

Ανάπτυξη Λογισμικού για Αλγοριθμικά Προβλήματα: Project 2

Σταματόπουλος Βασίλειος 1115201400188

3/12/2018

1 Εισαγωγή

Στην εργασία αυτή υλοποιήθηκαν 12 διαφορετικοί τρόποι clustering, όπως ακριβώς ζητήθηκε και στην εκφώνηση.

Δηλαδή, **2 τρόποι αρχικοποίησης των δεδομένων**, τυχαία και με k-means++ **2 τρόποι update**, **Basic Update με μέσα και Partitioning Around Medoids**. Επιπλέον, το assignment των σημείων γίνεται με τρεις τρόπους. **Normal k-means, Assignment with LSH και assignment with Hypercube**.

2 Αρχεία

Ο φάκελος περιέχει τα κάτωθι αρχεία στα οποία έγινε η υλοποίηση του κώδικα.

1. **cluster.c**: Περιέχει την main function στην οποία με τη σειρά διαβάζουμε τα arguments. ύστερα διαβάζουμε το αρχείο εισόδου και στη συνέχεια αρχικοποιούμε τις μεταβλητές που θα χρειαστούμε. Δημιουργείται ένας πίνακας point* data, ο οποίος περιέχει στοιχεία struct point που έχουν την παρακάτω μορφή

```
typedef struct point{
    long int id;
    double * coordinates;
    long long int ** g_functions;
    int centroid_id;
    int centroid2_id;
    double dist; //distance from centroids
    double dist_as_centroid;
    struct point * next;
    double silhouette;
} *point;
```

Πάνω σε αυτά τα στοιχεία κάνουμε την διαδικασία της συσταδοποίησης. Αρχικοποιούμε τα κέντρα που είναι τύπου struct centroid με την ακόλουθη μορφή

```
typedef struct centroid{
long int id;
int count;
int prev_count;
double dist;
long long int ** g_functions;
double * coordinates;
double silhouette_of_cluster;
point* assigned_points;
} *centroid;
```

Αφού τα αρχικοποιήσουμε με την εκάστοτε μέθοδο, κάλουμε την συνάρτηση **k-means** η οποία είναι αυτή που κάνει την δουλεία καλώντας στην πορεία τις απαραίτητες συναρτήσεις που βρίσκονται στο αρχείο functions.c/functions.h. Κατά τη διάρκεια όλης αυτής της διαδικασίας εκτυπώνονται τα κατάλληλα στοιχεία στο αρχείο εξόδου, με τον ίδιο τρόπο που περιγράφεται και στην εκφώνηση.

2. **functions.c:** Στο αρχείο αυτό βρίσκουμε όλες τις συναρτήσεις που χρησιμοποιεί ο k-means, με μια μεγάλη πλειοψηφία εξ'αυτών να προέρχονται από τον φάκελο της πρώτης εργασίας. Λόγω του μεγάλου όγκου του αρχείου, οι συναρτήσεις δεν περιγράφονται εδώ, αλλά έχουν τα απαραίτητα σχόλια μέσα στο ίδιο το αρχείο.
3. **hashtable.c hashtable.h** Τα αρχεία αυτά περιέχουν τους τύπους των στοιχείων που χρησιμοποιήθηκαν καθώς και τις απαραίτητες συναρτήσεις για να λειτουργήσει ένας πίνακας κατακερματισμού, όπως οι insert, print. Όπως τα δύο προηγούμενα, είναι κατάλληλα σχολιασμένα για τρίτους.

3 Μεταγλώττιση

Το πρόγραμμα χρησιμοποιεί makefile, συνεπώς μπορεί να μεταγλωττιστεί με την εντολή make.

4 Εκτέλεση

Η εκτέλεση του αρχείου ακολουθεί το πρότυπο που δώθηκε στην εκφώνηση

5 Version Control

Για το πρόγραμμα χρησιμοποιήθηκε το github για να γίνει το απαραίτητο version control.[1]

6 Αποτελέσματα

Παρακάτω δίνονται τα αρχεία αποτελεσμάτων για τις default τιμές, για τους δύο διαφορετικούς τύπους μετริกών

6.1 Euclidean Distance

6.2 Cosine Similarity

6.3 Πορίσματα

References

- [1] Vasileios Stamatopoulos, Github
<https://github.com/billstam12/Project/tree/master/2>