Υπογραμμικοί Αλγόριθμοι
Σημειώσεις 11ης διάλεξης

Επιμέλεια διαφανειών - Ευάγγελος Πρωτόπαπας

7/1/2020

Property Testing (Έλεγχος Ιδιότητας)

Στο μάθημα είδαμε διάφορες εφαρμογές σχετικές με τον έλεγχο ιδιότητας. Η γενική ιδέα είναι ότι έχοντας πρόσβαση σε ένα μαντείο για ένα στιγμιότυπο ενός προβλήματος, θέλουμε να απαντήσουμε προσεγγιστικά αν το στιγμιότυπο έχει μια συγκεκριμένη ιδιότητα χωρίς να το δούμε ολόκληρο, το οποίο απαιτεί σε κάποιες περιπτώσεις γραμμικό χρόνο, ο οποίος ενδέχεται να είναι απαγορευτικός.

Η ιδέα είναι οτι υποδειγματοληπτούμε με κάποιο κατάλληλο τρόπο, ώστε να μπορούμε να απαντήσουμε στο εξής ερώτημα:

Έχει το στιγμιότυπο την ζητούμενη ιδιότητα ή είναι πολύ μακρυά από αυτή;

Συνήθως υπάρχει μία ενδιάμεση περιοχή μεταξύ των δύο καταστάσεων στην οποία δεν μπορούμε να δώσουμε σαφή απάντηση. Συνήθως η ύπαρξη αυτής της ενδιάμεσης περιοχής είναι αναγκαία, καθώς διαφορετικά θα πρέπει να διαβάσουμε όλη την πληροφορία. Είναι εύκολο να δει κανείς ότι ένας υπογραμμικός αλγόριθμος δεν μπορεί να αποφύγει την ύπαρξη αυτής της περιοχής υποδειγματοληπτώντας την πληροφορία του μαντείου.

Παρακάτω θα δούμε τέσσερα προβλήματα σε αυτό το πλαίσιο.

Έλεγχος μονοτονικότητας

Μας δίνεται πρόσβαση σε έναν πίνακα n στοιχείων $A[\,]$ ως μαντείο.

Ερώτηση: Θέλουμε να διακρίνουμε ανάμεσα στις περιπτώσεις:

(I) $\forall i\ A[i] < A[i+1]$ (δηλαδή ο A είναι ταξινομημένος (σε αύξουσα σειρά)).

(II) Ο A είναι "πολύ αταξινόμητος".

Ορισμός. *Αποκαλούμε τον πίνακα A "**πολύ αταξινόμητο**" αν πρέπει να αλλάξουμε τουλάχιστον $\delta \cdot n$ στοιχεία του για να γίνει ταξινομημενος (σε αύξουσα σειρά).*

Παράδειγμα: Ο $A = [1\ 2\ 3\ 2]$ είναι $\delta = 1/4$ αταξινόμητος.

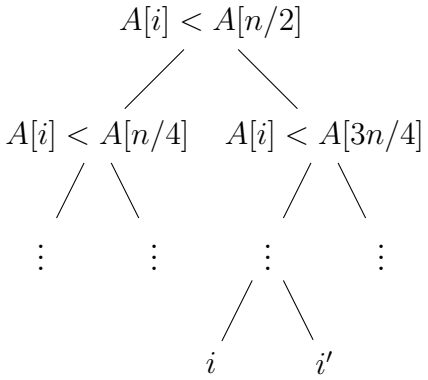
Ο παρακάτω αλγόριθμος μας επιτρέπει να λύσουμε το παραπάνω πρόβλημα.

Algorithm 1: Αλγόριθμος

Επέλεξε τυχαίο $i \in [n]$.
Κάνε δυαδική αναζήτηση στον A για το i.
if Αν κατέληξες στο i **then**
  Επέστρεψε (I).
else
  Επέστρεψε (II).
end if

Παρατήρηση. Η τυχαιότητα στον παραπάνω αλγόριθμο υπάρχει μόνο στη επιλογή του i. Μόλις φιξάρω το i είναι όλα ντετερμινιστικά.

Η εκτέλεση του αλγόριθμου με κάποιο $i \in [n]$ μας δίνει το εξής δέντρο:



Ένα i είναι κακό στην περίπτωση (II), αν η αναζήτηση στο δέντρο καταλήξει στο i.

