Υλοποίηση του αλγορίθμου DBSCAN και η εφαρμογή του σε δεδομένα της αγοράς

Φωτεινή Καλαφάτη
Πολυτεχνείο Κρήτης
Σχολή Μηχανικών Παραγωγής και Διοίκησης
Πολυτεχνειούπολη, 73100 Χανιά
email: fot.kalafati@yahoo.com

Περίληψη

Τα τελευταία χρόνια η αναγκαιότητα της ύπαρξης αναλυτικών, με τις οποίες θα πραγματοποιείται η ανάλυση των πληροφοριών που βρίσκονται μέσα σε μεγάλο όγκο δεδομένων γίνεται όλο και πιο επιτακτική. Για τον παρόντα λόγο, καθώς επίσης και για την πολυπλοκότητα των δεδομένων εντείνεται η αναγκαιότητα της χρήσης τεχνικών ομαδοποίησης. Επίσης, ο μεγάλος και πολύπλοκος όγκος δεδομένων λαμβάνει χώρα σε πραγματικές εφαρμογές, με αποτέλεσμα να χρειάζεται προ επεξεργασία των δεδομένων προτού εφαρμοστεί μια τεχνική μέθοδος, προκειμένου να γίνεται μια καλύτερη και γρηγορότερη επεξεργασία τους.
Ο Tryon (1939) ήταν πρώτος που χρησιμοποίησε τον όρο «cluster analysis». Σύμφωνα με τον όρο «ομαδοποίηση» εννοείται η διαδικασία σύμφωνα με την οποία, παρέχεται ένα σύνολο δεδομένων, το οποίο περιγράφεται από διάφορες ιδιότητες και χαρακτηριστικά (μεταβλητές). Ο στόχος μας είναι να διερευνηθούν οι μεταβλητές μας, ως προς τα χαρακτηριστικά τους, με στόχο την ομαδοποίησή τους (classes, clusters), με τέτοιο τρόπο ώστε τα δεδομένα της ίδιας ομάδας να διαθέτουν παραπλήσια χαρακτηριστικά με τη μικρότερη δυνατή απόκλιση μεταξύ τους, ενώ τα δεδομένα που ανήκουν σε διαφορετικές ομάδες να διαφέρουν όσο το δυνατόν περισσότερο. Συνεπώς, δύο είναι τα βασικά κριτήρια με βάση τα οποία πραγματοποιείται η ομαδοποίηση.
Τα κριτήρια είναι τα εξής:

1. Η εσωτερική ομοιογένεια
2. Η εξωτερική διαφοροποίηση

Επισημαίνεται ότι αρκετές φορές δεν δύναται ο διαχωρισμός των δεδομένων σε ομάδες με τέτοιο τρόπο ώστε να ικανοποιούνται τα παραπάνω κριτήρια. Εν αντιθέσει όμως δύναται η προσπάθεια ομαδοποίησης να καταλήξει στο παραπάνω συμπέρασμα. Σε περίπτωση όμως που γίνεται να πραγματοποιηθεί ομαδοποίηση, ο στόχος της διαδικασίας DBSCAN είναι να την υλοποιήσει.
Στην παρούσα εργασία, κατασκευάζεται ο αλγόριθμος DBSCAN και έπειτα εφαρμόζεται σε έναν σχετικά μικρό όγκο δεδομένων προκειμένου, να εξαχθούν ορισμένα συμπεράσματα, σχετικά με την εφαρμογή του αλγορίθμου. Επιπρόσθετα, πραγματοποιείται σύγκριση μεταξύ του DBSCAN και του K-means.

[DBSCAN, ομαδοποίηση, K-means]

1. Ο αλγόριθμος DBSCAN

Οι μέθοδοι, οι οποίες βασίζονται στην πυκνότητα έχουν την δυνατότητα να δημιουργούν συστάδες με αυθαίρετες μορφές και δεν χρειάζεται να προκαθορίζεται ο αριθμός των συστάδων, καθώς αναγνωρίζουν τα ακραία σημεία και δεν επηρεάζονται από τον θόρυβο. Όσον αφορά τον αλγόριθμο DBSCAN (Density Based Spatial Clustering of Applications with Noise) είναι ένας αλγόριθμος ομαδοποίησης δεδομένων, ο οποίος προτάθηκε από τον Martin Ester, Hans-Peter Kriegel, Jorg Sander και Xiaowei Xu το 1996. Συνεπώς, είναι ένας αλγόριθμος ομαδοποίησης με ένα ελάχιστο μέγεθος και πυκνότητα. Ο όρος πυκνότητα είναι το ελάχιστο πλήθος των σημείων, τα οποία απέχουν συγκεκριμένη απόσταση μεταξύ τους. Αυτό έχει σαν αποτέλεσμα να αντιμετωπίζεται το πρόβλημα των απομακρυσμένων σημείων καθώς τα σημεία αυτά δεν ομαδοποιούνται άρα δεν δημιουργείται συστάδα. Ο παρόν αλγόριθμος ως είσοδο έχει το MinPts και το Eps. Το MinPts είναι ο ελάχιστος αριθμός των

