Τμήμα Μηχανικών Η/Υ και Πληροφορικής Πανεπιστημίου Ιωαννίνων
Μεταπτυχιακό Μάθημα: *Αλγόριθμοι Επιστήμης Δεδομένων*
Ακαδημαϊκό Έτος 2019-20

Λουκάς Γεωργιάδης & Σπύρος Κοντογιάννης

1ο Σετ Ασκήσεων (2ο μέρος)

Ημερομηνία παράδοσης: Τρίτη, 14/01/2020

Άσκηση 2 – Αλγόριθμοι για Ροές Δεδμένων

Η παρούσα άσκηση αποσκοπεί στην πρόσβαση σε πραγματικές ροές δεδομένων, και την εφαρμογή αλγορίθμων που κάνουν στατιστική ανάλυση σε αυτές. Ως παράδειγμα ροής θα χρησιμοποιηθεί το Twitter. Ζητούνται τα εξής:

(2α) Παρακολούθηση του Twitter STREAM: Υλοποίηση ενός προγράμματος παρακολούθησης (listener) του Twitter Stream σε πραγματικό χρόνο. Για το πρόγραμμα αυτό μπορείτε να αξιοποιήσετε τη βιβλιοθήκη **tweepy** (https://www.tweepy.org/). Δείτε επίσης το εξής tutorial: https://www.youtube.com/watch?v=wlnx-7cm4Gg. Ως είσοδος του προγράμματός σας θα δίνεται ο χρόνος παρακολούθησης του Twitter Stream (ακέραιος αριθμός, σε δευτερόλεπτα, πχ, ο αριθμός 86400 εφόσον επιθυμούμε να γίνει παρακολούθηση για 24 ώρες, ή ο αριθμός 300 εφόσον θέλουμε να γίνει παρακολούθηση για 5 λεπτά) από τη στιγμή εκκίνησης του προγράμματος. Η έξοδος (δηλαδή, η ροή των tweets για το συγκεκριμένο διάστημα) θα είναι σε **JSON μορφή** και είτε θα αποθηκεύεται σε CSV αρχείο της επιλογής του χρήστη, ή θα αποτυπώνεται στην οθόνη (όπως ακριβώς γίνεται και στο tutorial), ανάλογα με την επιλογή που θα κάνει ο χρήστης του προγράμματος. Προτείνεται να ακολουθήσετε πιστά τις οδηγίες του tutorial.

(2β) Καταγραφή Ιστοριών στο Twitter: Μέσα στο Twitter εμφανίζονται διαφορετικές εγγραφές που αφορούν είτε αρχικά Tweets (με δικό τους Tweet-ID, και χρήστη-συγγραφέα), ή κάποιου είδους αντιδράσεις (πχ, comments, re-tweets, likes/dislikes, κ.λπ.) που επίσης έχουν το δικό τους Tweet-ID, έχουν όμως και το Tweet-ID του αρχικού Tweet. Έτσι, εντός της ροής υπάρχει μια ολόκληρη αλυσίδα από εγγραφές (original tweet και αντιδράσεις σε αυτό) που απαρτίζουν μια τη **ιστορία** (Tweet-Story). Επιθυμούμε να μετρήσουμε κάποια στατιστικά στοιχεία ως προς αυτές τις ιστορίες. Για τον λόγο αυτό, θα κατασκευάσετε μια νέα ροή, χρησιμοποιώντας ως είσοδο ένα αρχείο σε JSON μορφή με απόσπασμα του Twitter-STREAM, όπως αυτό που παράγεται στο ερώτημα **2α**. Το όνομα του αρχείου εισόδου θα δίνεται ως παράμετρος στο πρόγραμμά σας. Για κάθε εγγραφή στο αρχείο εισόδου (αφορά ένα καινούργιο tweet, ή κάποια αντίδραση σε υπάρχον tweet), θα εντοπίζεται το πεδίο που περιγράφει το Tweet-ID του που αντιστοιχεί στο αρχικό tweet, κι όχι απαραίτητα το συγκεκριμένο μήνυμα (υπάρχουν και τα δύο ως πληροφορία μέσα στην εγγραφή, όταν πρόκειται για αντίδραση). Αυτό το Tweet-ID θα χρησιμοποιείται ως το κλειδί (key), ενώ η τιμή (value) θα είναι ο αύξων αριθμός της συγκεκριμένης εγγραφής μέσα στη ροή. Η νέα ροή που δημιουργείτε θα απαρτίζεται από ακριβώς αυτά τα key-value ζεύγη, και τελικά θα αποθηκεύεται σε συγκεκριμένο αρχείο εξόδου (μορφής CSV), για μελλοντική χρήση αλλά και για επαλήθευση, χρησιμοποιώντας μια γραμμή ανά ζεύγος και χωρίζοντας με κόμμα το κλειδί (Originating Twitter-Story User ID) από την τιμή (αύξων αριθμός εγγραφής μέσα στη ροή). Διαφορετικά ζεύγη κλειδιού-τιμής θα αποθηκεύονται σε διαφορετικές γραμμές.