Έστω i, i' στο παραπάνω δέντρο κακά. Ο χαμηλότερος κοινός πρόγονος των i, i' μας υποδηλώνει ότι $A[i] < A[i']$ αν i αριστερά του i' στο δέντρο. Έπεται ότι αν συγκρίνω όλα τα κακά $i_1 < i_2 < \cdots < i_k$ θα ισχύει ότι $A[i_1] < A[i_2] < \cdots < A[i_k]$.

Συνεπώς στη περίπτωση (II) όπου ο A είναι δ αταξινόμητος έπεται ότι $k \leq n(1-\delta)$.

Άρα ο αλγόριθμος επίλεγει κακό i με πιθανότητα $k/n \leq 1-\delta$. Άρα η πιθανότητα να μην ξεγελαστεί είναι $\geq \delta$.

Αν επαναλάβουμε τον αλγόριθμο c/δ φορές όπου c σταθερά, με πιθανότητα $(1-\delta)^{c/\delta} \approx (1/e)^c \approx 0.01$ θα ξεγελιέται συνέχεια.

Τελικά αν επανάλουμε τον αλγόριθμο $O(1/\delta)$ φορές πετυχαίνουμε με πολύ μεγάλη πιθανότητα χρησιμοποιώντας $O(\log n/\delta)$ δείγματα από τον A.

Παρατήρηση. Αν το δ γίνει πολύ μικρό ουσιαστικά ο αλγόριθμος γίνεται χειρότερος από τον τετριμμένο γραμμικό.

Σταθμοί ΔΕΗ

Έχουμε ένα δίκτυο με n σταθμούς οι οποίοι βρίσκονται όλοι σε λειτουργία. Κάποιες φορές κάποιος σταθμός μπορεί να χαλάσει και ταυτόχρονα να ρίξει και πολλούς γειτονικούς σταθμούς σε ένα συνεχομενο υποδιάστημα του $[n]$. Αποδίδουμε τιμή 1 στους ενεργούς σταθμούς και 0 σε αυτούς που έχουν πέσει. Θέλω ανα πάσα στιγμή να έχω μια εκτίμηση για το πόσοι σταθμοί έχουν πέσει. Το δίκτυο ουσιαστικά σχηματίζει μια συμβολοσειρά μήκους n.

n υποσταθμοί, πλήθος μηδενικών

Στο παρόν πρόβλημα θέλουμε χρησιμοποιώντας n/k δείγματα να απαντήσουμε στην παρακάτω ερώτηση.

Ερώτηση. Είναι τουλάχιστον $5k$ σταθμοί απο αυτούς κάτω ή είναι το πολύ k απο αυτούς κάτω; (ισοδύναμα έχουμε $\geq 5k$ ή $\leq k$ μηδενικά στη συμβολοσειρά).

Η ιδέα είναι η εξής. Δειγματοληπτούμε την συμβολοσειρά με ρυθμό $\Theta(1/k)$. Ορίζουμε τυχαία μεταβλητή:

$$Y = \sum_{\substack{i \\ \text{υποσταθμοί}}} Y_i$$

όπου Y_i τυχαίες μεταβλητές με $Y_i = 1$ αν δειγματοληπτήσουμε τον i-οστό σταθμό και αυτός έχει πέσει.

Η αναμενόμενη τιμή της Y στην περίπτωση που έχουμε $\geq 5k$ σταθμούς κάτω είναι ≥ 5, ενώ στην περίπτωση που έχουμε $\leq k$ σταθμούς κάτω είναι ≤ 1. Έπειτα χρησιμοποιούμε Chernoff Bound για να φράξουμε την πιθανότητα η Y να απομακρυνθει πολύ απο την $\mathbb{E}[Y]$ ώστε να εγγυηθούμε με πιθανότητα $2/3$ ότι μπορούμε να διαχωρίσουμε τις δύο περιπτώσεις.