σημείων που μπορούν να εισαχθούν σε κάποια συστάδα. Επιπρόσθετα, για κάθε σημείο της συστάδας είναι αναγκαίο να υπάρχει κάποιο άλλο σημείο, του οποίου η απόσταση από το αρχικό σημείο να είναι μικρότερη από το κατώφλι εισόδου, Eps. Οι γείτονες ενός σημείου είναι το σύνολο των σημείων που απέχουν Eps από το σημείο. Όσον αφορά το πλήθος των συστάδων, δεν δίνεται ως είσοδο αλλά εν αντιθέσει προσδιορίζεται από τον ίδιο τον αλγόριθμο. [1], [2]

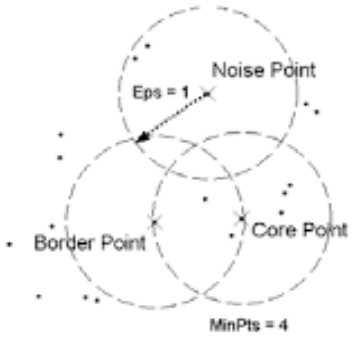
Συγκεκριμένα ο DBSCAN κάνει χρήση μια νέα έννοια της πυκνότητας. Στην συνέχεια αναπαρίστανται σε σχήμα (σχήμα 1) τα σημεία, τα οποία είναι άμεσα προσεγγίσιμα με βάση την πυκνότητα (directly density-reachable). Το dis(p,q) διασφαλίζει ότι το δεύτερο σημείο είναι πολύ κοντά στο πρώτο σημείο. Επιπλέον, όσον αφορά το δεύτερο μέρος του περιορισμού δείχνει ότι διατίθενται αρκετά σημεία πυρήνες (core points) σε κοντινές μεταξύ τους αποστάσεις. Αυτό έχει σαν αποτέλεσμα τα σημεία, που προκύπτουν με βάση τον περιορισμό, να απαρτίζουν μια συστάδα καθώς όλα τα σημεία βρίσκονται αρκετά κοντά. Όσον αφορά το οριακό σημείο (border point) είναι εκείνο το σημείο, το οποίο είναι άμεσα προσεγγίσιμο, με βάση τα παραπάνω, και θα πρέπει να ανήκει σε μια συστάδα, καθώς είναι κοντά σε ένα σημείο, το οποίο καλείται πυρήνας αλλά δεν είναι απαραίτητο το ίδιο να είναι πυρήνας.

Δίνοντας τιμές στις εισερχόμενες μεταβλητές Eps και MinPts, επιλέγεται ένα σημείο που καλείται p, το οποίο σύμφωνα με την πυκνότητα, προσεγγίζεται εύκολα από ένα άλλο σημείο q, αν και μόνο αν ισχύει [1], [2]:

- $dis(p,q) \leq Eps$ (1)
- $|\{r|dis(r,q) \leq Eps\}| \geq MinPts$ (2)

όπου,

p: Core Point
q: Border Point



Σχήμα 1: Απεικόνιση της συσταδοποίησης με DBSCAN

Έστω ότι υπάρχουν δύο σημεία ένα p και ένα άλλο q, τότε το p είναι προσεγγίσιμο από το q σύμφωνα με την πυκνότητα εφόσον υπάρχει αλυσίδα n σημείων p_1, p_2, p_3, …, p_n,και όπου p_1=p και p_1=q, και υπάρχουν μόνο σημεία από το ένα στο άλλο σημείο. Καθώς, επίσης καθένα εκ των δύο σημείων δύναται να προσεγγίζεται άμεσα, σύμφωνα με την πυκνότητα. από το προηγούμενο σημείο. Αυτό έχει σαν αποτέλεσμα κάθε συστάδα να αποτελείται από ένα σύνολο σημείων που βρίσκονται αρκετά κοντά σε πυρήνες και ορισμένα άλλα σημεία, τα οποία βρίσκονται πολύ κοντά σε έναν τουλάχιστον πυρήνα, τα ονομαζόμενα οριακά σημεία. [1]

Συνεπώς, συμπερένεται ότι για κάθε ζεύγος p,q:

- εάν το p ανήκει σε μια συστάδα C τότε και το q θα ανήκει στην C, εφόσον είναι προσεγγίσιμο με βάση την πυκνότητα.
- που ανήκει στην συστάδα C θα ισχύει ότι σημείο p θα είναι συνδεδεμένο με το σημείο q σύμφωνα με την πυκνότητα.