(2γ) Στατιστική Επεξεργασία Ιστοριών στο Twitter: Υλοποίηση των αλγορίθμων **FM** για την εκτίμηση του πλήθους των ξεχωριστών αντικειμένων (distinct elements), και **AMS** για την εκτίμηση της k-στής στιγμής ($k \geq 2$), αντίστοιχα, σε μια ροή δεδομένων. Ως είσοδος θα δίνεται το όνομα ενός CSV αρχείου, το οποίο θα πρέπει να είναι στη μορφή που ορίζει το ερώτημα **2β**, η τιμή του k (για k=0 θα χρησιμοποιείται ο αλγόριθμος **FM**, για $k \geq 2$ θα χρησιμοποιείται ο **AMS**), καθώς και το πλήθος N των αρχικών εγγραφών (από την 1η μέχρι και τη N-στή) που θα ληφθούν υπόψη για τη μέτρηση. Θα πρέπει να τυπώνεται η τρέχουσα τιμή της εκτίμησης, σε κάθε βήμα (δηλαδή, για κάθε καινούργια εγγραφή που εμφανίζεται στη ροή), μέχρι και το N-στό βήμα. Ειδικά για τον **AMS** αλγόριθμο, σημειώνεται ότι δε θα πρέπει να θεωρείται γνωστή εκ των προτέρων η τιμή του N, απλά μετά και τη N-στή εγγραφή θα τερματίζεται ο

αλγόριθμος. Θυμηθείτε λοιπόν ότι, για να γίνεται σωστά η δειγματοληψία σε κάθε βήμα, θα χρειαστεί να χρησιμοποιήσετε τον αλγόριθμο **ReservoirSampling**.

Τέλος, και για τους δυο αλγορίθμους θα πρέπει να χρησιμοποιηθούν a*b μεταβλητές (πχ, R[1],R[2], …,R[a*b] για τον FM, και X[1],X[2],…,X[a*b] για τον AMS), οι οποίες θα χωρίζονται σε a ομάδες των b μεταβλητών η καθεμιά, και ως απάντηση θα δίνεται ο μέσος όρος (average) των διάμεσων (medians) όλων των ομάδων.

Για παραγωγή τυχαίων συναρτήσεων κατακερματισμού, μπορείτε να χρησιμοποιήσετε (όπως και στην 1η άσκηση) μια καθολική οικογένεια $\mathcal{H}$ συναρτήσεων κατακερματισμού $h_{a,b}: \{0, \dots, m-1\} \to \{0, \dots, m-1\}$. Μια τέτοια οικογένεια υλοποιεί το αρχείο **universalHashFunctions.py**.

ΠΑΡΑΔΟΣΗ ΕΡΓΑΣΙΑΣ:

Θα πρέπει να αναρτήσετε στο eCourse, το αργότερα μέχρι την Τρίτη 14/1/2020, ένα ZIP αρχείο με όνομα της μορφής:

2019-20_CSE-UOI_A2-ADS_< EPWNYMO >-< ONOMA >_< AM >_ASSIGNMENT-1B.ZIP

το οποίο περιλαμβάνει τα εξής:

(ι) Φάκελο **SOURCES** με όλα τα προγράμματά σας σε Python.

(ιι) Φάκελο **DATA** με το αρχείο εξόδου που παράγεται από το **2α** για συγκεκριμένο παράδειγμα εκτέλεσης (πχ, για παρακολούθηση του Twitter API για 5 ώρες), και το αρχείο εξόδου που παράγεται από το **2β** (η ροή με τα key-value pairs).

(ιν) Σύντομη αναφορά (σε μορφή MS WORD ή LaTeX), η οποία να περιγράφει την υλοποίησή σας, και να παρουσιάζει συνοπτικά τα αποτελέσματα (και συνοπτική ερμηνεία τους) του παραδείγματος εκτέλεσης των προγραμμάτων. Συγκεκριμένα, θα αναγράφεται για το παράδειγμα εκτέλεσης το μήκος της ροής, το πραγματικό πλήθος των διαφορετικών ιστοριών, το πραγματικό διάνυσμα με τις συχνότητες εμφάνισης των ιστοριών, και οι εκτιμήσεις που παρέχουν υλοποιήσεις σας για τις διαφορετικές ιστορίες (k=0) και για τον αριθμό-έκπληξη της ροής (k=2).