Πλήθος συνεχόμενων μηδενικών υποσυμβολοσειρών

Θα εξετάσουμε τώρα ένα διαφορετικό πρόβλημα στο οποίο μας ενδιαφέρει να μετρήσουμε το πλήθος των συνεχόμενων μηδενικών υποσυμβολοσειρών. Έστω μια συμβολοσειρά που αναπαριστά το δίκτυο:

$$1\,1\,1\,1\,1\,1\,\mathbf{1}\,\mathbf{0}\,0\,0\,1\ldots1\,\mathbf{1}\,\mathbf{0}\,0\,0\,0\ldots1\,1$$

Θέλουμε να μετρήσουμε το πλήθος των εναλλαγών απο 1 σε 0. Μας ενδιαφέρει λοιπόν μόνο το μοτίβο $\ldots1\ 0\ldots$. Έστω r το πλήθος των συνεχόμενων μηδενικών υποσυμβολοσειρών. Αποκαλούμε κάθε r ένα "τρέξιμο". Θέλουμε όπως και παραπάνω με n/k δείγματα να απαντήσουμε στην ερώτηση:

Ερώτηση. Υπάρχουν τουλάχιστον $5k$ τρεξίματα ή το πολύ k τρεξίματα;

Η ιδεά είναι η εξής. Δειγματοληπτώ με ρυθμό $1/k$. Αναμένουμε νε πιάσουμε r/k τέτοια μοτίβα. Ο αλγόριθμος μέτρησης είναι ο εξής:

Algorithm 2: Αλγόριθμος

Αρχικοποίησε $cnt \leftarrow 0$.
for $i = 2$ εως n **do**
  Με πιθανότητα $\frac{1}{k}$ κοίτα τα $A[i], A[i-1]$.
  if $A[i] == 0$ και $A[i-1] == 1$ **then**
    Θέσε $cnt \leftarrow cnt + 1$.
  end if
end for
Επέστρεψε cnt.

Συνεπώς μπορούμε να διαχωρίσουμε τις δύο περιπτώσεις όπως και στο προηγούμενο πρόβλημα. Η ανάλυση γίνεται με παρόμοιο τρόπο με Chernoff Bound.

Έλεγχος συνεκτικότητας σε γραφήματα

Συνεκτικότητα γραφήματος

Μας δίνεται ένα γράφημα G με βαθμό d κας μας δίνεται πρόσβαση σε ερωτήσεις της μορφής $Query(u, i)$, όπου u κορυφή του γραφήματος και $i \in [d]$, όπου $Query(u, i) = v$ αν ο i-οστός γείτονας του u είναι ο v.

Θέλουμε να δώσουμε απάντηση στην εξής ερώτηση. *Είναι το γράφημα συνεκτικό;*

Τετριμμένα γνωρίζουμε ότι με BFS ή DFS μπορούμε να δώσουμε την απάντηση σε χρόνο $O(n+m)$. Θα θέλαμε να κάνουμε κάτι πολύ καλύτερο απαντώντας προσεγγιστικά στο παρακάτω ερώτημα.

Ερώτηση. Μπορώ να διακρίνω σε $o(n)$ τις παρακάτω περιπτώσεις;

(I) G συνεκτικό.

(II) G "πολύ μη-συνεκτικό".

Ορισμός. *Αποκαλούμε το γράφημα G "**πολύ μη-συνεκτικό**" αν πρέπει να αλλάξουμε (προσθέτουμε, αφαιρούμε) τουλάχιστον $\zeta \cdot n$ ακμές για να γίνει συνεκτικό, αυξάνοντας το βαθμό του το πολύ κατά 1.*

Πως μοιάζει ένα γράφημα G το οποίο είναι πολύ μη-συνεκτικό;

Μπορεί το γράφημα να έχει λιγότερες απο $\zeta \cdot n + 1$ συνεκτικές συνιστώσες αν επιτρέψουμε την αύξηση του βαθμού κατά 1; Η απάντηση είναι οχι. Αρχικά ας παρατηρήσουμε ότι, αν έχουμε ≥ 2 συνεκτικές συνιστώσες και το $d \geq 2$, μπορούμε να συνδέσουμε τις συνεκτικές συνιστώσες σε ένα "μονοπάτι" ώστε το γράφημα να γίνει συνεκτικό, αυξάνοντας τον βαθμό σε το πολύ $d + 1$. καθώς θα μπορούσα να συνδέσω τις συνεκτικές συνιστώσες σε ένα "μονοπάτι" χρησιμοποιώντας $\zeta \cdot n - 1$ ακμές ώστε το γράφημα να γίνει συνεκτικό.

