ομοιότητα ορίζεται μια αριθμητική μέτρηση για το πόσο όμοια είναι δυο αντικείμενα ενώ ως μη ομοιότητα ορίζεται μια αριθμητική μέτρηση για το πόσο διαφορετικά είναι δυο αντικείμενα.

Η ομοιότητα ή μη ομοιότητα μεταξύ δυο αντικειμένων υπολογίζεται συνήθως σύμφωνα με κάποια συνάρτηση απόστασης ανάμεσα στα δύο αντικείμενα. Ακολουθούν κάποιες από τις πιο σημαντικές μετρικές αποστάσεων [16].

3.3.1. Ευκλείδεια Απόσταση

Η πιο απλή και γνωστή μετρική απόστασης είναι φυσικά η ευκλείδια απόσταση,δηλαδή η γεωμετρική απόσταση στον πολυδιάστατο χώρο [12].

Έστω δύο διανύσματα x, y του n-διάστατου χώρουR^n με x = {x_1,….,x_n} και y = {y_1,….,y_n}. Τότε η ευκλείδεια απόσταση υπολογίζεται ως [17]:

$$\mathrm{dist}(x, y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}$$

Όπως παρατηρούμε απο τον παραπάνω τύπο, για να υπολογιστεί η ευκλείδεια απόσταση χρησιμοποιούνται τετραγωνικές αποκλίσεις. Αυτό έχει ως άμεση συνέπεια τα outliers να έχουν μεγάλη επίδραση στον υπολογισμό της απόστασης.

3.3.2. Απόσταση Manhattan

Η απόσταση Manhattan μοιάζει πολύ με την ευκλείδεια απόσταση με τη διαφορά ότι δεν χρησιμοποιεί τετραγωνικές αποκλίσεις αλλά απόλυτες αποκλίσεις. Συνήθως λόγω της ομοιότητας της με την ευκλείδεια απόσταση δίνει περίπου τα ίδια αποτελέσματα, εκτός από τη περίπτωση που υπάρχουν outliers [18].

Έστω τα διανύσματα x, y του n-διάστατου χώρου R^n με x = {x_1,….,x_n} και y = {y_1,….,y_n}. Τότε η απόσταση Manhattan υπολογίζεται ως :

$$\mathrm{dist}(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

3.3.3. Απόσταση Minkowsky

Η απόσταση Minkowsky γενικεύει την ευκλείδεια απόσταση και την απόσταση Manhattan. Ορίζεται ως [18]:

$$\mathrm{dist}\,(\mathrm{x},\mathrm{y}) = \sum_{\mathrm{i}=1}^{\mathrm{n}} [(|\mathrm{x_i} - \mathrm{y_i}|)^{\mathrm{q}}]^{1/\mathrm{q}}$$

Η τιμή της παραμέτρου q μπορεί να χρησιμοποιηθεί για να δώσει ιδιαίτερο βάρος σε κάποιες αποκλίσεις. Για q=2 προκύπτει η ευκλείδεια και για q=1 η Manhattan.

3.3.4. Απόσταση Mahalanobis

Η απόσταση Mahalanobis για δυο αντικείμενα $x = \{x_1,\ldots,x_n\}$ και $y = \{y_1,\ldots,y_n\}$ ορίζεται ως :

$$\mathrm{dist}(\mathrm{x},\mathrm{y}) = \sqrt{(\mathrm{x}-\mathrm{y})\mathrm{C}^{-1}(\mathrm{x}-\mathrm{y})^{\mathrm{T}}} \qquad (3.15)$$

όπου C είναι ο αντίστοιχος covariance matrix. Η δεδομένη μετρική απόστασης υπερέχει έναντι της ευκλείδειας διότι λαμβάνει υπόψη της, της εξής δύο παραμέτρους:

- τη μεταβλητότητα (variance) κάθε μεταβλητής
- τις συσχετίσεις (correlation) που πιθανόν να υπάρχουν ανάμεσα στις μεταβλητές

4. ΜΕΘΟΔΟΛΟΓΙΕΣ – ΜΕΤΡΙΚΕΣ ΣΥΣΤΑΔΟΠΟΙΗΣΗΣ

Ένα από τα σημαντικότερα ζητήματα τα οποία μας απασχολούν κατά τη διαδικασία της συσταδοποίησης είναι το κατά πόσο μπορούμε τελικά να επιτύχουμε όσο το δυνατόν καλύτερη ποιότητα συσταδοποίησης.

Για αυτό το λόγο έχουν αναπτυχθεί διάφορες μετρικές όπως κριτήρια ομοιότητας, μέθοδοι εντοπισμού των outliers αλλά και τεχνικές προσδιορισμού του βέλτιστου αριθμού συστάδων που μπορούν να προκύψουν από ένα σύνολο δεδομένων. Ο πιο διαδεδομένος τρόπος συσταδοποίησης είναι η ομαδοποίηση αντικειμένων σε συστάδες σύμφωνα με κάποια μετρική απόστασης.

Ωστόσο υπάρχουν περιπτώσεις, ειδικότερα όταν πρόκειται για on-line συσταδοποίηση, που αυτό δεν αρκεί και απαιτείται συγχώνευση συστάδων ή διαχωρισμός συστάδας. Όταν συμβαίνει αυτό δημιουργείται η αναγκαιότητα χρήσης τεχνικών καθ' όλη τη διάρκεια της συσταδοποίησης, οι οποίες ανάλογα με το αποτέλεσμα που θα επιφέρουν θα οδηγήσουν σε μια από τις εξής ενέργειες: συγχώνευση, διάσπαση, καμία ενέργεια.

Ιδιαίτερα σημαντικό είναι επίσης το γεγονός ότι η συσταδοποίηση εξαρτάται σε σημαντικό βαθμό από το πως είναι κατανεμημένα τα δεδομένα και από το είδος των συστάδων που προκύπτουν. Συνεπώς πρέπει να δίνεται ιδιαίτερη προσοχή στο ποια μέθοδος θα επιλεχθεί. Επιπλέον σε ορισμένες περιπτώσεις, είναι πιθανό να συνδυαστούν παραπάνω από μία μετρικές ώστε να επιτευχθεί μια αρκετά καλή ποιότητας συσταδοποίησης.

Στο κεφάλαιο αυτό παρουσιάζονται αναλυτικά οι μετρικές που χρησιμοποιήθηκαν στην εργασία αυτή καθώς και τα βήματα για την υλοποίηση τους.

4.1. Cumulative Sum πολλαπλων καταστάσεων

Σε μια πρώτη προσπάθεια συσταδοποίησης, τροποποιήθηκε ο αλγόριθμος ανίχνευσης αλλαγών Cusum έτσι ώστε να λειτουργεί με πολλαπλές καταστάσεις.

Ο αλγόριθμος αυτός λειτουργεί χώρις αρχική κατάσταση από τον χρήστη. Στην αρχή δεν υπάρχει καμιά αποθηκευμένη κατάσταση. Όταν λοιπόν δεχτεί το πρώτο στοιχείο του δείγματος, δημιουργεί μία κατάσταση με βάρος ω_n το ίδιο το στοιχείο. Στην συνέχεια λειτουργεί όπως ο κανονικός Cusum:

$S_0=0$
$S_{n+1}=\max(0,S_n + x_n - \omega_n)$
$S^2_{n+1}=\min(0,S_n - x_n + \omega_n)$

Ώσπου να βρει αλλαγή, δηλαδή $S_{n+1}>$threshold. Σε αυτό το σημείο δημιουργεί μια καινούρια συστάδα με βάρος ω_n το στοιχείο που προκάλεσε την αλλαγή, με καινούργιο S_{n+1}, ξεχωριστό από το προηγούμενο. Από εκεί και έπειτα γίνεται έλεγχος για ανίχνευση αλλαγής σε κάθε συστάδα, και αν παρατηρηθεί αλλαγή για όλες τις συστάδες, δημιουργείται με τον ίδιο τρόπο καινούργια συστάδα. Αντίθετα αν σε μια ή περισσότερες συσταδες δεν παρουσιαστεί αλλαγή, το στοιχείο εντάσεται σε αυτήν με το μικρότερο άθροισμα $S_{n+1}+ S^2_{n+1}$, δηλαδή στην συστάδα με την μικρότερη απόκλιση από αυτό.

4.2. Silhouette Coefficient

Η συνοχή και ο διαχωρισμός είναι οι δύο αρχές πάνω στις οποίες στηρίζεται το silhouette coefficient. Ο όρος συνοχή αναφέρεται στο πόσο στενά συνδεδεμένα είναι τα στοιχεία σε μία συστάδα, ενώ ο όρος διαχωρισμός αναφέρεται στο εάν μια συστάδα είναι επαρκώς διαχωρίσιμη από τις υπόλοιπες συστάδες[4].

Για κάθε σημείο n, το silhouette Sil(n) δίνεται από το τύπο :

$$Sil(n) = \frac{b(n) - a(n)}{\max(a(n), b(n))} \qquad (3.1)$$

όπου b(n) είναι η μέση απόσταση του n από όλα τα σημεία κάθε άλλης συστάδας και α(n) είναι η μέση απόσταση από τα σημεία της συγκεκριμένης συστάδας. Από όλα τα b(n) που υπολογίζονται επιλέγεται αυτό με την ελάχιστη τιμή.

Με λίγα λόγια το b(n) είναι η ομοιομορφία (similarity) του n με τα στοιχεία των υπολοίπων συστάδων, ενώ το α(n) είναι η ανομοιομορφία (dissimilarity) με τα στοιχεία της υποψήφιας συστάδας.

Το Sil(n) παίρνει τιμές στο διάστημα [-1, 1]. Όσο πιο πολύ τείνει το αποτέλεσμα στο 1, τόσο καλύτερη θεωρείται η συσταδοποίηση. Το ακριβώς αντίθετο ισχύει εάν το αποτέλεσμα τείνει στο -1. Αποτελεί δηλαδή την «ορθότητα» της συσταδοποίησης.

