

ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ ΓΙΑ ΑΛΓΟΡΙΘΜΙΚΑ ΠΡΟΒΛΗΜΑΤΑ

2η Προγραμματιστική Εργασία

Υλοποίηση δομής για την εύρεση κοντινών γειτόνων στη γλώσσα C/C++

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: Παπαγεωργίου Βασίλειος-Νικηφόρος
ΑΜ=1115201400140

Γενικές Πληροφορίες:

Η εργασία μου υλοποιεί το ζητούμενο και είναι χωρισμένο στα παρακάτω αρχεία .cpp , .h: το cluster.cpp(που αποτελεί την main στην ουσία), τα lsh.h και lsh.cpp(βιβλιοθήκη lsh απο την πρώτη εργασία) και τα cube.h και ψube.cpp(βιβλιοθήκη hypercube απο την πρώτη εργασία)).Επίσης, υπάρχει το αρχείο επικεφαλίδας structures.h με τον ορισμό των κλάσεων, τα datasets που μας έχετε δώσει και το αρχείο: structures.cpp , που περιέχει τις υλοποιήσεις των διαφόρων δευτερεύουσων και κύριων **ΔΙΚΩΝ ΜΟΥ** δομών(hashtable, lists καθώς επέλεξα να μην κάνω χρήση STL, διότι ήθελα να έχω πλήρη επίγνωση της κάθε δομής κι όχι να χρησιμοποιήσω κάτι έτοιμο) και αντίστοιχων συναρτήσεων της εργασίας(των ζητούμενων μεθόδων για Init, Assign, Update).Επίσης στο φακελο έχω φτιάξει και το αντίστοιχο Makefile και συνεπώς με την εντολή make μεταγλωττίζεται το αντίστοιχο πρόγραμμα και παράγεται το εκτελέσιμο cluster!Συνεπώς, τα πρόγραμμα μου εκτελούνται με την εντολή ./cluster, ακολουθούμενη από τις ζητούμενες παραμέτρους. Στην υλοποίηση μου, έχω υλοποιήσει όλες τις ζητούμενες μεθόδους Init, Assign, Update.Επιπλέον, **υλοποίησα και την προβολή σε υπερκύβο την οποία δεν είχα προλάβει στην πρώτη άσκηση**, ώστε να την χρησιμοποιήσω για το Assign.Ακόμη, έχω υλοποιήσει τη μορφή εκτύπωσης στο αρχείο output, ΑΚΡΙΒΩΣ, όπως ζητείται και με την προαιρετική χρήση του -complete, εκτυπώνει αναλυτικά τα σημεία από τα οποία αποτελείται κάθε cluster, αναλυτικά.Επίσης έχω κάνει **version control** , με χρήση **git**(**παρατίθεται screenshot παρακάτω**).

Η κυκλική διαδικασία assign-update στο πρόγραμμα μου, γίνεται είτε μέχρι να έχουμε σε δύο συνεχόμενες επαναλήψεις τα ίδια κεντροειδή, είτε μέχρι να φτάσουμε τις 30 επαναλήψεις.

Επιπλέον, στο cluster.conf εκτός απ' τις βοηθητικές παραμέτρους για τα cluster, το lsh και τον κύβο, προσδιορίζεται και ο τύπος Init, Assign, Update που θα χρησιμοποιήσει την εκάστοτε φορά ο αλγόριθμος.

```
sdi1400140@sdi1400140-Lenovo-Ideapad-110-15ACL: ~/Επιφάνεια εργασίας/project(data science)/2nd part/Project2(sdi1400140)/Clustering
sdi1400140@sdi1400140-Lenovo-Ideapad-110-15ACL:~/Επιφάνεια εργασίας/project(data science)/2nd part/Project2(sdi1400140)/Clustering$ git log .
commit e6b82b1e71ca2baf487861a25cf3f8962c0d9035 (HEAD -> master)
Author: sdi1400140 <sdi1400140@di.uoa.gr>
Date: Mon Dec 3 17:25:06 2018 +0200

    final push

commit 6f7f36ff5fc6d2671a62f82570f2ba8e09097f06 (origin/master)
Author: sdi1400140 <sdi1400140@di.uoa.gr>
Date: Mon Dec 3 16:50:31 2018 +0200

    Τερματικό change

commit 8b69510ef7ff72114366513a42fae358f63e8360
Author: sdi1400140 <sdi1400140@di.uoa.gr>
Date: Mon Dec 3 15:44:31 2018 +0200

    README added

commit ea173afd34056d3cf86339bebbaed6a0cedc180f
Author: sdi1400140 <sdi1400140@di.uoa.gr>
Date: Mon Dec 3 14:47:06 2018 +0200

    New pushhh

commit 4eaf929e942bb4a9ded19cf38241969e0851c9e1
Author: sdi1400140 <sdi1400140@di.uoa.gr>
Date: Mon Dec 3 13:41:41 2018 +0200

    Clustering push

sdi1400140@sdi1400140-Lenovo-Ideapad-110-15ACL:~/Επιφάνεια εργασίας/project(data science)/2nd part/Project2(sdi1400140)/Clustering$
```