Έχει το γράφημα τουλάχιστον $(\zeta \cdot n)/2$ συνεκτικές συνιστώσες οι οποίες σέβονται το βαθμό;. Η απάντηση είναι ναι. Έστω ότι έχουμε λιγότερες από $(\zeta \cdot n)/2$ συνεκτικές συνιστώσες. Συνδέουμε όπως και παραπάνω τις συνεκτικές συνιστώσες, με τη διαφορά ότι συνδέουμε την συνεκτική συνιστώσα με 2 άλλες χρησιμοποιώντας 2 κορυφές που δεν ενώνονται με ακμή. Αν δεν υπάρχουν 2 τέτοιες κορυφές τότε η συνεκτική συνιστώσα είναι κλίκα όπου αφαιρώ μια ακμή και ενώνω τις 2 συνεκτικές συνιστώσες στις προσπίπτουσες κορυφές της ακμής που αφαίρεσα.

Ο παρακάτω αλγόριθμος μας επιτρέπει να λύσουμε το παραπάνω πρόβλημα.

Algorithm 3: Αλγόριθμος

for $O(\frac{1}{\zeta \cdot d})$ φορές **do**
  Επέλεξε τυχαία κορυφή u.
  Κάνε BFS απο την κορυφή u.
  Σταμάτα μετά απο $50 \cdot \frac{1}{\zeta \cdot d}$ κορυφές.
  if δεν βρήκες τόσες κορυφές **then**
    Επέστρεψε (II).
  end if
end for
Επέστρεψε (I).

Παρατήρηση. Αν το γράφημα είναι συνεκτικό επιστρέφω πάντα (I). Αν το γράφημα ικανοποιεί την (II), ο αλγόριθμος θα πετύχει αν πέσει σε μία συνεκτική συνιστώσα που είναι μικρή.

Συνεπώς πρέπει να δείξουμε ότι υπάρχουν πολλές συνιστώσες που είναι μικρές. Αρχικά δεν υπάρχουν πάνω απο $(\zeta \cdot n)/4$ συνεκτικές συνιστώσες με την κάθε μία να έχει $\geq 5/\zeta$ κορυφές. Διαφορετικά θα έχουμε $> n$ κορυφές στο γράφημα. Άρα έχουμε $(\zeta \cdot n)/2 - (\zeta \cdot n)/4 = (\zeta \cdot n)/4$ συνεκτικές συνιστώσες με την κάθε μία να έχει $< 5/\zeta$ κορυφές.

Συνεπώς έχουμε πολλές συνεκτικές συνιστώσες ($(\zeta \cdot n)/4$ συνεκτικές συνιστώσες), οι οποίες είναι μικρές ($5/\zeta$ κορυφές).

Ποιά είναι η πιθανότητα να επιλέξουμε κορυφή σε μικρή συνιστώσα; Αναμένουμε ότι μετά από $O(1/\zeta)$ τυχαίες κορυφές θα πέσουμε σε μια μικρή συνιστώσα. Η ιδεά είναι ότι πρέπει να πέσουμε σε 1 απο τις $(\zeta \cdot n)/4$ μικρές συνιστώσες. Η πιθανότητα να πέσουμε σε κάποια συνιστώσα είναι τουλάχιστον $1/n$, συνεπώς εφόσον έχουμε $(\zeta \cdot n)/4$ μικρές συνιστώσες, η πιθανότητα να πέσουμε σε μία απο αυτές είναι τουλάχιστον $\zeta/4$.

Παρατήρηση. Η γενική ιδέα είναι ότι πρέπει να κόψουμε το BFS νωρίς, αλλά σε σημείο που μας επιτρέπει να διακρίνουμε την περίπτωση (II). Με βάση τα παραπάνω προκύπτει ότι το $O(1/\zeta)$ αρκεί.

Πλήθος Συνεκτικών Συνιστωσών

Βρισκόματε πάλι στο ίδιο setting με ένα γράφημα G με βαθμό $\leq d$.

Θέλουμε να δώσουμε απάντηση στο εξής ερώτημα.