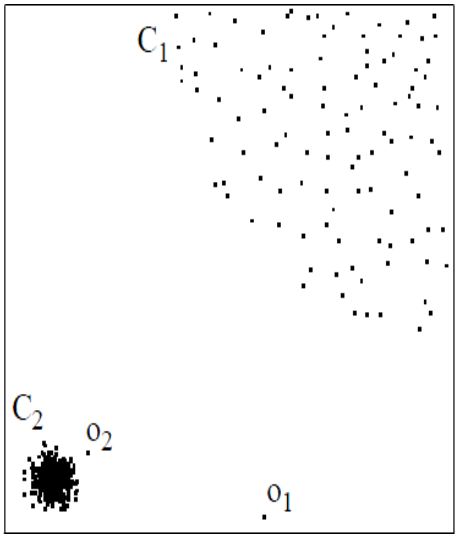
Συνεπώς η μέση τιμή των silhouette AvgSil(k), με k να δηλώνεται ο αριθμός των συστάδων, ορίζεται ως :

$$AvgSil(k) = \frac{1}{N}\sum_{1}^{N} Sil(n) \qquad (3.2)$$

Από τα k που δοκιμάζονται επιλέγεται τελικά εκείνο που μεγιστοποιεί το AvgSil(k).

4.3. Local Outlier Factor

Ένα ακόμα σημαντικό θέμα το οποίο μας απασχολεί κατά τη συσταδοποίηση είναι η ανίχνευση ακραίων τιμών (outlier detection).Ο όρος ακραίες τιμές, αναφέρεται σε τιμές των οποίων η συμπεριφορά αποκλίνει σε σχέση με τις υπόλοιπες. Συνήθως, στο πεδίο των τεχνικών εξόρυξης δεδομένων, οι ακραίες τιμές δηλώνουν τη παρουσία θορύβου. Στην Εικόνα 4.1 για παράδειγμα τα σημεία O_1 και O_2 θεωρούνται ως outliers για τις συστάδες.



Εικόνα 6:LOF

Local Outlier Factor, ή με συντομία LOF, πρόκειται για μια μέθοδο, η οποία εφαρμόζεται για εντοπισμό ακραίων σημείων. Βασικό στοιχείο του Local Outlier Factor είναι η εισαγωγή της έννοιας του παράγοντα έκτοπου (outlier factor). Ο Local Outlier Factor προσπαθεί να αποφασίσει εάν ένα σημείο είναι ακραίο λαμβάνοντας υπόψη την πυκνότητα της ευρύτερης γειτονιάς[5].

Ειδικότερα για τον υπολογισμό του LOF απαιτούνται τα παρακάτω βήματα:

- Πρώτο βήμα είναι ο υπολογισμός του k-distance ενός στοιχείου X.

k-distance ονομάζεται η απόσταση ενός στοιχείου με το $k_{οστό}$ κοντινότερο γείτονά του Y, όπου k θετικός ακέραιος αριθμός.

- Στην συνέχεια υπολογίζεται το k-distance της γειτονιάς του X.

Γειτονιά του X αποτελούν τα k κοντινότερα στοιχεία του X. Για κάθε στοιχείο από αυτά λοιπόν υπολογίζεται το k-distance του και στην συνέχεια η μέση τιμή αυτών.

- Στο επόμενο βήμα προσδιορίζεται το reachability distance[6] του X και των γειτώνων του, το οποίο είναι το max[k-distance(Y) , d(X,Y)].

- Μετά υπολογίζεται το local reachability density του X με τον τύπο:

$$\mathrm{lrd}(\mathrm{X}) = \frac{|N(X)|}{\sum_{\mathrm{o} \in \mathrm{N}(\mathrm{X})} \mathrm{reachdist}_k(\mathrm{X}, \mathrm{Y})}$$

- Τέλος, το LOF ορίζεται ως:

$$LOF(X) = \frac{\sum_{\mathrm{o} \in \mathrm{N}(\mathrm{X})} \frac{\mathrm{lrd}_{\mathrm{MinPts}}(\mathrm{Y})}{\mathrm{lrd}_{\mathrm{MinPts}}(\mathrm{X})}}{\mathrm{lrd}(\mathrm{X})}$$

Με βάση την τιμή αυτή και το κατώφλι που έχουμε θέσει εξ'αρχής, αποφασίζεται εάν ένα σημείο θεωρείται ή όχι outlier.

4.4. Αλγοριθμος ελέγχου απόστασης συστάδων

Ο αλγόριθμος, τον αυτός ομαδοποιεί δεδομένα λαμβάνοντας υπ' όψη τις αποστάσεις μεταξύ όλων των συστάδων, για να αποφασίσει την επιτρεπόμενη απόκλιση ενός στοιχείου την συστάδα στην οποία ανοικει.

Τα βήματα αυτου του αλγορίθμου είναι τα εξής:

- Αρχικά υπολογίζει την μέση απόσταση meanD για όλες τις συστάδες μεταξύ τους.

Η απόσταση αυτή μπορεί να οριστεί με διαφόρους τρόπους. Σε αυτήν την περίπτωση ορίζω την απόσταση μεταξύ των συστάδων ως την απόσταση των μέσων ώρων των στοιχείων τους.

- Με την εμφάνιση καινούργιου στοιχείου, υπολογίζεται η απόσταση d από την κοντινότερη συστάδα. Αν αυτή η απόσταση d είναι μικρότερη από το meanD επί κάποιο ποσοστό/threshold P που έχουμε επιλέξει τότε εντάσεται σε αυτή τη συστάδα, αλλίως δημιουργεί καινούργια.

Το ποσοστό P αποτελεί την «αυστηρότητα» με την οποία θέλουμε να λειτουργήσει ο αλγόριθμος και μπορούμε να το μεταβάλουμε για τις ανάγκες του κάθε dataset.

- Τέλος πρέπει να υπολογίσουμε ξανά τις αποστάσεις μεταξύ των συστάδων καθώς θα έχουν αλλάξει κατά πάσα πιθανότητα με την εμφάνιση του νέου στοιχείου.

4.5. Λεπτομέρειες Υλοποίησης

Ως γλώσσα προγραμματισμού για την υλοποίηση των παραπάνω αλγορίθμων επιλέχθηκε η Java.

Πρόκειται για μια σύγχρονη αντικειμενοστραφή γλώσσα που έχει παρόμοια σύνταξη με τη C. Τα πλεονεκτήματα που προσφέρει η Java σε σύγκριση με άλλες γλώσσες προγραμματισμού, μας οδήγησαν στην επιλογή της για την υλοποίηση του συγκεκριμένου πλαισίου.

Αρχικά, η Java σαν μια καθαρά αντικειμενοστραφής γλώσσα, οργανώνει τον κώδικα σε αυτόνομες μονάδες που λέγονται κλάσεις οι οποίες χρησιμοποιούνται για να δομήσουν μεγαλύτερα «οικοδομήματα». Αυτό είναι πολύ μεγάλο πλεονέκτημα διότι υλοποιώντας μικρές-μικρές ψηφίδες, συνδέοντας τις και επαναχρησιμοποιώντας τις για να φτιάξεις κάτι μεγαλύτερο έχει ως αποτέλεσμα : κώδικα που είναι πιο εύκολα επεκτάσιμος και γρηγορότερη αποσφαλμάτωση. Σημαντικό είναι ακόμα το πόσο απλή και κομψή είναι σαν γλώσσα, με ένα καλοσχεδιασμένο API, δίνοντας έτσι τη δυνατότητα παραγωγής κώδικα με λιγότερα bugs και μειώνοντας το χρόνο ανάπτυξης μιας εφαρμογής. Επιπλέον, παρέχει τη δυνατότητα να τρέξουμε την εφαρμογή οπουδήποτε, αφού γράψουμε τον κώδικα, αρκεί να υποστηρίζεται εκεί η πλατφόρμα της Java, κάτι που συμβαίνει σε όλα τα κύρια λειτουργικά συστήματα σήμερα.

Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε το παρακάτω λογισμικό :

- JDK: Java SE Development Kit 7
- IDE: Netbeans 7.3
- OS: Windows 7, x64 και Ubuntu 11.04, x64
- Matlab 2012

Το λογισμικό Matlab χρησιμοποιήθηκε για την οπτικοποίηση των αποτελεσμάτων.

5. ΠΕΙΡΑΜΑΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ

Στο κεφάλαιο αυτό παρουσιάζονται η απόδοση και η λειτουργικότητα των αλγορίθμων που περιγράφηκαν στις ενότητες 2 και 3. Μερικοί αλγόριθμοι δοκιμάζονται πάνω από μια φορά με διαφορετικές αρχικές τιμές για να φανούν οι διαφορές στην λειτουργία τους.

Στην ενότητα 5.1 δίνεται μια περιγραφή των συνόλων δεδομένων που χρησιμοποιήθηκαν για την τεκμηρίωση της παρούσας μελέτης ενώ στην ενότητα 5.2 παρουσιάζονται τα κριτήρια που λήφθηκαν υπόψη προκειμένου να εξαχθούν συμπεράσματα για την απόδοση της προτεινόμενης μεθοδολογίας, συγκριτικοί πίνακες αποτελεσμάτων αλλά και χρόνοι απόκρισης.