Μια συστάδα ορίζεται ως ένα σύνολο από σημεία, τα οποία είναι συνδεδεμένα μεταξύ τους με βάση την πυκνότητα. Ως θόρυβος ορίζονται εκείνα τα σημεία τα οποία δεν ανήκουν σε κάποια συστάδα. [1]

2. *Η υλοποίηση και τα αποτελέσματα του αλγορίθμου*

Όπως αναφέρεται στην ενότητα 1.1, ο αλγόριθμος DBSCAN καλείται αλγόριθμος συσταδοποίησης και σύμφωνα με την έννοια της πυκνότητας, δημιουργεί συστάδες. Στη συνέχεια γίνεται ανάλυση του αλγορίθμου, ο οποίος έχει υλοποιηθεί στη matlab, προκειμένου να γίνει καλύτερα κατανοητή η διαδικασία της συσταδοποίησης. [2]

Ο αλγόριθμος συσταδοποίησης αποτελείται από τρεις συναρτήσεις. Στο κυρίως μέρος του προγράμματος εντοπίζεται η συνάρτηση DSCAN, όπως φαίνεται και παρακάτω:

```
clc
clear
    eps=3500;
    MinPts=3;
    D=xlsread('Wholesale customers data.xlsx');
[IDD, noise]=DBSCAN(D,eps,MinPts);
```

Αρχικά ορίζεται ένα κατώφλι eps και ο ελάχιστος αριθμός των σημείων που ορίζουν μια κλάση, δηλαδή ο ελάχιστος αριθμός των σημείων που είναι επιθυμητό να διαθέτει κάθε κλάση. Στην παρούσα περίπτωση η ελάχιστη απόσταση είναι eps = 3500 διότι τα δεδομένα έχουν αρκετά υψηλές τιμές και ο ελάχιστος αριθμός των σημείων σε μια κλάση έχει ορισθεί το MinPts = 3. Στη συνέχεια διαβάζεται το αρχείο excel με τα δεδομένα μας με τη χρήση της εντολής xlsread και τοποθετούνται σε έναν πίνακα D. Έπειτα καλείται η συνάρτηση DBSCAN, η οποία επιστρέφει τις κλάσεις, που ανήκουν τα σημεία, σε έναν πίνακα IDD και επιστρέφει και τον πίνακα noise, ο οποίος δείχνει ποια σημεία έχουν ορισθεί ως θόρυβος και ποια όχι.

Στην συνάρτηση DBSCAN δίνεται ως είσοδος το σύνολο των δεδομένων (D), η ελάχιστη απόσταση των σημείων (eps) και ο ελάχιστος αριθμός των σημείων σε μια κλάση

```
function [IDD, noise] = DBSCAN(D,eps,MinPts)
    C=0;

    n=size(D,1);

    IDD=zeros(n,1);

    visited=zeros(n,1);
    noise=zeros(n,1);

    for i=1:n
        if visited(i)==0
            visited(i)=1;
            NeighborPts=regionQuery(D,i,eps);

            if numel(NeighborPts)< MinPts
                noise(i)=1;
            else
                C=C+1;
[IDD,visited,noise]=ExpandCluster(IDD,D,i,NeighborPts,C,MinPts,eps,v
isited,noise);

            end
        end
    end
end
```

