Δεύτερη Σειρά Ασκήσεων

Η προθεσμία για την δεύτερη σειρά ασκήσεων είναι στις 2 Μαΐου πριν το μάθημα. Κάνετε turn-in τον κώδικα σας, με οδηγίες πώς για το πώς τρέχει. Η αναφορά θα πρέπει να έχει λεπτομερείς παρατηρήσεις για τα αποτελέσματα σας. Για καθυστερημένες υποβολές ισχύει η πολιτική στην σελίδα του μαθήματος. Λεπτομέρειες για το turn-in στη σελίδα Ασκήσεις του μαθήματος. Θα γίνει προφορική εξέταση των ασκήσεων.

Ερώτηση 1 (Singular Value Decomposition)

Καταρχάς, να υπενθυμίσουμε ότι το eigenvalue decomposition ενός πραγματικού, τετράγωνου και συμμετρικού πίνακα B (μεγέθους $n \times n$) μας δίνει την παρακάτω έκφραση:

$$B = Q\Lambda Q^T$$

όπου $\Lambda = diag(\lambda_1, \dots, \lambda_n)$ είναι ένας διαγώνιος πίνακας που περιέχει τις ιδιοτιμές του B (οι οποίες είναι πάντα πραγματικές), και Q είναι ένας ορθοκανονικός πίνακας που περιέχει τα ιδιοδιανύσματα του B στις στήλες του.

Επίσης, το Singular Value Decomposition (SVD) ενός πραγματικού πίνακα M (μεγέθους $n \times d$) μπορεί να γραφτεί ως:

$$M = U\Sigma V^T$$

όπου $\Sigma = diag(\sigma_1, \dots, \sigma_k)$ είναι ένας $k \times k$ διαγώνιος πίνακας που περιέχει τα singular values του M (τα οποία είναι πάντα πραγματικά), U είναι ενας $n \times k$ ορθοκανονικός πίνακας που περιέχει τα left singular vectors του M στις στήλες του, και V είναι ένας $n \times k$ ορθοκανονικός πίνακας που περιέχει τα right singular vectors του M στις στήλες του.

1. Δείξτε ότι οι πίνακες $M^T M$ και MM^T είναι πραγματικοί, τετράγωνοι και συμμετρικοί.
2. Δείξτε ότι οι πίνακες $M^T M$ και MM^T έχουν τις ίδιες ιδιοτιμές. Δηλαδή, αν υπάρχουν λ, x έτσι ώστε $M^T Mx = \lambda x$ τότε υπάρχει y έτσι ώστε $MM^T y = \lambda y$. Είναι τα x και y τα ίδια?
3. Αν $M = U\Sigma V^T$ είναι το singular value decomposition του πίνακα M γράψετε μια απλοποιημένη έκφραση του πίνακα $M^T M$ χρησιμοποιώντας τους πίνακες V, V^T και Σ.
4. Σας δίνεται ο παρακάτω πίνακας M:

$$M = \begin{bmatrix} -1 & 1 & 0 \\ -7 & -8 & -6 \\ -4 & -1 & -2 \\ 5 & 5 & 4 \end{bmatrix}$$

   Υπολογίστε το SVD του M (χρησιμοποιήσετε την συνάρτηση *scipy.linalg.svd* της Python και θέσετε *full_matrices=False*). Κοιτάξετε τα singular values στο Σ. Τι παρατηρείτε? Μπορούμε να προσεγγίσουμε τον πίνακα M με ένα πίνακα μικρότερου βαθμού? Ποιο είναι το λάθος της προσέγγισης?

Ερώτηση 2 (Min-Hashing και LSH)

Σε αυτή την άσκηση θα υλοποιήσετε το Min-Hashing και το Locality Sensitive Hashing και θα τα εφαρμόσετε πάνω στα δεδομένα από τα tweets. Ο στόχος μας είναι να βρούμε πολύ όμοιους χρήστες. Υποψιαζόμαστε ότι αυτοί οι χρήστες μπορεί να είναι bots ή trolls μιας και θέλουμε να τους βγάλουμε από το σύστημα.