5.1. Σύνολα δεδομένων που χρησιμοποιήθηκαν

Στην πειραματική μελέτη αυτή χρησιμοποιήθηκαν 9 Datasets των 21145 ακατέργαστων δεδομένων (raw data), τα οποία προέρχονται από μετρήσεις αισθητήρων πλοίων, με στόχο την ομαδοποίησή τους.
Τα ονόματα των Datasets είναι:
DISPANCESLR, INCLINOMETERYMAX, INCLINOMETERYZC, GROUNDSPEED, DEPTH, INCLINOMETERXZC, MEPOWER, METORQUE, INCLINOMETERXMAX

5.2. Πειράματα – Αποτελέσματα

Στην ενότητα αυτή παρουσιάζονται τα κριτήρια με βάση τα οποία πραγματοποιήθηκε η σύγκριση των αλγορίθμων. Επιπλέον παραθέτονται συγκριτικοί πίνακες αποτελεσμάτων, οπτικές απεικονίσεις αποτελεσμάτων συσταδοποίησης, αλλά και διαγράμματα χρόνων απόκρισης του συστήματος.

5.2.1. Μετρικές – Σενάρια που δοκιμάστηκαν

Στα πειράματα που πραγματοποιήσαμε, χρησιμοποιήσαμε εκτός από τους αλγορίθμους που περιγράψαμε στο κεφάλαιο 2 και 3, έναν συνδιασμό του Sillhoutte Coefficient και του Local Outlier Factor, καθώς ο LOF δεν μπορεί να χρησιμοποιηθεί μόνος του.

Προκειμένου να αξιολογήσουμε τα αποτελέσματα που προέκυψαν, βασιστήκαμε σε κάποιες μετρικές. Ειδικότερα, εκτελέσαμε τους αλγορίθμους με τα ίδια σύνολα δεδομένων και με αντίστοιχα ορίσματα εισόδου.

Έπειτα καταγράψαμε τα εξής:

- αριθμό συστάδων που προέκυψαν
- βέλτιστο αριθμό συστάδων
- απόκλιση από βέλτιστο αριθμό συστάδων
- χρόνος απόκρισης

5.2.2. Αποτελέσματα/Συγκριση Αλγορίθμων

5.2.2.1. Αλγόριθμοι ανίχνευσης αλλαγής

Ο παρακάτω πίνακες (5.1) και (5.2) αναφέρονται στην μετρική του Προσαρμοστικού CuSum και του Shewhart Controller αντίστοιχα και συγκρίνονται με τον κλασικό αλγόριθμο Cumulative Sum ως προς τον αριθμό αλλαγών που ανιχνεύτικαν. Ο προσαρμοστικός CuSum έχει εκτελεστεί με διάφορες αρχικές τιμές για βάρος ω, μέγεθος μνήμης στοιχείων v καθώς και για το κατώφλι h, ενώ ο Shewhart Controller έχει εκτελεστεί με διάφορες τιμές του a.

Πίνακας 1:Αποτελέσματα ανίχνευσης αλλαγών προσαρμοστικού CuSum

Μετρική / Σύνολα	CuSum	Προσαρμοστικός Cumulative Sum				
		ω=40, h=20, v=5	ω=20, h=20, v=5	ω=40, h=10, v=5	ω=40, h=20, v=10	ω=40, h=20, v=3
DISPANCESLR	11950	8253	8253	8253	13273	5382
INCLINOMETERYMAX	4579	7241	7242	7241	11188	4828
INCLINOMETERYZC	2717	4099	4126	4104	6735	2759
GROUNDSPEED	2120	2166	2190	2175	3785	1430
DEPTH	4119	4293	4790	4395	6237	3278
INCLINOMETERXZC	4250	5490	6012	5589	7926	4153
MEPOWER	7016	8318	9040	8498	11206	6610
METORQUE	5226	7524	8075	7644	10379	5911
INCLINOMETERXMAX	4258	4574	5394	4842	6366	3438

Πίνακας 2:Αποτελέσματα ανίχνευσης αλλαγών Shewhart Controller

Μετρική / Σύνολα	CuSum	Shewhart Controller		
		a = 2	a = 3	a = 4
DISPANCESLR	11950	4	2	2
INCLINOMETERYMAX	4579	1535	7	4
INCLINOMETERYZC	2717	2264	1241	2
GROUNDSPEED	2120	1718	1696	113
DEPTH	4119	534	333	247
INCLINOMETERXZC	4250	1367	1068	746
MEPOWER	7016	791	601	596
METORQUE	5226	1463	463	99
INCLINOMETERXMAX	4258	906	618	459

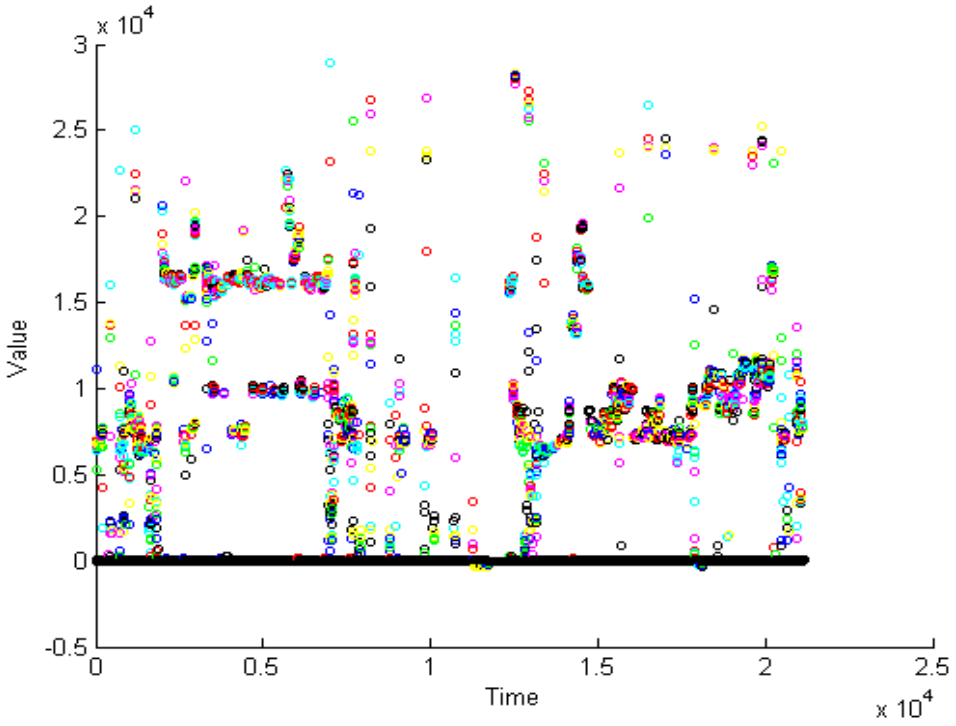
5.2.2.2. Αλγόριθμοι συσταδοποίησης

Στον Πίνακα 5.3 παρουσιάζονται γενικά τα αποτελέσματα των αλγορίθμων συσταδοποίησης για όλα τα σύνολα δεδομένων (ενότητα 5.1), καθώς και ο αναμενόμενος αριθμός συστάδων για κάθε σύνολο.

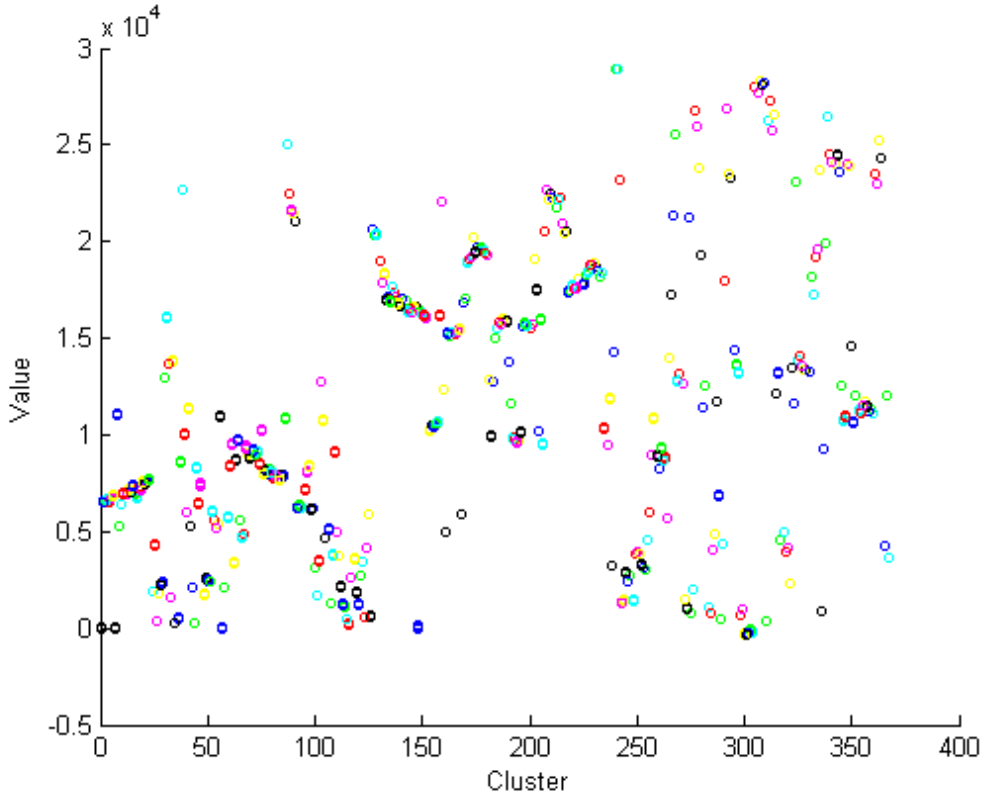
Πίνακας 3:Αποτελέσματα συσταδοποίησης

Μετρικές / Σύνολα	CuSum Πολλαπλών καταστάσεων	LoF	Sil. Coef.	Sil. Coef. +LoF	Αλγόριθμος ελέγχου απόστασης	Αλγόριθμος ελέγχου απόστασης (λιγότερο αυστηρός)	Αναμενόμενος αριθμός συστάδων
DISPANCESLR	213	376	182	144	2	2	2
INCLINOMETERYMAX	226	488	186	168	3	3	3
INCLINOMETERYZC	269	438	743	205	3	2	3
GROUNDSPEED	209	263	550	164	5	3	5
DEPTH	282	444	2973	302	21	10	10
INCLINOMETERXZC	294	404	2554	276	16	11	11
MEPOWER	380	557	3376	383	15	9	9
METORQUE	367	561	2976	343	12	6	8
INCLINOMETERXMAX	263	431	2786	323	12	6	8