(MinPts). Στη συνέχεια γίνεται αρχικοποίηση της μεταβλητής C, που δείχνει την κλάση, δίνοντάς την τιμή μηδέν. Όπου, n είναι το μέγεθος του δείγματος (D). Στη συνέχεια, ο πίνακας IDD, ο οποίος κρατάει τις κλάσεις αλλά και ο πίνακας noise, ο οποίος κρατάει ποια σημεία ορίζονται ως θόρυβος, μηδενίζονται. Καθώς επίσης μηδενίζεται και ο πίνακας visited, ο οποίος αποθηκεύει τα σημεία, τα οποία έχουν μελετηθεί (επισκεφθεί). Έπειτα, με την χρήση μιας λούπας ελέγχονται όλα τα σημεία αν έχουν κάποιο γείτονα ή αν ορίζονται ως θόρυβοι. Ο έλεγχος αυτός γίνεται με τη κλίση της συνάρτησης NeighborPts και στη συνέχεια ελέγχεται με την if αν ο αριθμός των γειτόνων (numel(NeighborPts)) είναι μικρότερος από το όριο που έχει ορισθεί ως είσοδος. Αν είναι μεγαλύτερος από το ορισθέν όριο, τότε το σημείο ορίζεται ως θόρυβος. Εν αντιθέσει, αυξάνεται ο δείκτης της κλάσης κατά ένα και δημιουργείται μια συστάδα έπειτα από την κλίση της συνάρτησης Expancluster, στην οποία καλείται και η συνάρτηση regionQuery, η οποία αναλύεται στην συνέχεια ο τρόπος με τον οποίο υλοποιείται.

```
function
[IDD,visited,noise]=ExpandCluster(IDD,D,i,NeighborPts,C,MinPts,eps,v
isited,noise)

        IDD(i)=C;

        k = 1;

        while true
            j = NeighborPts(k);

            if j~=0
                if visited(j)==0
                    visited(j)=1;

                    NeighborPts1=regionQuery(D,j,eps);

                    if numel(NeighborPts1)>=MinPts
                       NeighborPts=[NeighborPts NeighborPts1];
                    end
                end

                if IDD(j)==0
                    noise(j)=0;
                    IDD(j)=C;
                end
            end

            k = k + 1;

            if k > numel(NeighborPts)
                    break;
            end

        end
end
```

Η συνάρτηση ExpandCluster δέχεται ως εισόδους τις εξής μεταβλητές:

- IDD: πίνακας που αποθηκεύει τις κλάσεις
- D: το δείγμα που μελετάτε
- i: ο δείκτης που δείχνει ποιο σημείο μελετάτε

- NeighborPts: ο πίνακας που αποθηκεύει τους γείτονες για κάθε σημείο
- C: η κλάση
- MinPts: ο ελάχιστος αριθμός σημείων που πρέπει να ανήκουν σε μια κλάση και καθορίζεται από τον μελετητή
- Eps: η ελάχιστη απόσταση μεταξύ των σημείων
- visited: ο πίνακας που αποθηκεύονται όσα σημεία έχουν μελετηθεί
- noise: ο πίνακας, ο οποίος αποθηκεύει όσα σημεία ορίζονται ως θόρυβοι

Στη συνέχεια στον πίνακα IDD, στη θέση i του σημείου που μελετάτε, αποθηκεύεται η κλάση C στην οποία ανήκει. Ορίζεται μια μεταβλητή k, η οποία είναι βοηθητική προκειμένου να δύναται η σάρωση των γειτόνων. Όσο, λοιπόν, η συνθήκη είναι αληθής, πραγματοποιείται σάρωση των γειτόνων και ταυτόχρονα σε μια μεταβλητή j δίνεται η τιμή του γείτονα που υπάρχει στον πίνακα NeighborPts(k), για την αντίστοιχη τιμή k, κάθε φορά. Αν το j δεν είναι μηδέν τότε εκτελούνται οι ακόλουθες εντολές. Διαφορετικά αυξάνετε η τιμή της μεταβλητής k κατά μια μονάδα και ξεκινάει η διαδικασία από την αρχή. Εφόσον, λοιπόν το j είναι διάφορο του μηδενός, τότε αν για το αντίστοιχο j στον πίνακα visited ελέγχουμε αν είναι μηδέν δηλαδή, αν το αντίστοιχο σημείο δεν έχει μελετηθεί. Στην περίπτωση που είναι μηδέν, γίνεται ένα, δηλαδή θεωρείται ότι μελετήθηκε και καλείται η συνάρτηση regionQuery ώστε να βρεθούν οι γείτονες του σημείου. Αν ο αριθμός των γειτόνων του γείτονα είναι περισσότεροι από το MinPts τότε τους ομαδοποιεί όλους μαζί. Πραγματοποιείται, λοιπόν, ομαδοποίηση όλων των γειτόνων του γείτονα που βρίσκεται υπό εξέταση. Έπειτα, αν το IDD(j) είναι μηδέν, τότε βάζει τον κάθε γείτονα στην ίδια κλάση με τον αρχικό. Αυτό γίνεται σε περίπτωση που είναι αταξινόμητος. Σε περίπτωση που σε προηγούμενη επανάληψη στον πίνακα noise το σημείο που μελετάτε έχει οριστεί ως θόρυβος, τώρα του δίνεται η τιμή μηδέν διότι εντάχθηκε στην κλάση C. Τέλος, αν το k έχει γίνει μεγαλύτερο από τον αριθμό των γειτόνων του αρχικού τότε σταματάει, διότι έχουν ελεγχθεί όλοι οι γείτονες.

```
function [NeighborPts]=regionQuery(D,ind,eps)


    n=size(D,1);
    NeighborPts=[];
    k=0;

for i = 1:size(D,1)

if i ~= ind
       v = D(ind,:)-D(i,:);
       dist2 = v*v';

       if sqrt(dist2) < eps
           k=k+1;
           NeighborPts(k)= i;
       end

    end
end

end
```