Θα χρησιμοποιήσετε και πάλι το αρχείο "tweets_dataset.txt" που χρησιμοποιήσατε στην προηγούμενη άσκηση. Κάνετε την ίδια επεξεργασία όπως πριν, αλλά αυτή τη φορά θα δημιουργήσετε μία εγγραφή για κάθε χρήστη όπου θα έχουμε το **σύνολο** των hashtags και handles που έχει χρησιμοποιήσει ο χρήστης. Για να κάνουμε τα δεδομένα μας πιο πυκνά, αφαιρέσετε τους χρήστες με λιγότερα από 20 hashtags/handles και τα hashtags/handles με λιγότερες από 20 εμφανίσεις. Εφαρμόστε αυτή τη διαδικασία κλαδέματος επαναληπτικά, μέχρι να μην μπορείτε να αφαιρέσετε κάποιο άλλο χρήστη ή hashtag/handle. Το αποτέλεσμα του κλαδέματος είναι το σύνολο των δεδομένων με το οποίο θα δουλέψουμε στη συνέχεια. (Υπόδειξη: Τα δεδομένα είναι μεγάλα οπότε είναι σημαντικό να χρησιμοποιήσετε αποδοτικές δομές για την διαδικασία του κλαδέματος).

Υπολογίσετε τα min-hash signatures για όλους τους χρήστες. Για την υλοποίηση σας θα πρέπει να μετατρέψετε τα strings των hashtags/handles σε 32-bit ακεραίους. Χρησιμοποιήστε την συνάρτηση hashString η οποία σας δίνεται στη σελίδα των Ασκήσεων. Για τα min-hashing functions χρησιμοποιήσετε hash functions της μορφής $h(x) = (a * x + b)\%R$, όπου τα a, b είναι τυχαίοι 32-bit ακέραιοι και το R είναι ένας πρώτος αριθμός μεγαλύτερος του $2^{32} - 1$ (μπορείτε να χρησιμοποιήσετε τον αριθμό $R = 4294967311$).

Για να αξιολογήσετε τα min-hash signatures που δημιουργήσατε, υπολογίσετε το πραγματικό Jaccard similarity μεταξύ των χρηστών. Στη συνέχεια υπολογίσετε την προσεγγιστική Jaccard similarity όπως το περιγράψαμε στην τάξη χρησιμοποιώντας 10, 50, 100, και 200 hash functions. Αξιολογήσετε την ακρίβεια της προσέγγισης χρησιμοποιώντας το sum of square errors (SSE) $\sum_{i,j}\left(J_{ij} - \overline{J_{ij}}\right)^2$, μεταξύ της πραγματικής ομοιότητας (J_{ij}) και της προσεγγιστικής ομοιότητας ($\overline{J_{ij}}$) για κάθε ζευγάρι χρηστών i, j. Επίσης βρείτε πόσα ζευγάρια από χρήστες έχουν Jaccard similarity μεγαλύτερο του 0.8 στα πραγματικά δεδομένα και βάσει των min-hash signatures. Υπολογίσετε τον αριθμό των false positives (ζευγάρια από χρήστες που στα min-hash signatures έχουν ομοιότητα μεγαλύτερη του 0.8 αλλά στην πραγματικότητα έχουν ομοιότητα μικρότερη του 0.8) και τον αριθμό των false negatives (ζευγάρια από χρήστες που στα min-hash signatures έχουν ομοιότητα μικρότερη του 0.8 αλλά στην πραγματικότητα έχουν ομοιότητα μεγαλύτερη του 0.8). Δημιουργήσετε γραφήματα που δείχνουν πως αυτές οι μετρικές αλλάζουν όταν αλλάζουμε τον αριθμό των min-hash functions.

Στη συνέχεια υλοποιήστε το Locality Sensitive Hashing, με b bands με r hash functions ανά band. Ο στόχος μας είναι να βρούμε τα ζευγάρια που έχουν ομοιότητα πάνω από 0.8. Χρησιμοποιήστε την φόρμουλα που μας δίνει την πιθανότητα να δημιουργήσουμε ένα υποψήφιο ζευγάρι ως συνάρτηση της ομοιότητας του ζευγαριού για να αποφασίσετε την τιμή των b, r (συμπεριλάβετε το γράφημα της συνάρτησης γι αυτές τις τιμές στην αναφορά σας). Δοκιμάσετε μερικές τιμές για τα b, r και αναφέρεται τον αριθμό των υποψήφιων ζευγαριών, και τον αριθμό των false positives και false negatives.

Τέλος για τις τιμές των b, r που θεωρείται καλύτερες κοιτάξετε τα υποψήφια ζευγάρια και τα tweets που έχουν κάνει. Πιστεύετε ότι κάποια από αυτά είναι bots ή trolls? Δικαιολογήστε την απάντηση σας

Ερώτηση 3 (Συστήματα συστάσεων)

Για την ερώτηση αυτή θα προσπαθήσουμε να προβλέψουμε και να συστήσουμε σε χρήστες του Twitter hashtags ή handles τα οποία θα τους ενδιαφέρουν.