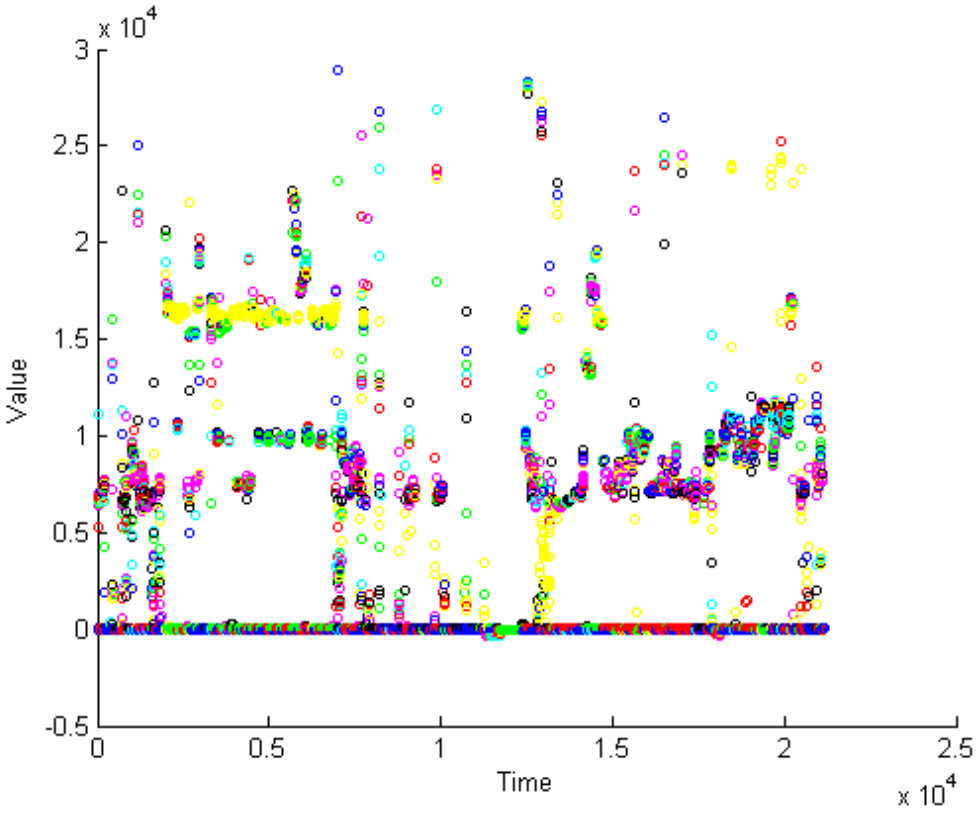
Ακολουθούν έγχρωμες εικόνες αποτελεσμάτων, που προέκυψαν από την εκτέλεση των αλγορίθμων πάνω στο σύνολο δεδομένων METORQUE. Για κάθε μετρική υπάρχουν δύο εικόνες. Η πρώτη δείχνει τα ομαδοποιημένα δεδομένα με την σειρά που εμφανίστηκαν σε βάθος χρόνου, ενώ η δεύτερη δείχνει τις συστάδες που δημιουργήθηκαν με τα δεδομένα τους.



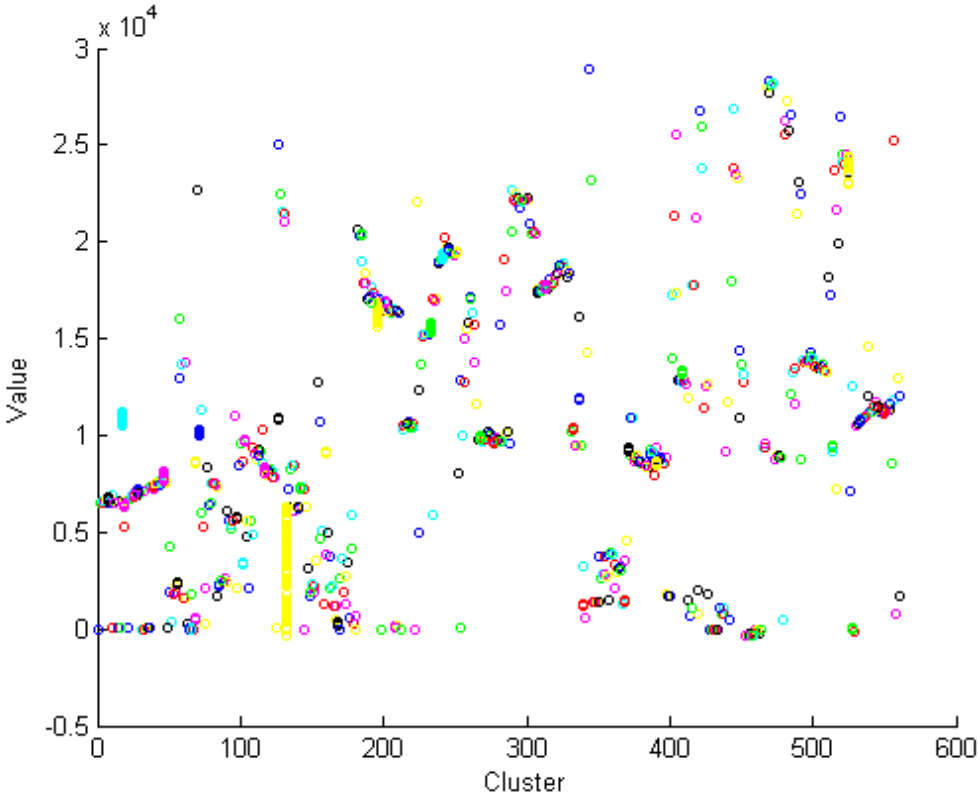
Εικόνα 7:Αποτέλεσμα CuSum Πολλαπλών καταστάσεων – METORQUE Dataset. (δεδομένα-χρόνος)



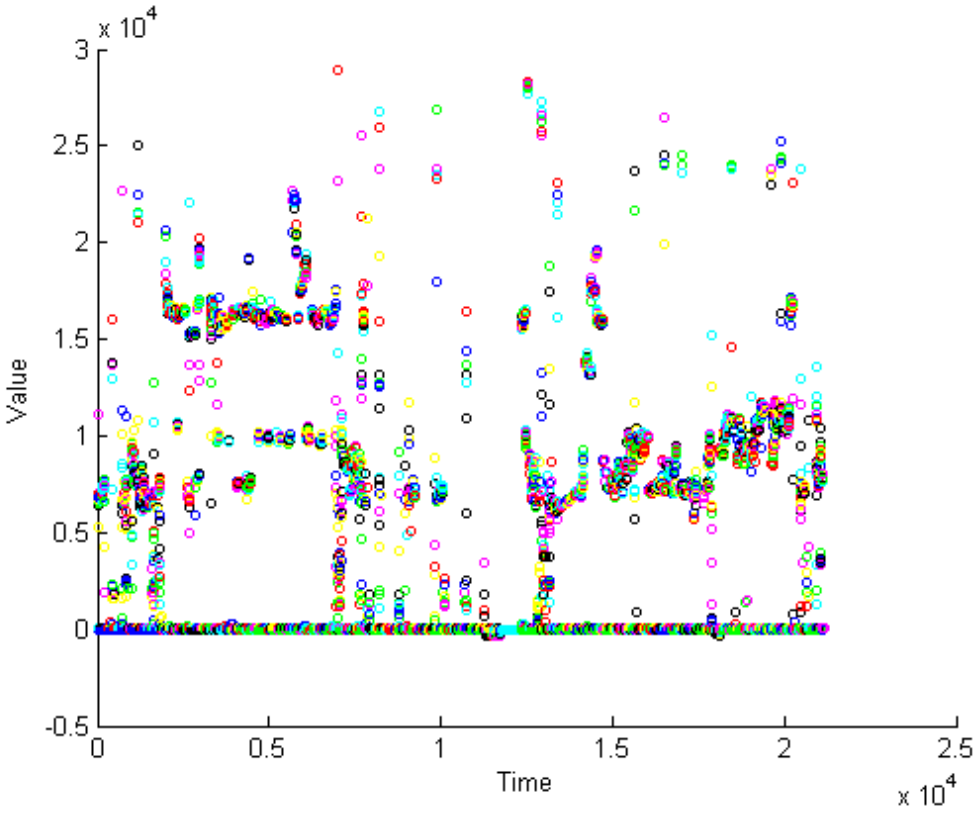
Εικόνα 8:Αποτέλεσμα CuSum Πολλαπλών καταστάσεων – METORQUE Dataset. (δεδομένα-συστάδες)