Η συνάρτηση regionQuery δέχεται ως είσοδο τις εξής μεταβλητές:

- D: το δείγμα που μελετάτε
- Eps: η ελάχιστη απόσταση μεταξύ των σημείων
- ind: ο δείκτης του σημείου που εξετάζεται

Αρχικά ορίζεται ως κενός ο πίνακας των γειτόνων NeighborPts ώστε να δύναται η αποθήκευση των γειτόνων, η οποίοι υπολογίζονται παρακάτω. Έπειτα δίνεται μια αρχική τιμή, μηδέν, στην μεταβλητή k, η οποία είναι βοηθητική για τον πίνακα NeighborPts ώστε να μεταβαίνει στο επόμενη κελί του πίνακα και να αποθηκεύει το σημείο το οποίο ανιχνεύθηκε ως γείτονας. Στη συνέχεια πραγματοποιείται σάρωση, με τη βοήθεια μιας for, όλων των σημείων εκτός από εκείνο που εξετάζεται. Εντός της λούπας γίνεται έλεγχος αν ο δείκτης i, ο οποίος είναι το σημείο που μελετάτε, είναι διαφορετικό από το j, όπου δείχνει τον γείτονα στο αντίστοιχο σημείο k και το έχει παρθεί ως είσοδο από την συνάρτηση ExpandCluster, τότε υπολογίζεται το διάνυσμα διαφορών στις επιμέρους διαστάσεις. Αυτός ο υπολογισμός γίνεται προκειμένου να μπορεί να υπολογιστεί ο πολλαπλασιασμός των πινάκων, ο οποίος επιστρέφει ως αποτέλεσμα την απόσταση μεταξύ των σημείων. Στη συνέχεια, μέσα στη λούπα, ελέγχεται αν η ρίζα της απόστασης των σημείων είναι μικρότερη από το κατώφλι (eps), το οποίο έχει δοθεί ως είσοδος στην main. Στην περίπτωση, την οποία είναι αληθής ο περιορισμός τότε η μεταβλητή k αυξάνετε κατά μια μονάδα και για την συγκεκριμένη τιμή k, προστίθεται το σημείο i στους γείτονες.
Στη συνέχεια παρουσιάζονται τα αποτελέσματα, τα οποία προέκυψαν από την εκτέλεση του παραπάνω αλγορίθμου. Τα δεδομένα αποτελούνταν από 440 πελάτες, τα οποία σχετίζονταν με αγορές που πραγματοποιούν σε ένα σούπερ μάρκετ. Οι κατηγορίες από τις οποίες πάρθηκαν τα δεδομένα είναι οι εξής: fresh, milk, grocery, frozen, detergents_paper, delicassen.

Eps	3500	MinPts	3
		Πλήθος Σημείων	Ποσοστό
Noise	0	145	32,95
Κλάση	1	287	65,23
Κλάση	2	4	0,91
Κλάση	3	4	0,91

Πίνακας 1: Αριθμός κλάσεων και πλήθος σημείων σε κάθε κλάση

Πραγματοποιώντας, επιπλέον δοκιμές τιμών όσον αφορά την απόσταση των σημείων (eps) προκύπτουν οι παρακάτω πίνακες (πίνακας 2 και πίνακας 3).