INITIALIZATION

Random Init:Κάνω $K \text{ rand}() \% \text{numofpoints}$ (με έλεγχο για παραγωγή διαφορετικών αριθμών)κι ετσι παράγω τα indexes ,τον αρχικών centroids.

K-Means++:Ακολουθώ τον αλγόριθμο των διαφανειών παράγοντας με αυτό το τρόπο πιο “αντιπροσωπευτικά” αρχικά κέντρα(με μια καθυστέρηση σε σχέση με το Random βέβαια,η οποία εξισοροπείται λόγω της σύγκλισης σε λιγότερες επαναλήψεις).

ASSIGNMENT

Lloyd's:Ακολουθώ τον αλγόριθμο των διαφανειών υπολογίζοντας για κάθε σημείο το κοντινότερο κεντροειδές με βάση την αντίστοιχη μετρική

Assignment by Range search with LSH-Cube:Εδώ ακολουθώ τον αλγόριθμο των διαφανειών.Η αναζήτηση-διπλασιασμός ακτίνας σταματά είτε αν ανατεθούν όλα τα σημεία,είτε αν γίνουν 10 διπλασιασμοί χωρίς να αλλάξει κάτι(χωρίς να ανατεθεί κάποιο νεότερο σημείο).Στην τελευταία περίπτωση,αναθέτω τα εναπομείναντα στοιχεία με Lloyd's Assignment.Όπως λέει και στις διαφάνιες χρειάζεται κάπως να μαρκάρουμε τα ανατεθημένα σημεία.

Για τον παραπάνω λόγο,έχω εισαγάγει σε κάθε σημείο το πεδίο flag,το οποίο αρχικοποιώ με -1 και κάθε φορά που ανατίθεται σε κάποιο cluster,του δίνω την αντίστοιχη τιμή.Επιπλέον αν για κάποια ακτίνα κάποιο σημείο βρίσκεται σε πάνω απο 2 clusters,συγκρίνω την εκάστοτε απόσταση απ τα κεντροειδή και το αναθέτω στην μικρότερη.

UPDATE

Lloyd's:Για κάθε cluster υπολογίζω το βαρύκεντρο και το θέτω ως κεντροειδές.

PAM improved like Lloyd's:Για κάθε cluster υπολογίζω το medoid και το θέτω ως κεντροειδές.Εδώ για τον υπολογισμό του medoid,έχω σχολιασμένη μια βελτιστοποίηση δική μου όπου χρησιμοποιώ ένα πίνακα,για να αποθηκεύσω τις αποστάσεις τις οποίες έχω ήδη υπολογίσει(ώστε να αποφεύγονται περιττοί υπολογισμοί).Οι παρακάτω χρονοι-Silhouettes,είναι με βάση το εξαντλητικό PAM που προτείνεται στις διαφάνειες και το οποίο έχω αφήσει στο παραδοτέο μου.

ΕΝΔΕΙΚΤΙΚΕΣ ΕΚΤΕΛΕΣΕΙΣ(στο twitter dataset small):