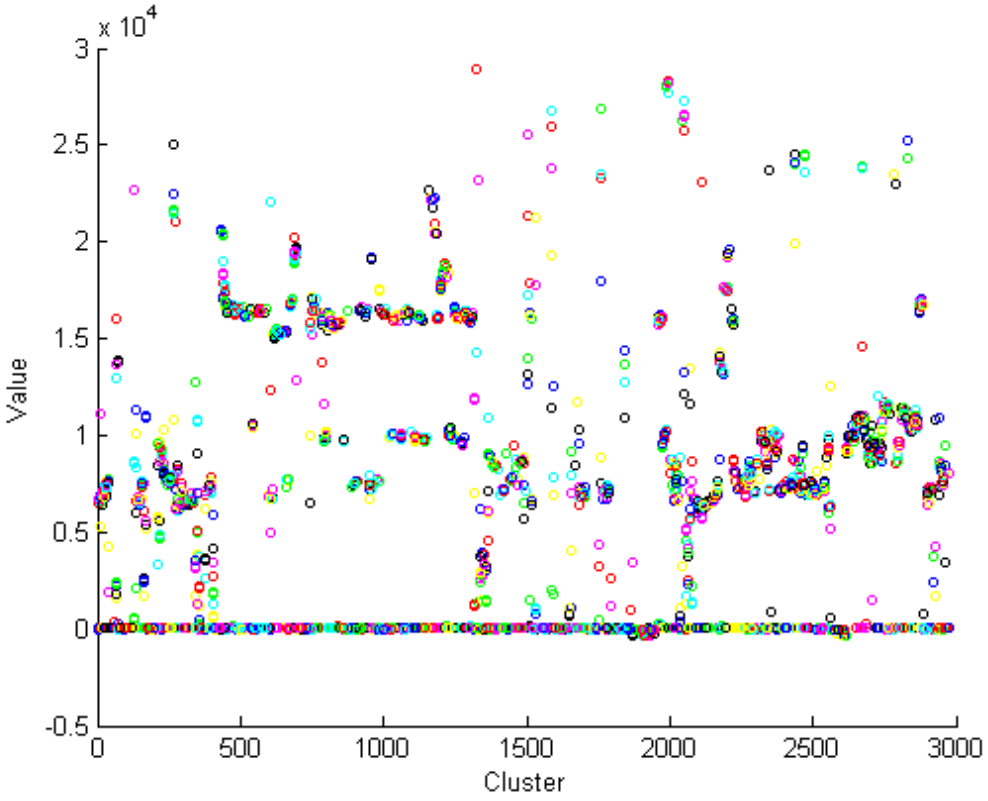
Εικόνα 9:Αποτέλεσμα Local Outlier Factor – METORQUE Dataset. (δεδομένα-χρόνος)



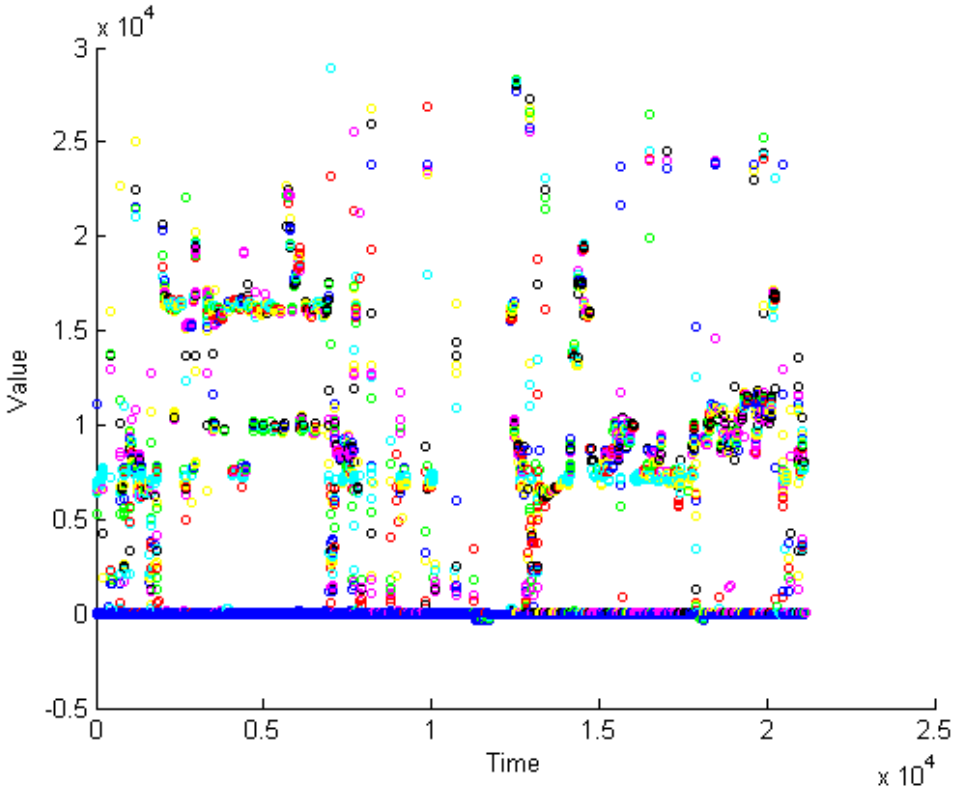
Εικόνα 10: Αποτέλεσμα Local Outlier Factor – METORQUE Dataset. (δεδομένα-συστάδες)



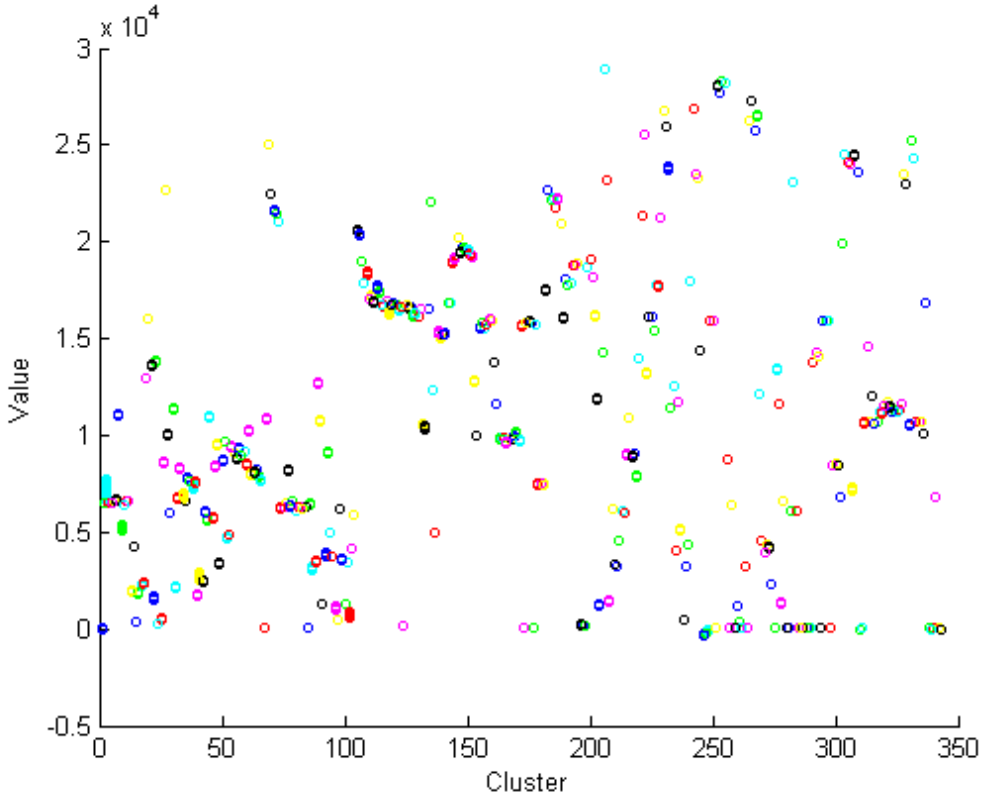
Εικόνα 11:Αποτέλεσμα Silhouette Coefficient – METORQUE Dataset. (δεδομένα-χρόνος)



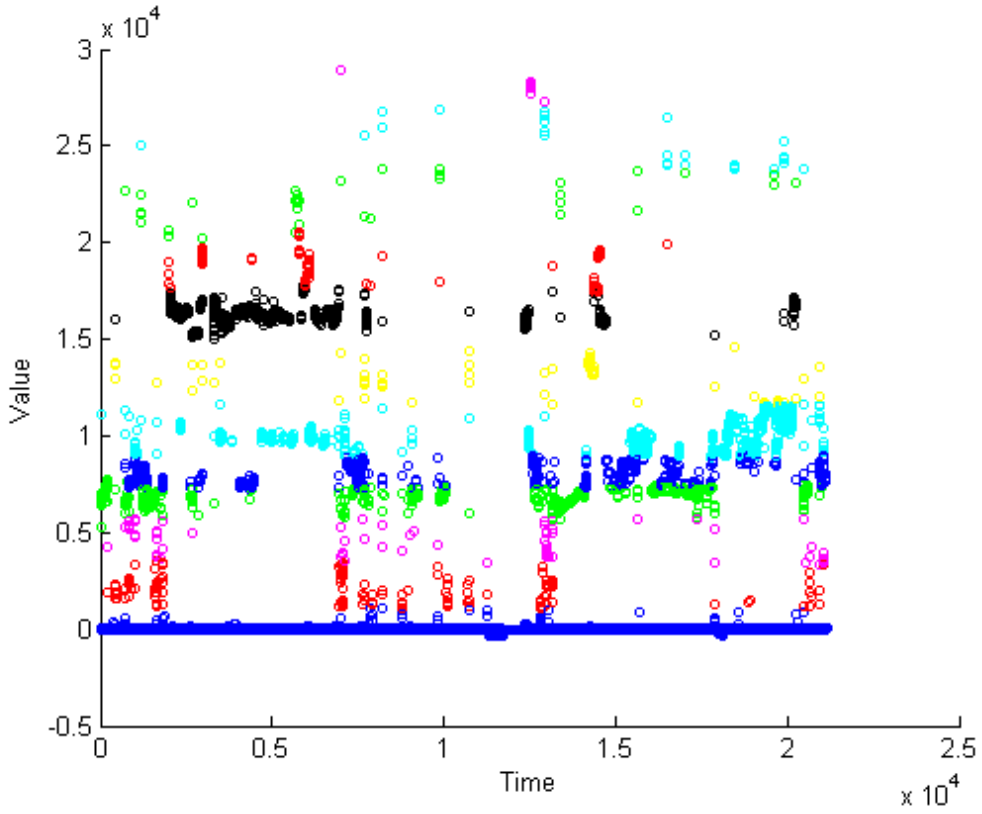
Εικόνα 12:Αποτέλεσμα Silhouette Coefficient – METORQUE Dataset. (δεδομένα-συστάδες)