Eps	4000	MinPts	3
		Πλήθος Σημείων	Ποσοστό
Noise	0	115	26,14
Κλάση	1	310	70,45
Κλάση	2	7	1,59
Κλάση	3	5	1,14
Κλάση	4	3	0,68

Πίνακας 2: Αριθμός κλάσεων και πλήθος σημείων σε κάθε κλάση

Με βάση τα παραπάνω αποτελέσματα, αν αυξήσουμε την απόσταση μεταξύ των σημείων (eps) και κρατώντας σταθερή την τιμή, η οποία σχετίζεται με τον αριθμό των σημείων που ανήκουν σε μια κλάση (MinPts), προκύπτει ότι μειώνεται ο αριθμός των σημείων που ορίζονται ως θόρυβοι αλλά παρατηρείται ταυτόχρονη αύξηση των σημείων που ανήκουν στην κλάση 1. Δηλαδή παρατηρείται μια συσσώρευση των σημείων στην κλάση 1. Καθώς επίσης, προκύπτει ότι το δείγμα, πλέον, χωρίζεται σε 4 κλάσεις, εν αντιθέσει με προηγουμένως (πίνακας 1) που ο διαχωρισμός των δεδομένων γινόταν σε 3 κλάσεις.

Eps	3000	MinPts	3
		Πλήθος Σημείων	Ποσοστό
Noise	0	190	43,18
Κλάση	1	9	2,05
Κλάση	2	205	46,59
Κλάση	3	19	4,32
Κλάση	4	3	0,68
Κλάση	5	6	1,36
Κλάση	6	4	0,91
Κλάση	7	4	0,91

Πίνακας 3:Αριθμός κλάσεων και πλήθος σημείων σε κάθε κλάση

Σύμφωνα με τα παραπάνω αποτελέσματα, αν μειωθεί η απόσταση μεταξύ των σημείων (eps) και κρατώντας σταθερή την τιμή της μεταβλητής MinPts, προκύπτει ότι αυξάνεται ο αριθμός των σημείων που ορίζονται ως θόρυβοι σε σύγκριση και με τον πίνακα 1 και με τον πίνακα 2 αλλά παρατηρείται αύξηση των σημείων που ανήκουν στην κλάση 2. Επίσης, προκύπτει ότι το δείγμα, πλέον, χωρίζεται σε 7 κλάσεις, σε εν αντιθέσει με προηγουμένως, οι πίνακες 1 και 2, που χωρίζονταν σε 3 κλάσεις και σε 4 κλάσεις, αντίστοιχα. Επιπλέον, παρατηρείται ότι είναι πολύ μικρός ο αριθμός των σημείων που ανήκουν σε κάθε κλάση και αυτό συμβαίνει διότι το 43,18 του δείγματος δεν έχει ομαδοποιηθεί. Συνεπώς, όσο μειώνεται η απόσταση μεταξύ των σημείων τόσο θα αυξάνεται και ο αριθμός των σημείων που δεν θα ανήκουν σε κάποια κλάση και θα ορίζονται ως θόρυβοι. Αυτό είναι λογικό να συμβαίνει, καθώς όπως αναφέρθηκε–παραπάνω, το δείγμα αποτελείται από σημεία με αρκετά υψηλές τιμές και μεγάλές διαφορές μεταξύ των αποστάσεών τους.

2.1. Σύγκριση αποτελεσμάτων του αλγορίθμου DBSACN και K-MEANS

Στη συνέχεια πραγματοποιείται σύγκριση μεταξύ των αποτελεσμάτων του αλγορίθμου DBSCAN και k-means. Η σύγκριση τους γίνεται σύμφωνα με το σχολιασμό των αποτελεσμάτων που προέκυψαν με τη βοήθεια της εντολής silhouette. Η τιμή του δείκτη silhouette κυμαίνεται από -1 έως 1. Μια υψηλή τιμή του συντελεστή δηλώνει ότι τα αποτελέσματα είναι ορθά. Αν τα περισσότερα σημεία έχουν υψηλή τιμή στον συντελεστή silhouette, τότε η λύση της ομαδοποίησης είναι η καταλληλότερη. Σε περίπτωση που πολλά σημεία έχουν χαμηλή τιμή ή αρνητική τότε η ομαδοποίηση μπορεί να έχει είτε πάρα πολλές ή πολύ λιγότερες κλάσεις από τις απαιτούμενες.

