

Πρώτη Σειρά Ασκήσεων

Αυτό είναι το δεύτερο μέρος της πρώτης σειράς ασκήσεων. Η προθεσμία για την παράδοση αυτού του κομματιού είναι στις 25 Νοεμβρίου, στο ξεκίνημα του μαθήματος. Για καθυστερημένες υποβολές ισχύει η πολιτική στην σελίδα του μαθήματος. Οι λεπτομέρειες για το turn-in είναι στη σελίδα του μαθήματος.

Ερώτηση 1

Υποθέστε ότι σας δίνεται σαν είσοδος ένας πίνακας με n γραμμές και m στήλες, με 0/1 τιμές. Θέλετε να βρείτε όλα τα (r,c) -πλακίδια (*tiles*) από 1, δηλαδή συνδυασμούς από r γραμμές και c στήλες ώστε ο υπο-πίνακας με αυτές τις γραμμές και αυτές τις στήλες να έχει μόνο άσσους. Τα πλακίδια μπορεί να είναι επικαλυπτόμενα. Δώσετε ένα αποτελεσματικό αλγόριθμο για το πρόβλημα χρησιμοποιώντας την ιδέα του APriori.

Ερώτηση 2

Ο στόχος αυτής της άσκησης είναι να χρησιμοποιήσετε την εξόρυξη συχνών στοιχειοσυνόλων στην πράξη.

Για τα δεδομένα εισόδου, σας δίνεται το αρχείο “anonymized_grades_data.txt” το οποίο έχει (ανώνυμα) του βαθμούς φοιτητών για διαφορετικά μαθήματα. Το αρχείο αποτελείται από τρεις στήλες χωρισμένες με tab: η πρώτη στήλη είναι ένα μοναδικός αύξων αριθμός για κάθε φοιτητή, η δεύτερη στήλη είναι ο κωδικός του μαθήματος, και η τρίτη στήλη είναι ο βαθμός με τον οποίο έχει περάσει ο φοιτητής το μάθημα.

Επεξεργαστείτε αυτό το αρχείο ώστε να παράγετε ένα “καλάθι” για κάθε φοιτητή που περιέχει τα μαθήματα που έχει περάσει ο φοιτητής με βαθμό μεγαλύτερο ή ίσο με 0.7. Στην συνέχεια χρησιμοποιήστε ένα υπάρχον πακέτο για την εξόρυξη συχνών στοιχειοσυνόλων από τη σελίδα του FIMI, και βρείτε μαθήματα που εμφανίζονται συχνά μαζί. Λεπτομέρειες για το FIMI είναι στη σελίδα του μαθήματος. Θα πρέπει να μετατρέψετε τα δεδομένα στο format εισόδου που απαιτεί το πακέτο. Χρησιμοποιείτε ένα αρκετά υψηλό κατώφλι για την εύρεση συχνών στοιχειοσυνόλων, ώστε να μην πάρετε ένα πολύ μεγάλο αριθμό από στοιχειοσύνολα. Χρησιμοποιείτε το αρχείο “course_names.txt” που αντιστοιχεί κωδικούς μαθημάτων με το όνομα τους για να μετατρέψετε την έξοδο σε κάτι πιο εύκολο να διαβαστεί, επιθεωρήστε τα αποτελέσματα και σχολιάστε για στοιχειοσύνολα που βρήκατε.

Θα πρέπει να παραδώσετε τα ακόλουθα:

- Όλο τον κώδικα που θα γράψετε εσείς.
- Το αρχείο με τα συχνά στοιχειοσύνολα
- Μια σύντομη αναφορά όπου θα περιγράφετε πώς κάνατε την προεπεξεργασία, την επιλογή για το κατώφλι υποστήριξης, και ένα σχολιασμό για το τι βρήκατε.

Bonus: Υπολογίστε τα κλειστά και μεγιστικά στοιχειοσύνολα. Επίσης, πειραματιστείτε με διαφορετικά κατώφλια στο βαθμό του μαθήματος.

Ερώτηση 3

Στην Ερώτηση αυτή θα πειραματιστείτε με αλγορίθμους συστάσεων.

Χρησιμοποιώντας (όλα) τα δεδομένα από την Ερώτηση 2, αφαιρέστε τυχαία ένα 10% από τις εγγραφές του πίνακα. Ο στόχος είναι να υπολογίσετε αυτές τις τιμές χρησιμοποιώντας τις collaborative filtering τεχνικές που είδαμε στο μάθημα. Για να υπολογίσετε τον βαθμό για του φοιτητή S για το μάθημα C θα χρησιμοποιήσετε τον εξής αλγόριθμο. Χρησιμοποιώντας το υπόλοιπο 90% των δεδομένων θα βρείτε για τον φοιτητή S τους K πιο όμοιους άλλους φοιτητές ως προς το cosine similarity. Στην συνέχεια θα υπολογίσετε τον βαθμό του φοιτητή S για το μάθημα C ως τη μέση τιμή των βαθμών των K πιο όμοιων φοιτητών για το μάθημα C.

Θα υλοποιήσετε επίσης και δύο απλούς αλγορίθμους για σύγκριση. Ο πρώτος υπολογίζει τον βαθμό του φοιτητή S για το μάθημα C, ως τη μέση τιμή των βαθμών που παρατηρούμε για τον φοιτητή S (με την λογική ότι η ποιότητα του φοιτητή καθορίζει τον βαθμό του). Ο δεύτερος αλγόριθμος υπολογίζει τον βαθμό του S για το C ως την μέση τιμή των βαθμών που παρατηρούμε για το μάθημα C (με τη λογική ότι η δυσκολία του μαθήματος καθορίζει τον βαθμό).