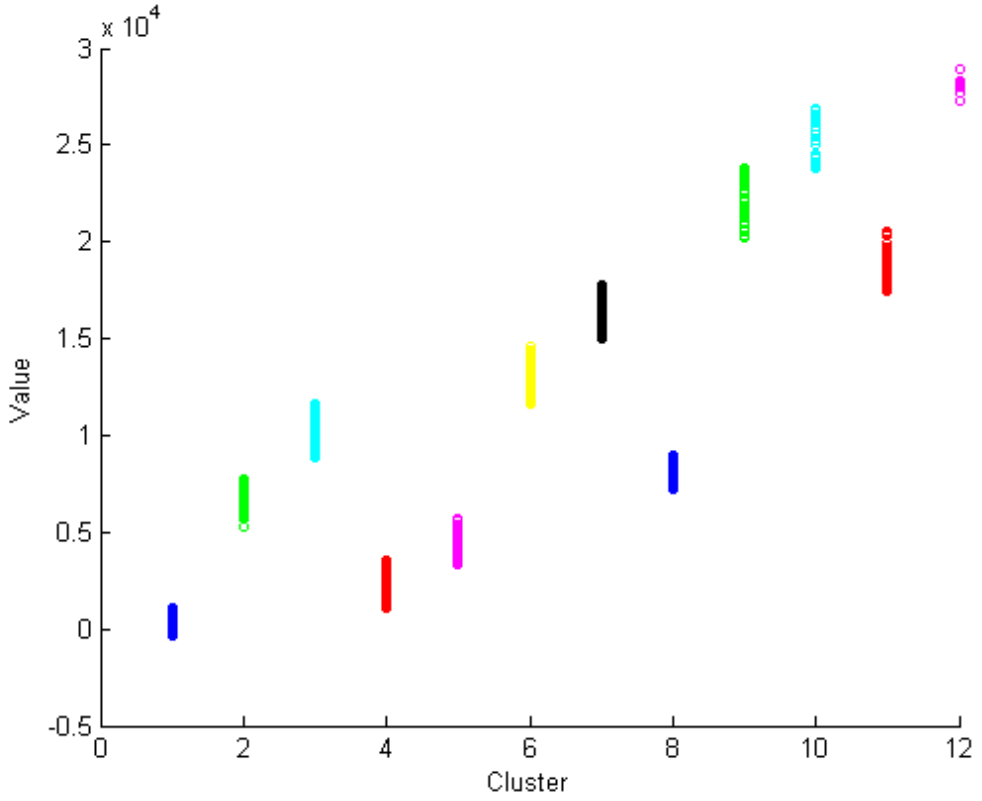
Εικόνα 13: Αποτέλεσμα Silhouette Coefficient+LOF – METORQUE Dataset. (δεδομένα-χρόνος)



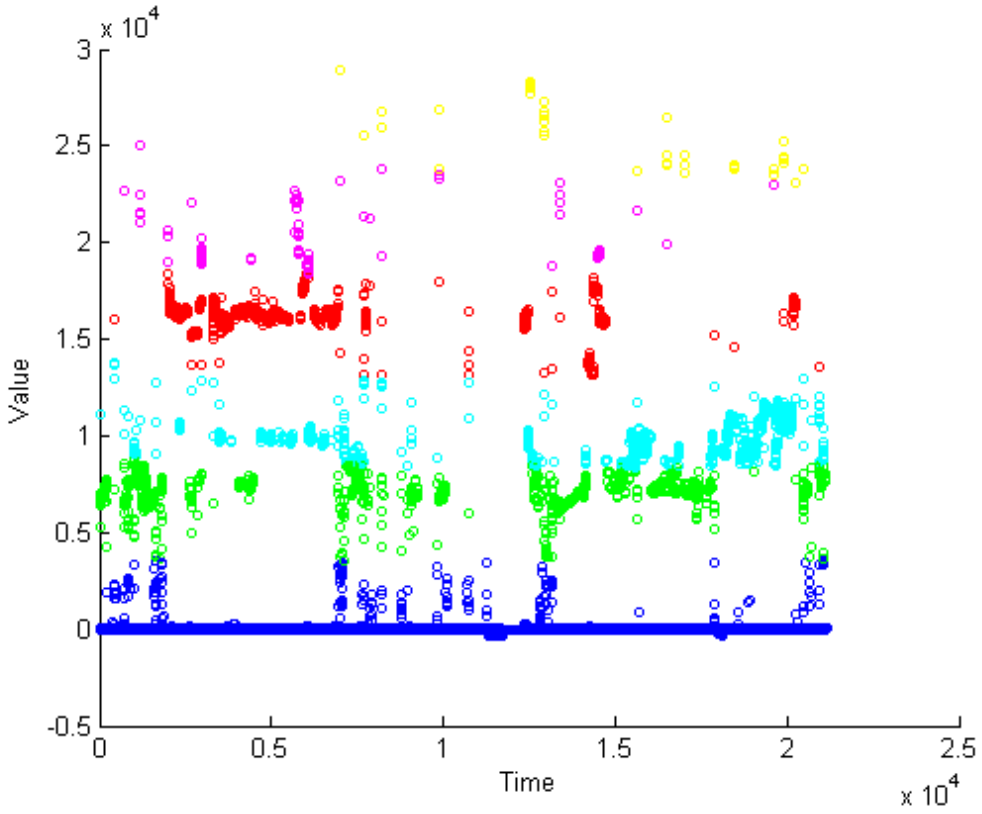
Εικόνα 14: Αποτέλεσμα Silhouette Coefficient+LOF – METORQUE Dataset. (δεδομένα-συστάδες)



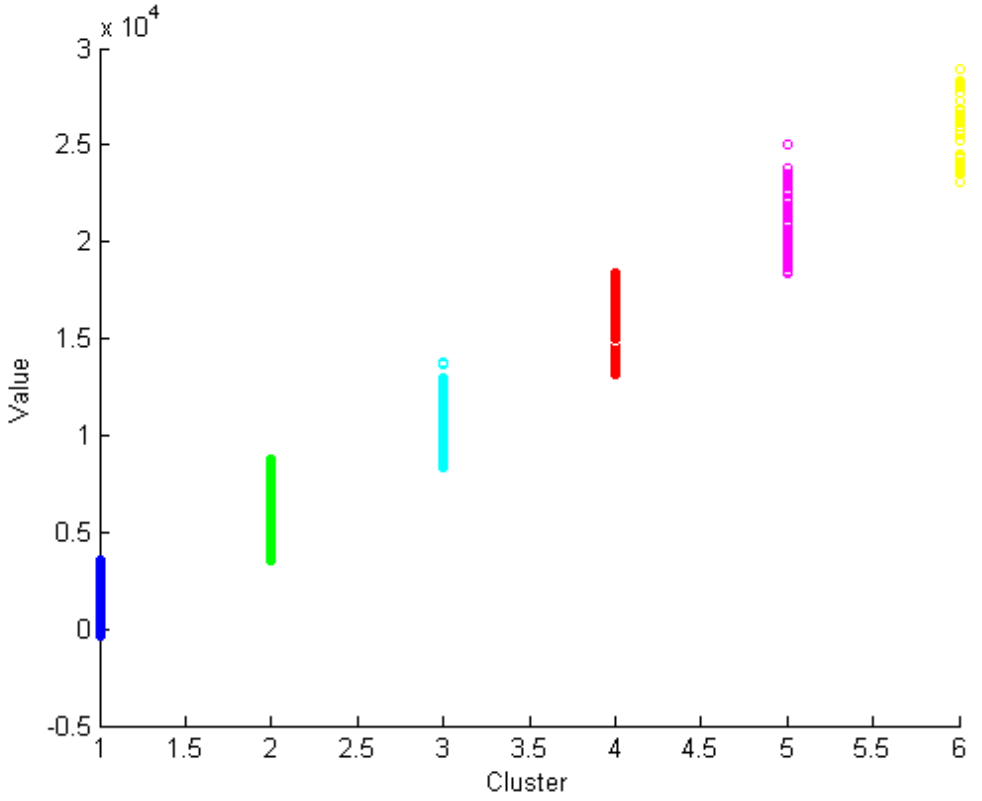
Εικόνα 15:Αποτέλεσμα Αλγόριθμος ελέγχου απόστασης – METORQUE Dataset. (δεδομένα-χρόνος)



Εικόνα 16: Αποτέλεσμα Αλγόριθμος ελέγχου απόστασης – METORQUE Dataset. (δεδομένα-συστάδες)



Εικόνα 17:Αποτέλεσμα Αλγόριθμος ελέγχου απόστασης(λιγότερο αυστηρός) – METORQUE Dataset. (δεδομένα-χρόνος)



Εικόνα 18: Αποτέλεσμα Αλγόριθμος ελέγχου απόστασης(λιγότερο αυστηρός) – METORQUE Dataset. (δεδομένα-συστάδες)

