Ανάπτυξη λογισμικού για Αλγοριθμικά Προβλήματα

Μάρκος Βαρβαγιάννης

sdi1400017@di.uoa.gr

2η Εργασία

Υλοποίηση των αλγορίθμων συσταδοποίησης K-means / K-medoids στη γλώσσα C/C++

Η άσκηση αφορά την υλοποίηση αλγορίθμων συσταδοποίησης (clustering) διανυσμάτων στον χώρο Rd, χρησιμοποιώντας 12 συνδυασμούς από τις παραλλαγές Initialization, Assignment και Update. Έχουν υλοποιηθεί σε C++ όλα τα ζητούμενα της εργασίας, όπως περιγράφεται στην εκφώνηση. Για τον αλγόριθμο assignment, χρησιμοποιήθηκε εκτός από τον απλό Lloyd’s, οι αλγόριθμοι LSH και Hypercube της πρώτης εργασίας για πιο γρήγορη αρχικοποίηση όταν χρησιμοποιούνται μεγάλα δεδομένα (big data).

Μεταγλώτιση

Για να μεταγλωτιστεί το πρόγραμμα τρέξτε **make**

Εκτέλεση

- ./cluster –i <input file> –c <configuration file> -o <output file> -d <metric>
- ./cluster –i <input file> –c <configuration file> -o <output file> -d
  <metric> -a
  σε περίπτωση που θέλουμε να τυπώσουμε τα στοιχεία του κάθε επιμέρους cluster (complete)

Μορφή αρχείων:

Το πρόγραμμα τρέχει στη μορφή των αρχείων input και query που δόθηκαν στο eclass

Αρχεία Κώδικα/Επικεφαλίδες:

- **clustering.cpp**
  Τα αρχείο που περιέχει τη main
- **clustering_helping.cpp**, **clustering_helping.h**
  Βοηθητικές συναρτήσεις για το clustering όπως οι συναρτήσεις που διαβάζουν το configuration file, το αρχείο με τα vectors, κλπ.
- **dVector.cpp**, **dVector.h**
  Περιέχει τον ορισμό της κλάσης του σημείου - vector
- **hFunction.cpp**, **hFunction.h**
  Περιέχει τον ορισμό της κλάσης συνάρτησης h
- **Cluster.cpp**, **Cluster.h**
  Περιέχει τον ορισμό της κλάσης του Cluster
- **init.cpp**, **init.h**

Περιέχει τις απαιτούμενες συναρτήσεις για την αρχικοποίηση του αλγορίθμου

- **assignment.cpp**, **assignment.h**
  Περιέχει τις απαιτούμενες συναρτήσεις για τις αναθέσεις του αλγορίθμου
- **lsh.cpp**, **lsh.h**
  Συναρτήσεις σχετικά με τον αλγόριθμο κοντινών γειτόνων lsh
- **cube.cpp**, **cube.h**
  Συναρτήσεις σχετικά με τον αλγόριθμο κοντινών γειτόνων hypercube
- **parameter_values.h**
  Περιέχει ως defines κάποιες σταθερές, όπως το w και το k για τους αλγόριθμους κοντινών γειτόνων

Μετρήσεις

Ο αλγόριθμος μπορεί να τρέχει με όλους τους 12 πιθανούς συνδυασμούς, απλώς εδώ θα αναφέρουμε τα βασικά συμπεράσματα. Παραδείγματα εκτέλεσης:

Init	Assignment	Update	k=5	k=20
Random	Lloyd’s	Kmeans	0.0409427	0.063523
Kmeans++	Lloyd’s	Kmeans	0.0416973	0.0680353

Initialization:
Ως συμπέρασμα, η ύπαρξη του kmeans++ βελτιώνει την απόδοση του αλγορίθμου, αλλά όχι πολύ σημαντικά, καθώς η βελτίωση εξαρτάται

πολύ και από την αρχικοποίηση του πρώτου κέντρου, και το πού βρίσκεται αυτό το σημείο στο χώρο.
Χρόνος: Ο πρώτος συνδυασμός έκανε 7 sec και ο δεύτερος 14 sec, κάτι που είναι αναμενόμενο, καθώς χρειάζεται και επιπλέον χρόνο ο kmeans++

Assignment:
Γενικά, παρατηρήθηκε ότι ο Lloyd's είναι καλύτερος όσον αφορά τις τελικές silhouettes. Παράδειγμα, για αριθμό clusters 20, ο **LSH** έβγαλε έξοδο **0.243543**, ενώ ο **υπερκύβος 0.015824**, την ώρα που βλέπουμε ότι με τον Lloyd's assignment έχουμε silhouettes κοντά στο 0.7 για τον ίδιο αριθμό clusters. Αυτό είναι λογικό να συμβαίνει επειδή κάνουμε hashing τα input vectors και δεν τα αναθέτουμε εξαντλητικά. Ωστόσο, στα πολύ μεγάλα δεδομένα (big data) οι αλγόριμοι hashing έχουν πολύ μεγάλο *χρονικό πλεονέκτημα*. Τέλος, η διαφορά του LSH με τον υπερκύβο, είναι όντως αναμενόμενη, καθώς όπως καταλήξαμε και στην πρώτη εργασία, το ποια θα είναι κατάλληλη μέθοδος για να χρησιμοποιήσουμε εξαρτάται αν θέλουμε να δώσουμε μεγαλύτερη βαρύτητα στο χρόνο (που επιλέγουμε τον LSH) ή στο χώρο (επιλέγουμε τον υπερκύβο).

Update:
Ο Kmedoids (PAM) improved like Lloyd's γενικά κάνει αρκετά παραπάνω ώρα, κάτι που είναι αναμενόμενο λόγω της τετραγωνικής πολυπλοκότητας που έχει. Επιπλέον, τα αποτελέσματα δεν έχουν μεγάλη απόκλιση μεταξύ τους, κάτι που μας οδηγεί στη χρήση του απλού Kmeans στην περίπτωση που έχουμε διανυσματικά δεδομένα και μπορούμε να υπολογίσουμε το μέσο όρο των διανυσμάτων για το νέο

κέντρο. Αν τα δεδομένα δεν είναι διανυσματικά, τότε η χρήση του PAM είναι αναπόφευκτη.

Σημείωση:

- Έχει χρησιμοποιηθεί το εργαλείο git για versioning.