Init = 1 , Assign =1 , Update =1

κ=2 time: 3.34298, Silhouette:0.04707918157308028518745653

κ=5 time:9.31395 , Silhouette:0.05790446549649541946951071

κ=10 time: 24.5632, Silhouette:0.06440656610136931918558648

κ=50 time:127.586 , Silhouette:0.1380604925247894610622693

κ=200 time:350.134 , Silhouette:0.245678390089927788488347

Init = 2 , Assign =1 , Update =2

κ=2 time:410.32 , Silhouette:0.0426934705407911221762161

κ=5 time:156.333 , Silhouette:0.04159385756406907099215985

κ=10 time: 71.9499, Silhouette:0.04842880400443286306903855

κ=50(Init=1) time: 35.9891, Silhouette:0.1087249036630299149476064

κ=200(Init=1) time: 134.907, Silhouette:0.1460112686627602766637098

Init = 2, Assign =3 (probes=3,dim=3), Update =1

κ=5 time:30.6332 , Silhouette:0.02869928640520400399330626

κ=10 time:56.5326 , Silhouette:0.02962183545784066375180545

κ=100(Init=1) time:481.348 , Silhouette:0.1610352871008647545573046

Init = 2 , Assign =3 (probes=3,dim=3) , Update =2

κ=5 time:278.839, Silhouette:0.02735969237771845192350368

κ=10 time: 123.805, Silhouette:0.03943234396363177159958787

κ=50(Init=1) time:214.177 , Silhouette:0.08160252066079894160357698

κ=100(Init=1) time:194.519 , Silhouette:0.12789098754328700907215643

ΣΥΜΠΕΡΑΣΜΑΤΑ(με βάση τα παραπάνω):

- Το PAM-Update,συγκλίνει σε πολύ λιγότερες επαναλήψεις από το Lloyds-Update,αλλά κάθε επανάληψη του διαρκεί περισσότερο λόγω των πολλών συγκρίσεων που κάνει.Τα αποτελέσματα για ίδια k,μεταξύ PAM και Lloyds είναι ελάχιστα καλύτερα(καλύτερο Silhouette)-πολύ κοντινά όμως- για το Lloyds.Ωστόσο,παρατηρούμε οτι αυξάνοντας το πλήθος των clusters,το PAM ολοκληρώνει σε λιγότερο χρόνο και μάλιστα **ΣΗΜΑΝΤΙΚΑ ΠΙΟ ΓΡΗΓΟΡΑ** απ'το Update like Lloyd's(πράγμα λογικό καθώς “χωρίζουμε” το dataset σε περισσότερα κομμάτια,άρα η εκάστοτε επανάληψη του PAM κάνει λιγότερες συγκρίσεις).ΟΜΟΙΑ,μειώνοντας τα clusters(πχ για κ=2),το PAM καθυστερεί πααρα πολύ,καθώς χωρίζει τον χώρο σε μεγάλα κομμάτια για τα οποία κάνει παρα πολλές συγκρίσεις.
- Το Assign με Range LSH ή Cube,είναι αρκετά αργό αλλά λόγω της καλύτερης διαχείρισης του χώρου που κάνει (ειδικά αυτό με τον κύβο),είναι ιδανικό και ίσως η μόνη αξιόπιστη λύση για το clustering σε big data ,όπως αυτά του big dataset.
- Το πλεονέκτημα του Lloyd's(assign-update) σε σχέση με τα υπόλοιπα είναι οτι ολοκληρώνεται σε πολύ λιγότερο χρόνο,ωστόσο δεν μπορεί να χρησιμοποιηθεί σε μεγάλα δεδομένα(Assign),και επίσης δεν παράγει τόσο καλά αποτελέσματα όσο το PAM για dataset εντελώς ανομοιόμορφο.
- Για μεγάλα K,το K-Means++ καθυστερεί πολύ περισσότερο απ το Random Init ,χωρίς να “αξίζει” αυτή η διαφορά στο αποτέλεσμα.
- Γενικότερα,αυξάνοντας το πλήθος των clusters,αυξάνεται το Silhouette και συνεπώς η ακρίβεια των αποτελεσμάτων.

*Οι παραπάνω μετρήσεις έγιναν με μετρική cosine. Η γενική παρατήρηση για την ευκλείδια μετρική είναι ότι γενικά έχει παρόμοια αποτελέσματα Silhouette με την cosine, αλλά λόγω της φύσης της απαιτεί αρκετά περισσότερο χρόνο (ειδικά για την παραγωγή του Silhouette) και συνεπώς είναι προτιμότερη η cosine.

Τα paths των αρχείων, δίνονται μέσω της γραμμής εντολών, από τις αντίστοιχες παραμέτρους. Το πρόγραμμα έχει ελεγχθεί και με valgrind για leaks και errors!

Περαιτέρω λεπτομέρειες παρέχονται σε σχόλια στα αρχεία και οτιδήποτε προκύψει είμαι διαθέσιμος να το αιτιολογήσω στην προφορική εξέταση!