5.3. Χρόνοι Απόκρισης Συστήματος

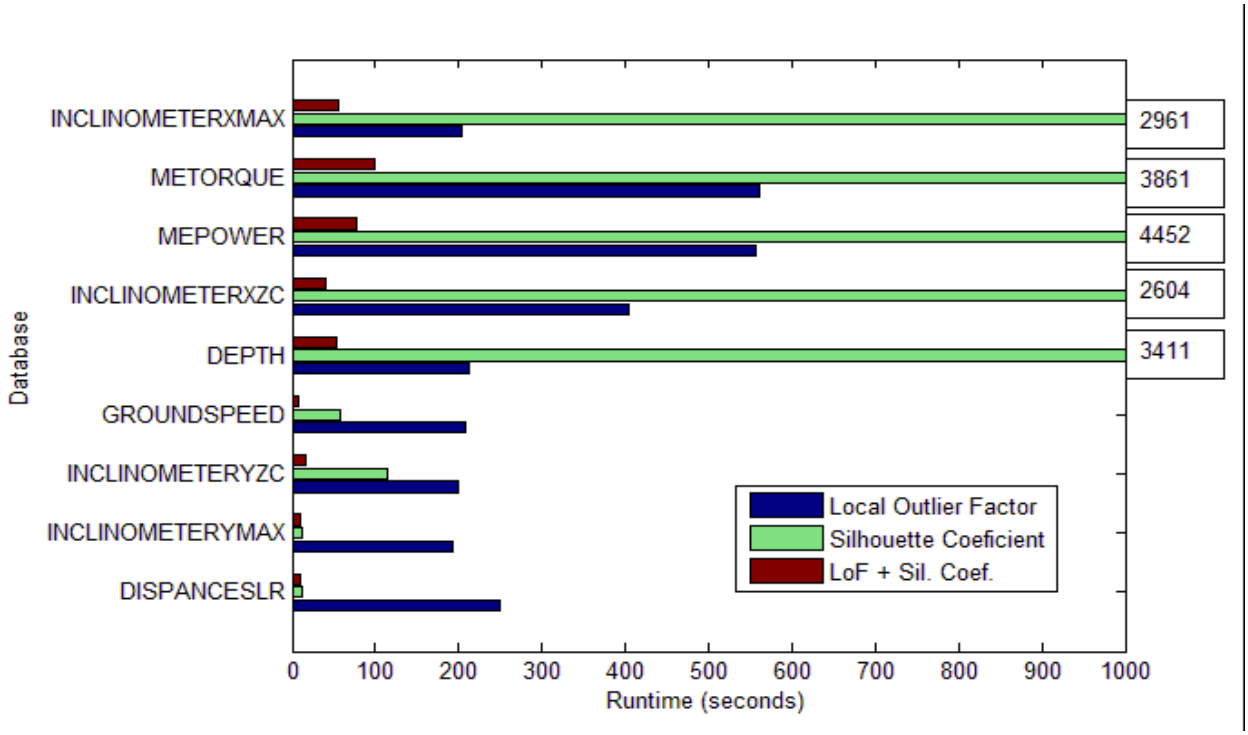
Στην ενότητα αυτή δίνεται ο πίνακας χρόνων απόκρισης κάθε αλγορίθμου για κάθε dataset, αλλά και το συγκεντρωτικό διάγραμμα που προκύπτει από τον πίνακα.

Πίνακας 4:Χρόνοι Απόκρισης αλγορίθμων ανίχνευσης αλλαγών

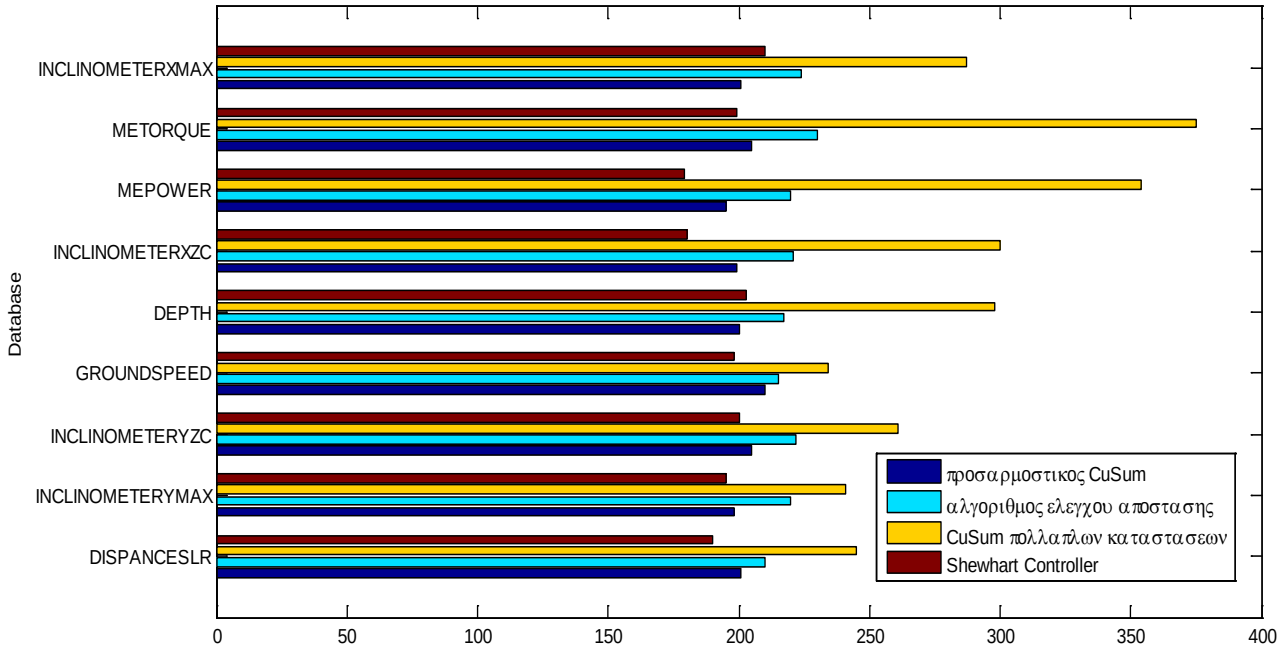
Μετρικές / Σύνολα	CuSum	Προσαρμοστικός CuSum	Shewhart Controller
DISPANCESLR	195ms	201ms	190ms
INCLINOMETERYMAX	183ms	198ms	195ms
INCLINOMETERYZC	190ms	205ms	200ms
GROUNDSPEED	200ms	210ms	198ms
DEPTH	203ms	200ms	203ms
INCLINOMETERXZC	187ms	199ms	180ms
MEPOWER	189ms	195ms	179ms
METORQUE	200ms	205ms	199ms
INCLINOMETERXMAX	199ms	201ms	210ms

Πίνακας 5:Χρόνοι Απόκρισης αλγορίθμων συσταδοποίησης

Μετρικές / Σύνολα	CuSum Πολλαπλών καταστάσεων	LoF	Sil. Coef.	Sil.Coef.+LoF	Αλγόριθμος ελέγχου απόστασης
DISPANCESLR	245ms	249s	11.9s	9s	210ms
INCLINOMETERYMAX	241ms	193s	12.5s	10.2s	220ms
INCLINOMETERYZC	261ms	200s	113.3s	15.5s	222ms
GROUNDSPEED	234ms	209s	57.3s	8.4s	215ms
DEPTH	298ms	212s	3411s	52.8s	217ms
INCLINOMETERXZC	300ms	404s	2604s	41.3s	221ms
MEPOWER	354ms	557s	4452s	77s	220ms
METORQUE	375ms	561s	3861s	99.7s	230ms
INCLINOMETERXMAX	287ms	204s	2961s	55.5s	224ms



Σχήμα 5.1: Συγκριτικά αποτελέσματα χρόνων απόκρισης συστήματος μεταξύ των αλγορίθμων LOF, Silhouette Coefficient και του συνδιασμού αυτών ανά dataset.



Σχήμα 5.2: Συγκριτικά αποτελέσματα χρόνων απόκρισης συστήματος μεταξύ των αλγορίθμων Προσαρμοστικού CuSum, ελέγχου απόστασης συτάδων, Cusum πολλαπλών και Shewhart Controller καταστάσεων ανά dataset.

5.4. Συμπεράσματα

5.4.1. CuSum

Ο CuSum είναι αιτιολογημένα ο πιο δημοφιλής αλγόριθμος ανίχνευσης αλλαγών. Παρόλο που η βασική του μορφή δεν είναι ιδανική (ενότητα 2.2.2.), η ευκολία τροποποίησης του τον καθιστά έναν από τους καλύτερους του είδους του.

5.4.2. Προσαρμοστικός CuSum

Ο προσαρμοστικός Cumulative Sum αλγόριθμος λειτουργεί σωστά υπό την προϋπόθεση ότι το σύνολο δεδομένων δεν ακολουθεί μια τυχαία κατανομή, αλλά ότι τα δεδομένα εμφανίζονται γύρω από μία ή περισσότερες συγκεκριμένες τιμές. Αυτές οι συγκεκριμένες τιμές αποτελούν τις βασικές καταστάσεις τις οποίες εξετάζει ο αλγόριθμος για αλλαγή. Σε αυτές τι περιπτώσεις ο αλγόριθμος αυτός παράγει καλύτερα αποτελέσματα απο τον βασικό CuSum.

Στην περίπτωση όπου τα δεδομένα ακολουθούν τυχαία κατανομή, ο προσαρμοστικός Cumulative Sum δεν μπορεί να αναγνωρίσει μια βασική κατάσταση και τα αποτελέσματα του είναι ανακριβή. Αυτό συμβαίνει γιατί στόχος του αλγορίθμου αυτού είναι να «θυμάται» την τελευταία κανονική κατάσταση για την περίπτωση που η αλλαγή είναι προσωρινή. Λειτουργεί δηλαδή υπό την προϋπόθεση ότι υπάρχουν βασικές καταστάσεις, μια ή και περισσότερες.

Συμπερασματικά ο προσαρμοστικός CuSum είναι μια πολύ καλη βελτίωση του βασικού CuSum, με πρακτικά ίδιους χρόνους εκτέλεσης, παρόλο που χρειάζεται κάποιες αρχικές τιμές για να λειτουργήσει.