Χρησιμοποιήσετε τα «κλαδεμένα» δεδομένα της προηγούμενης ερώτησης. Διαλέξετε τυχαία 10% των χρηστών και αφαιρέστε τυχαία ένα από τα hashtags/handles από το σύνολο τους. Ο στόχος μας είναι να βρούμε αυτό το hashtag/handle το οποίο αφαιρέσαμε.

Θα εξετάσετε τους διαφορετικούς αλγορίθμους γι αυτό το σκοπό. Οι αλγόριθμοι μας δίνουν ένα σκορ σε κάθε hashtag/handle και θα προτείνουμε στον χρήστη τα k hashtags/handles με το μεγαλύτερο σκορ.

1. **MostPopular**: Για τα hashtags/handles τα οποία δεν είναι στο σύνολο του χρήστη υπολόγισε το σκορ τους ως τον αριθμό των χρηστών που τα χρησιμοποιούν.
2. **User-based Collaborative Filtering** (**UCF**): Για ένα χρήστη u βρες τους m πιο όμοιους χρήστες ($N_m(u)$) χρησιμοποιώντας το Jaccard similarity, και υπολόγισε το σκορ ενός hashtag/handle h ως $\sum_{v \in N_m(u)} JSim(u,v) I(v,h)$, όπου $JSim(u,v)$είναι το Jaccard similarity μεταξύ u, v και $I(u,h)$ είναι 1 αν ο χρήστης u έχει το hashtag/handle h στο σύνολο του, και 0 διαφορετικά
3. **Item-based Collaborative Filtering** (**ICF**): Για ένα χρήστη u, και ένα υποψήφιο hashtag/handle h βρες τα m πιο όμοια hashtag/handles που είναι στο σύνολο του u ($N_m(h)$) χρησιμοποιώντας το Jaccard similarity, και υπολόγισε το σκορ του h ως $\sum_{h' \in N_m(u)} JSim(h,h')$, όπου $JSim(h,h')$είναι το Jaccard similarity μεταξύ των hashtags/handles h και h'.
4. **Singular Value Decomposition** (**SVD**). Υπολογίστε το Singular Value Decomposition του 0/1 user-hashtag/handle πίνακα M.Πάρετε ένα m-rank approximation M_rτου πίνακα M, και υπολογίσετε το σκορ του hashtag/handle h για τον χρήστη u ως $M_r(u,h)$.

Χρησιμοποιώντας οποιοδήποτε από τους παραπάνω αλγορίθμους, για κάποιο k, μπορούμε να βρούμε τα k hashtags/handles με το μεγαλύτερο σκορ. Θα αξιολογήσουμε τους αλγόριθμους υπολογίζοντας την ακρίβεια της πρόβλεψης. Θεωρούμε ότι μια πρόβλεψη είναι επιτυχής αν το hashtag/handle που θέλουμε να βρούμε είναι μέσα τα k προτεινόμενα hashtags/handles του αλγορίθμου. Η ακρίβεια της πρόβλεψης ενός αλγορίθμου είναι το ποσοστό των χρηστών για τους οποίους η πρόβλεψη ήταν επιτυχής.

Δημιουργήστε μια γραφική παράσταση με την ακρίβεια των αλγορίθμων για τιμές του k από 1 εώς 20. Για να θέσετε την τιμή του m για τους αλγορίθμους **UCF** και **ICF** φιξάρετε το $k = 10$ και κάνετε μια γραφική παράσταση της ακρίβειας για διαφορετικά m από 10 εώς 100 σε βήματα των 5. Για τον αλγόριθμο **SVD** κάνετε την γραφική παράσταση των singular values και θέσετε το m προσδιορίζοντας το σημείο που κάνει «γόνατο» η γραφική παράσταση. Συμπεριλάβετε όλες τις γραφικές παραστάσεις στην αναφορά σας. Σχολιάστε τα αποτελέσματα.

Bonus: Για κάθε ζευγάρι (u,h) χρήστη και hashtag/handle έχουμε τον αριθμό των φορών που ο χρήστης χρησιμοποίησε αυτό το hashtag/handle. Θεωρήστε ότι αυτό είναι το rating του χρήστη για το hashtag/handle και τροποποιήστε τους παραπάνω αλγορίθμους ώστε να χρησιμοποιούν το cosine similarity αντί για το Jaccard similarity, όπως τους περιγράψαμε στην τάξη. Συγκρίνετε με την υλοποίηση που χρησιμοποιεί 0/1.