Στη συνέχεια παρουσιάζονται τα αποτελέσματα που προέκυψαν με την εντολή silhouette για τον αλγόριθμο Dbscan για τα ίδια δεδομένα με την ενότητα 2. [3], [4]

Αριθμός κλάσεων	**Mean silhouette**
3	-0,0362
4	0,0458
7	-0,3473

Πίνακας 4: Μέσος όρος των αποτελεσμάτων για κάθε κλάση (DBSCAN)

Τα αποτελέσματα του πίνακα 4 προέκυψαν με τη χρήση της εντολής silhouette και έπειτα υπολογίζοντας το mean του πίνακα, ο οποίος έχει προκύψει από την εντολή αυτή (silhouette). Συμπερένεται, ότι για τις παραπάνω κλάσεις, ο κατάλληλος διαχωρισμός είναι στις 4 κλάσεις. Η διαφορά με τις 3 κλάσεις είναι πάρα πολύ μικρός αριθμός. Όπως, προαναφέρθηκε στην ενότητα 1.2, αν χωριστεί το δείγμα σε 4 κλάσεις παρατηρείται συσσώρευση των σημείων στην κλάση 1 αλλά και ταυτόχρονη μείωση των σημείων που ορίζονται ως θόρυβοι. Εν αντιθέσει με το αν χωριστεί σε 3 κλάσεις παρατηρείται μικρή αύξηση των σημείων που ορίζονται ως θόρυβοι και μείωση των σημείων που ανήκουν στην κλάση 1. Αν επιλεγεί το δείγμα να χωριστεί σε 7 κλάσεις παρατηρείται ότι ο συντελεστής silhouette μειώνεται αρκετά, συνεπώς δεν είναι επιθυμητός ο διαχωρισμός αυτός.

Όσον αφορά τα αποτελέσματα, τα οποία προκύπτουν με την εντολή silhouette για τον αλγόριθμο k-means, παρουσιάζονται εν συνεχεία στον πίνακα 5. [3], [4]

Αριθμός κλάσεων	**Mean silhouette**
3	0,6492
4	0,5467
7	0,4595

Πίνακας 5: Μέσος όρος των αποτελεσμάτων για κάθε κλάση (K-means)

3. *Συμπεράσματα*

Ο στόχος της παρούσας εργασίας ήταν να υλοποιηθεί ο αλγόριθμος DBSCAN με σκοπό την ομαδοποίηση 440 αγοραστών, οι οποίοι πραγματοποιούν τις αγορές τους σε ένα σούπερ μάρκετ. Οι κατηγορίες από τις οποίες πάρθηκαν τα δεδομένα ήταν οι εξής: fresh, milk, grocery, frozen, detergents_paper, delicassen.

Η ομαδοποίηση πραγματοποιήθηκε με τον αλγόριθμο DBSCAN, κάνοντας δοκιμές όσον αφορά τις μεταβλητές eps και MinPts. Τα τελικά αποτελέσματα προέκυψαν βασιζόμενα σε 3, 4 και 7 κλάσεις, όπως αναλύονται και στο κυρίως κομμάτι της εργασίας. Έπειτα, για τα ίδια δεδομένα εφαρμόστηκε ο αλγόριθμος K-means για τον αντίστοιχο αριθμό των κλάσεων. Η εφαρμογή του K-means, πραγματοποιήθηκε με στόχο να γίνει μια σύγκριση των αποτελεσμάτων μεταξύ των δύο μεθόδων. Όπως, προκύπτει με βάση τα αποτελέσματα, ο Dbscan έχει ορθότερα αποτελέσματα με 4 κλάσεις, εν αντιθέσει με τον k-means που έχει πιο ορθά αποτελέσματα με 3 κλάσεις. Η ορθότητα των αποτελεσμάτων ελέγχθηκε με την εντολή silhouette. Για τα παρόντα δεδομένα προτείνεται ο αλγόριθμος Dbscan να υλοποιηθεί με eps = 3500 και MinPts = 3, τα οποία δίνουν ως έξοδο τρεις κλάσεις. Η παρούσα πρόταση, γίνεται σύμφωνα με τον αλγόριθμο k-means, ο οποίος είναι καταλληλότερος για τέτοιου είδους ομαδοποιήσεις.

Εν συνεχεία αναλύονται τα πλεονεκτήματα και τα μειονεκτήματα του αλγορίθμου DBSCAN, προκειμένου να τεκμηριωθεί ορθά η παραπάνω επιλογή και εν τέλη η πρόταση που έγινε.

Τα κυριότερα πλεονεκτήματα του αλγορίθμου DBSCAN παρουσιάζονται ακολούθως:

1. Ο αλγόριθμος DBSCAN εν αντιθέσει με τον k-means δεν απαιτεί εκ των προτέρων καθορισμό των κλάσεων - συστάδων.
2. Δύναται να καταλήξει σε αυθαίρετα σχήματα συστάδων. Καθώς επίσης μπορεί να εντοπίσει μια συστάδα, η οποία βρίσκεται γύρω από κάποια άλλη. Αυτό οφείλεται στην παράμετρο MinPts, η οποία ελαττώνει την εμφάνιση του φαινομένου της αλυσίδας συστάδων.
3. Δεν επηρεάζεται από ακραίες τιμές, καθώς έχει καλή ευαισθησία στο θόρυβο.
4. Χρειάζεται να ορισθούν μόνο δύο παράμετροι για είσοδο (ε, MinPts) και δεν επηρεάζεται από τη σειρά των δεδομένων στη βάση.
5. Οι παράμετροι MinPts και ε, είναι εύκολο να ορισθούν, από κάποιον έμπειρο στον τομέα αυτό, εφόσον τα δεδομένα έχουν κατανοηθεί.