5.4.3. Shewhart Controller

Η λειτουργία του Shewhart Controller παρουσιάζει κάποιες ομοιότητες με τον προσαρμοστικό CuSum που υλοποιήθηκε στο πλαίσιο αυτής της εργασίας. Στόχος του είναι να προσαρμόζεται στα δεδομένα τα οποία δέχεται και να μεταβάλει το κατώφλι ελέγχου του καταλλήλως.

Αντίθετα όμως με τον προσαρμοστικό CuSum ο οποίος λαμβάνει υπόψει μόνο τα ν τελευταία δεδομένα, ο Shewhart Controller τα λαμβάνει υπόψη όλα, ανεξαρτήτως του χρόνου εμφάνισης τους.

Αποτελεί ένα πολύ αποτελεσματικό και γρήγορο αλγόριθμο, ο οποίο όμως καλύπτει μόνο συγκεκριμένες ανάγκες συνόλων δεδομένων, ενώ ο τρόπος λειτουργίας του δυσκολεύει την τροποποίηση του.

5.4.4. CuSum Πολλαπλών καταστάσεων

Ο CuSum Πολλαπλών καταστάσεων ήταν μια καλή προσπάθεια συσταδοποίησης που όμως δεν είχε ικανοποιητικά αποτελέσματα. Η αδυναμία του να μεταβάλει το κατώφλι με το οποίο ελέγχει αν ένα στοιχείο ανοίκει σε μια ομάδα ή όχι περιορίζει την ακρίβειά του.

Πιο συγκεκριμένα μπορεί να λειτουργήσει σωστά μόνο σε σύνολα δεδομένων όπου τα δεδομένα ανοίκουν σε συστάδες με την ίδια διακύμανση. Σε αντίθετη περίπτωση δεν μπορεί συσταδοποιήσει τα στοιχεία ορθώς στην ίδια συστάδα αλλά δημιουργεί και τα εντάσει σε καινούργιες.

5.4.5. Local Outlier Factor

Ο Local Outlier Factor ή LOF είναι ένας ακριβής βοηθητικός αλγόριθμος συσταδοποίησης, ο οποίος όμως έχει μερικά προβλήματα. Το πιο βασικό είναι η πολυπλοκότητά του σε θέμετα χώρου και χρόνου. Η λειτουργία του βασίζεται στον έλεγχο κάθε στοιχείου σε κάθε συστάδα σε τακτά χρονικά διαστήματα, το οποίο σημαίνει εκθετική πολυπλοκότητα.

Επίσης ένα άλλο πρόβλημα που παρατηρήθηκε είναι η αδυναμία σωστής λειτουργίας του σε σύνολα δεδομένων με ίδιες τιμές. Η λειτουργία του βασίζεται στον υπολογισμό του k-distance. Εάν αυτά τα k στοιχεία έχουν την ίδια τιμή, τότε το k-distance είναι 0 και η χρήση του σαν διαιρέτης γίνεται αδύνατη.

Παρόλα αυτά η ακρίβειά του και η δυνατότητα να εντοπίζει κάθε είδους outlier τον καθιστά πολύ δημοφιλή και χρησιμοποιείται σε πολλούς αλγορίθμους με διάφορες παραλλαγές.

5.4.6. Silhouette Coefficient

Ο Silhouette Coefficient χρησιμοποιείται και αυτός, όπως και ο LOF, βοηθητικά μαζί με άλλους αλγορίθμους, αλλά αντί να εντοπίζει Outlier έχει ως στόχο την επιβεβαίωση της ορθότητας του βασικού αλγορίθμου.

Αποτελεί έναν απλό αλλά αποτελεσματικό αλγόριθμο, ο οποίος όμως έχει εκθετική πολυπλοκότητα χρόνου. Η πολυπλοκότητα αυτή βέβαια δεν σχετίζεται με τον αριθμό τον δεδομένων αλλά με τον αριθμό των συστάδων που δημιουργούνται.

5.4.7. Αλγοριθμος ελέγχου απόστασης συστάδων

Ο αλγόριθμος που υλοποίησα έχει πολλά από τα πλεονεκτήματα και κανένα από τα μειονεκτήματα των παραπάνω αλγορίθμων. Η αποτελεσματικότητά του βασίζεται στο γεγονός ότι προσαρμόζεται στο σύνολο δεδομένων το οποίο ελέγχει. Ο μηδαμινός χρόνος εκτέλεσης οφείλεται στην πολυωνυμική πολυπλοκότητά του, καθώς δεν χρειάζεται να αποθηκέυει ή να ελέγχει όλα τα στοιχεία των συστάδων, αφού χρησιμοποιεί πληροφορίες που υπολογίζονται εύκολα χρησιμοποιώντας τα βασικά χαρακτηριστηκά των συστάδων όπως την μέση τιμή.

Η ανάγκη του για είσοδο της τιμής της «αυστηρότητας» από τον χρήστη δεν μπορεί να θεωρηθεί ως μειονέκτημα καθώς δίνει ελαστικότητα στον αλγόριθμο.

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

CUSUM	Cumulative Sum
LOF	Local Outlier Factor

ΑΝΑΦΟΡΕΣ

1. Wade, T. and Sommer, S. eds. A to Z GIS
2. Michèle Basseville and Igor V. Nikiforov (April 1993). Detection of Abrupt Changes: Theory and Application. Prentice-Hall, Englewood Cliffs, N.J. ISBN 0-13-126780-9.
3. Grigg et al.; Farewell, VT; Spiegelhalter, DJ (2003). "The Use of Risk-Adjusted CUSUM and RSPRT Charts for Monitoring in Medical Contexts". Statistical Methods in Medical Research 12 (2): 147–170. doi:10.1177/096228020301200205. PMID 12665208
4. Estivill-Castro, Vladimir (20 June 2002). "Why so many clustering algorithms — A Position Paper". ACM SIGKDD Explorations Newsletter 4 (1): 65–75. doi:10.1145/568574.568575.
5. Sibson, R. (1973). "SLINK: an optimally efficient algorithm for the single-link cluster method". The Computer Journal (British Computer Society) 16 (1): 30–34. doi:10.1093/comjnl/16.1.30.
6. Defays, D. (1977). "An efficient algorithm for a complete link method". The Computer Journal (British Computer Society) 20 (4): 364–366. doi:10.1093/comjnl/20.4.364.
7. Lloyd, S. (1982). "Least squares quantization in PCM". IEEE Transactions on Information Theory 28 (2): 129–137. doi:10.1109/TIT.1982.1056489
8. Dempster, A.P., Laird, N.M.: Rubin, D.B. (1977). “Maximum Likelihood from Incomplete Data via the EM Algorithm”, Journal of the Royal Statistical Society. Series B (Methological) 39 (1): 1-38, http://www.jstor.org/discover/10.2307/2984875?uid=3738128&uid=2&uid=4&sid=56310793833, July 2012.
9. Kriegel, Hans-Peter; Kröger, Peer; Sander, Jörg; Zimek, Arthur (2011). "Density-based Clustering". WIREs Data Mining and Knowledge Discovery 1 (3): 231–240. doi:10.1002/widm.30.
10. Microsoft academic search: most cited data mining articles: DBSCAN is on rank 24, when accessed on: 4/18/2010
11. Ester, Martin; Kriegel, Hans-Peter; Sander, Jörg; Xu, Xiaowei (1996). Simoudis, Evangelos; Han, Jiawei; Fayyad, Usama M., eds. "Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)". AAAI Press. pp. 226–231. ISBN 1-57735-004-9. CiteSeerX: 10.1.1.71.1980.
12. Ankerst, Mihael; Breunig, Markus M.; Kriegel, Hans-Peter; Sander, Jörg (1999). "ACM SIGMOD international conference on Management of data". ACM Press. pp. 49–60. CiteSeerX: 10.1.1.129.6542.
13. Achtert, E., Bohm, C., Kroger, P, (2006), “DeLi-Clu: Boosting Robustness, Completeness, Usability, and Efficiency of Hierarchical Clustering by a Closest Pair Ranking”, http://www.springerlink.com/content/55x11867m7646464/?MUD=MP, July 2012
14. Hans-Hermann BOCK, “Origins and extensions of the k-means algorithm in cluster analysis”, Electronic Journal for History of Probability and Statistics, Vol 4, n°2, December 2008 : 6-7 http://www.jehps.net/Decembre2008/Bock.pdf, July 2012

15. Eduardo J. Spinosa, “OLINDDA: a cluster-based approach for detecting novelty and concept drift in data streams”.,Proceedings of the 2007 ACM symposium on Applied Computing, pp 448-452. http://dl.acm.org/citation.cfm?id=1244107, July 2012.
16. “Συσταδοποίηση”, University of Ioannina, http://www.cs.uoi.gr/~pitoura/courses/dm/cluster1-11.pdf, July 2012.
17. <<Η έννοια της απόστασης >>, http://www.samos.aegean.gr/actuar/dlekkas/environmental%20stats/environmental%20statistics%203.pdf,.July 2012.
18. Ming Li, Department of Computer Science and Technology Nanjing University (2011), “Data Mining “,Chapter 2: Measurement and Data, http://cs.nju.edu.cn/lim/courses/dm/slides/Chapter2.pdf,July 2012.
19. Peter J. Rousseeuw (1987). "Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis". Computational and Applied Mathematics http://www.sciencedirect.com/science/article/pii/0377042787901257.
20. Breunig, M. M.; Kriegel, H.-P.; Ng, R. T.; Sander, J. (2000). "LOF: Identifying Density-based Local Outliers". Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. SIGMOD: 93–104. doi:10.1145/335191.335388. ISBN 1-58113-217-4
21. Schubert, E.; Zimek, A.; Kriegel, H. -P. (2012). "Local outlier detection reconsidered: A generalized view on locality with applications to spatial, video, and network outlier detection". Data Mining and Knowledge Discovery. doi:10.1007/s10618-012-0300-z