Για την αξιολόγηση και σύγκριση των αλγορίθμων θα χρησιμοποιήσετε τη μετρική το Άθροισμα Τετραγώνων των Λαθών. Αν $g_1, g_2, \dots, g_n$ είναι οι βαθμοί που θέλουμε να προβλέψουμε, και $p_1, p_2, \dots, p_n$ είναι οι προβλέψεις του αλγορίθμου, το Άθροισμα Τετραγώνων των Λαθών του αλγορίθμου ορίζεται ως

$$AT\Lambda = \sum_{i=1}^n (g_i - p_i)^2$$

Θα πρέπει να παραδώσετε τα ακόλουθα:

- Όλο τον κώδικα που θα γράψετε εσείς.
- Το αρχείο με το 10% που προσπαθείτε να προβλέψετε και το υπόλοιπο 90%
- Μια αναφορά που θα περιγράφει τι κάνατε και θα σχολιάζει τα αποτελέσματα.

Bonus: Οι βαθμοί σας περιλαμβάνονται στα δεδομένα. Αν θέλετε μπορώ να σας πω σε ποια εγγραφή αντιστοιχούν και να προσθέσετε στην αναφορά ένα σχόλιο για τα αποτελέσματα στους βαθμούς σας.

Ερώτηση 4

Στην Ερώτηση αυτή θα υλοποιήσετε το Min-Hashing και το Locality Sensitive Hashing για την περίπτωση πραγματικών διανυσμάτων.

Θα χρησιμοποιήσετε τα δεδομένα που χρησιμοποιήσατε και για τις Ερωτήσεις 2 και 3. Μια μικρή τροποποίηση που θα κάνετε είναι να κανονικοποιήσετε τα διανύσματα αφαιρώντας τη μέση τιμή για κάθε γραμμή του πίνακα των δεδομένων.

Καταρχάς θα υπολογίσετε την ακριβή cosine similarity μεταξύ όλων των ζευγαριών από φοιτητές και θα τυπώσετε αυτούς που έχουν ομοιότητα τουλάχιστον 0.9. Στη συνέχεια θα παράγετε τυχαία διανύσματα με συνιστώσες τιμές -1 ή +1 (τυχαία επιλεγμένες) και θα υπολογίσετε την προβολή των διανυσμάτων των φοιτητών πάνω σε αυτά. Θα πειραματιστείτε με 50, 100, και 200 τυχαία διανύσματα. Το διάνυσμα των προβολών για κάθε φοιτητή ορίζει την «υπογραφή» του φοιτητή. Στη συνέχεια θα προσεγγίσετε την ομοιότητα μεταξύ δύο φοιτητών ως το cosine similarity των υπογραφών τους, και θα υπολογίσετε τα ζεύγη με προσεγγιστική ομοιότητα μεγαλύτερη του 0.9. Υπολογίστε τον αριθμό από false positives και false negatives. Αναφέρετε τη μέση τιμή από 5 διαφορετικά τρεξίματα.

Στη συνέχεια απλοποιήστε τις υπογραφές, μετατρέποντας τις σε διανύσματα (ή strings) με +1, -1 ανάλογα με το πρόσημο της προβολής. Μετά, σπάστε τον πίνακα με τα signatures σε b bands (μίνι-υπογραφές) με r προβολές ανά band, όπως περιγράψαμε στην τάξη, και υλοποιήστε το Locality Sensitive Hashing. Ο στόχος είναι να βρούμε υποψήφια ζεύγη με ομοιότητα τουλάχιστον 0.9. Χρησιμοποιήστε τον πίνακα με τα 100 προβολές, και πειραματιστείτε με r=5 και b=20, r=10 και b=10, r=20 και b=5. Αναφέρετε τον αριθμό των false positives και false negatives παίρνοντας την μέση τιμή σε 5 τρεξίματα.

Κάνετε turn in τον κώδικα σας, τα αποτελέσματα σας (μέσες τιμές) και μια αναφορά με τις παρατηρήσεις σας.

Τεχνικές Λεπτομέρειες

1. Για το pre-processing, οι παρακάτω unix εντολές μπορεί να σας φανούν χρήσιμες:
 - a. cut: επιτρέπει να πάρουμε συγκεκριμένες κolumns από ένα αρχείο με διαχωριζόμενες τιμές
 - b. sort: ταξινομεί τις γραμμές ενός αρχείου σε αλφαβητική σειρά. -n for αριθμητική σειρά
 - c. uniq: αφαιρεί συνεχόμενες γραμμές που είναι ίδιες.
2. Για την υλοποίηση του LSH δεν χρειάζεται να υλοποιήσετε δικό σας hash table ή list. Χρησιμοποιείτε υπάρχουσες υλοποιήσεις που έρχονται με τη γλώσσα που χρησιμοποιείτε (π.χ., στην C++ υπάρχει η STL υλοποίηση). Επίσης ίσως είναι βολικό όταν κάνετε hash τα mini-signatures να τα μετατρέψετε σε strings με '+' και '-' χαρακτήρες.
3. Αν και δεν είναι αποτελεσματικό, προς χάριν απλότητας μπορείτε να δημιουργήσετε τον πλήρη πίνακα φοιτητών και βαθμών. Θα πάρετε bonus βαθμούς για μια πιο αποτελεσματική υλοποίηση.