Μολονότι, ο αλγόριθμος DBSCAN έχει αρκετά πλεονεκτήματα, διαθέτει και ορισμένα μειονεκτήματα, τα οποία παρουσιάζονται ακολούθως:

1. Ο DBSCAN δεν είναι απόλυτα ντετερμινιστικός, με την έννοια ότι τα οριακά σημεία μιας συστάδας δύναται να ανήκουν είτε σε αυτή είτε σε κάποια γειτονική της, ανάλογα με τη σειρά που θα μελετηθούν. Η παρούσα περίπτωση δεν συμβαίνει συχνά, και έχει μικρή επίδραση όσον αφορά τα αποτελέσματα της ομαδοποίησης.

2. Η ποιότητα των αποτελεσμάτων εξαρτάται από το μέτρο της απόστασης, το οποίο χρησιμοποιείται στη συνάρτηση regionQuery. Ο πιο συνηθισμένος υπολογισμός είναι η Ευκλείδεια απόσταση. Ειδικά για δεδομένα μεγάλων διαστάσεων, η συγκεκριμένη απόσταση είναι περιττή, λόγω της λεγόμενης «κατάρας της διαστατικότητας», με αποτέλεσμα να καθίσταται δύσκολος ο καθορισμός της τιμής ε. Αυτό δύναται να συμβεί σε οποιονδήποτε αλγόριθμο, ο οποίος χρησιμοποιεί την Ευκλείδεια απόσταση.
3. Σε περίπτωση που τα δεδομένα έχουν μεγάλες διαφορές πυκνότητας δεν δύναται να συσταδοποιηθούν σωστά. Αυτό συμβαίνει διότι δεν είναι εύκολος ο καθορισμός κατάλληλων τιμών για τις παραμέτρους MinPts και ε για όλες τις συστάδες.
4. Αν τα δεδομένα δεν έχουν κατανοηθεί ορθά και εις βάθος, τότε η επιλογή ενός σωστού κατωφλίου ε καθίσταται δύσκολη.

Συνεπώς δίνοντας ως είσοδο, τρεις διαφορετικές κλάσεις, στον αλγόριθμο K-means γίνεται εύκολα αντιληπτό ότι τα αποτελέσματα που προκύπτουν είναι πιο ορθά. Για αυτό το λόγο ο συνδυασμός, κατά μια έννοια, των αποτελεσμάτων των δύο αλγορίθμων κατευθύνει τον ερευνητή στην υλοποίηση και τη μελέτη του Dbscan για eps = 3500 και MinPts = 3.

Βιβλιογραφικές Αναφορές

- M. Ester, H. – P. Kriegel, J. Sander, X. XU (1996). A Density – Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. KDD vol.2, pp.226-231.

- Γ. Αυγουστής (2006). Ο αλγόριθμος Simulated Annealing στην κατευθυνόμενη στοχαστική αναζήτηση της βέλτιστης επιλογής χαρακτηριστικών για κατάταξη προτύπων χρονοσειρών από βάση δεδομένων Oracle με τον αλγόριθμο συσταδοποίησης DBSCAN. Πτυχιακή Εργασία, Σχολή Τεχνολογικών Εφαρμογών, Τεχνολογικό Εκπαιδευτικό Ίδρυμα Σερρών.

- Δ. Δεσπότης (2003). Ομαδοποίηση (Clustering). Σημειώσεις, Εργαστήριο Συστημάτων Υποστήριξης Αποφάσεων, Τμήμα Πληροφορικής, Πανεπιστήμιο Πειραιώς.

- Ε. Κρασάδακη (2015). Μέτρηση των προτιμήσεων των καταναλωτών, Τμηματοποίηση της αγοράς & Πρόταση Προϊόντος. Διαφάνειες Εργαστηρίου για το Μάθημα Μάρκετινγκ, Σχολή Μηχανικών Παραγωγής και Διοίκησης, Πολυτεχνείο Κρήτης, Χανιά.