Ερώτηση. Πόσες είναι οι συνεκτικές συνιστώσες του G; Πόσο χρόνο θέλεις να μετρήσεις πόσες είναι;

Έστω $\Sigma\Sigma$ = πλήθος συνεκτικών συνιστωσών του G. Θέλουμε να βρούμε μία προσέγγιση $\tilde{\Sigma\Sigma}$ του $\Sigma\Sigma$ τέτοια ώστε:

$$|\tilde{\Sigma\Sigma} - \Sigma\Sigma| \leq \epsilon \cdot n \quad \text{για κάποιο } \epsilon$$

Παρατήρηση. Το σφάλμα είναι αθροιστικό. Μπορεί να αποδειχθεί ότι δεν μπορούμε να πάρουμε πολλαπλασιαστικό σφάλμα χωρίς να δούμε όλο το γράφημα.

Η γενική ιδεά είναι η εξής.

Έστω u κορυφή του G. Ορίζουμε ως n_u το μέγεθος της συνεκτικής συνιστώσας στην οποία ανήκει ο u. Παρατηρούμε ότι $\sum_{u \in V}(1/n_u) = \Sigma\Sigma$.

Θα προσεγγίσουμε τις ποσότητες $1/n_u$. Επιλέγουμε μία τυχαία κορυφή u. Αν η συνεκτική συνιστώσα στην οποία ανήκει ο u είναι πολύ μικρή θα την πιάσουμε όλη με BFS.

Ο αλγόριθμος που θα ορίσουμε θα προσεγγίζει έμμεσα τις ποσότητες $1/n_u$, προσεγγίζοντας τις ποσότητες $\hat{n_u} = \min\{n_u, 2/\epsilon\}$.

Παρατηρούμε ότι για κάθε κορυφή u ισχύει ότι $|1/\hat{n_u} - 1/n_u| \leq \epsilon/2$. Διαισθητικά όσο το n_u αυξάνεται έχοντας φράξει το $\hat{n_u} \leq 2/\epsilon$ η απόσταση των δύο ποσοτήτων δεν μπορεί να ξεφύγει πάνω απο $\epsilon/2$.

Ορίζουμε ως $\hat{\Sigma\Sigma} = \sum_{u \in V} 1/\hat{n_u}$.

Μπορούμε να δείξουμε ότι $|\hat{\Sigma\Sigma}| \leq (\epsilon \cdot n)/2$. Έχουμε n όρους, αν τους γκρουπάρουμε ανα δύο όπως παραπάνω και πάρουμε την τριγωνική ανισότητα βγάινει.

Με βάση όλα τα παραπάνω αρκεί να βρούμε αλγόριθμο που να προσεγγίζει την ποσότητα $\hat{\Sigma\Sigma}$, δηλαδή θέλουμε να ισχύει ότι:

$$|\tilde{\Sigma\Sigma} - \hat{\Sigma\Sigma}| \leq \frac{\epsilon \cdot n}{2}$$

Ο παρακάτω αλγόριθμος μας δίνει αυτή την προσέγγιση.

Algorithm 4: Αλγόριθμος

Θέσε $ans \leftarrow 0$
for $\Theta(1/\epsilon)$ φορές **do**
  Επέλεξε τυχαία κορυφή u.
  Τρέξε BFS απο την κορυφή u και σταμάτα μετά απο $2/\epsilon$ κορυφές.
  Θέσε $ans \leftarrow ans + 1/$πλήθος κορυφών που βρέθηκαν.
end for
Θέσε $\tilde{\Sigma\Sigma} \leftarrow n \cdot ans/s$.
Επέστρεψε $\tilde{\Sigma\Sigma}$.

Παρατήρηση. Διαισθητικά ο λόγος που προσεγγίζουμε τις ποσότητες $1/\hat{n_u}$ αντί των $1/n_u$ οφείλεται στο ότι δεν μπορούμε να πάρουμε καλές προσεγγίσεις για τις μεγάλες συνεκτικές συνιστώσες. Φράσσοντας τις ποσότητες $1/\hat{n_u}$ απο $2/\epsilon$ και προσεγγίζοντας αυτές τις ποσότητες μας επιτρέπει να χειριστούμε τη συγκέντρωση των μεγάλων συνιστωσών μέσω φραγμάτων Chernoff.