Κανόνες Συσχέτισης ΙΙ

Οι διαφάνειες στηρίζονται στο **P.-N. Tan, M.Steinbach, V. Kumar, «Introduction to Data Mining»**, Addison Wesley, 2006

Σύντομη Ανακεφαλαίωση

Εισαγωγή

Market-Basket transactions (Το καλάθι της νοικοκυράς!)

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

δοσοληψία

- Προώθηση προϊόντων
- Τοποθέτηση προϊόντων στα ράφια
- Διαχείριση αποθεμάτων

Το πρόβλημα: Δεδομένου ενός συνόλου δοσοληψιών (transactions), βρες κανόνες που προβλέπουν την εμφάνιση στοιχείων (item) με βάση την εμφάνιση άλλων στοιχείων στις συναλλαγές

Παραδείγματα κανόνων συσχέτισης

{Diaper} → {Beer},
{Milk, Bread} → {Eggs,Coke},
{Beer, Bread} → {Milk}

Σημαίνει ότι εμφανίζονται μαζί, όχι ότι η εμφάνιση του ενός είναι η αιτία της εμφάνισης του άλλου (co-occurrence, not causality όχι έννοια χρόνου ή διάταξης)

Ορισμοί

στοιχειοσύνολο (itemset): Ένα υποσύνολο του συνόλου των στοιχείων

k-στοιχειοσύνολο (k-itemset): ένα στοιχειοσύνολο με ***k*** στοιχεία

support count (σ) ενός στοιχειοσυνόλου: ο αριθμός εμφανίσεων του στοιχείου

Υποστήριξη (Support (s)) ενός στοιχειοσυνόλου Το ποσοστό των δοσοληψιών που περιέχουν ένα στοιχειοσύνολο

Συχνό Στοιχειοσύνολο (Frequent Itemset) Ένα στοιχειοσύνολο του οποίου η υποστήριξη είναι μεγαλύτερη ή ίση από κάποια τιμή κατωφλίου *minsup*

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Ορισμοί

Κανόνας Συσχέτισης (Association Rule)
Είναι μια έκφραση της μορφής $X \rightarrow Y$,
όπου X και Y είναι στοιχειοσύνολα
$X \subseteq I$, $Y \subseteq I$, $X \cap Y = \varnothing$

Παράδειγμα: {Milk, Diaper} → {Beer}

TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Υποστήριξη Κανόνα Support (s)
Το ποσοστό των δοσοληψιών που περιέχουν και το X και το Y ($X \cup Y$)

Εμπιστοσύνη - Confidence (c)
Πόσες από τις δοσοληψίες (ποσοστό) που περιέχουν το X περιέχουν και το Y

Πρόβλημα

Εύρεση Κανόνων Συσχέτισης

Είσοδος: Ένα σύνολο από δοσοληψίες T
Έξοδος: Όλοι οι κανόνες με
support ≥ *minsup*
confidence ≥ *minconf*

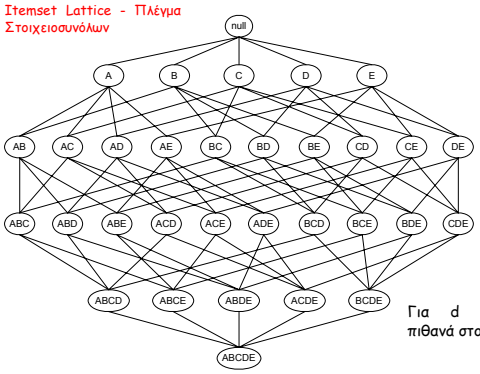
Εξόρυξη Κανόνων Συσχέτισης

Χωρισμός του προβλήματος σε δύο υπο-προβλήματα:

- Εύρεση όλων των συχνών στοιχειοσυνόλων (Frequent Itemset Generation)
  Εύρεση όλων των στοιχειοσυνόλων με υποστήριξη ≥ minsup
- Δημιουργία Κανόνων (Rule Generation)
  Για κάθε στοιχειοσύνολο, δημιούργησε κανόνες με μεγάλη υποστήριξη, όπου κάθε κανόνες είναι μια δυαδική διαμέριση του συχνού στοιχειοσυνόλου

Εύρεση Συχνών Στοιχειοσυνόλων

Itemset Lattice - Πλέγμα Στοιχειοσυνόλων



Για d στοιχεία, 2^d πιθανά στοιχειοσύνολα

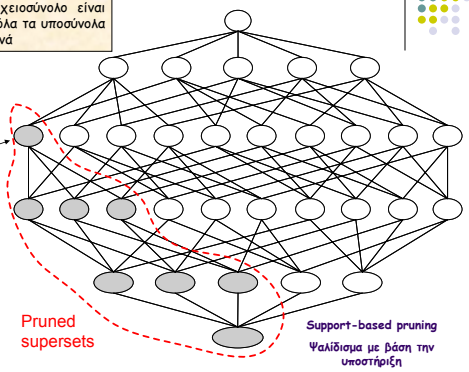
Εύρεση Συχνών Στοιχειοσυνόλων: Στρατηγική apriori

Αρχή Apriori

Αν ένα στοιχειοσύνολο είναι συχνό, τότε όλα τα υποσύνολα του είναι συχνά

Ή ισοδύναμα αν ένα στοιχειοσύνολο είναι μη συχνό, όλα τα υπερσύνολα του είναι μη συχνά

Βρέθηκε μη συχνό



Pruned supersets

Support-based pruning

Ψαλίδισμα με βάση την υποστήριξη

Στρατηγική apriori

Γενικός Αλγόριθμος

Έστω **k** = 1 *#k: μήκος στοιχειοσυνόλου*

Παρήγαγε τα συχνά **1-στοιχειοσύνολα**

Repeat until να μην παράγονται νέα συχνά στοιχειοσύνολα

1. Παρήγαγε υποψήφια **(k+1)**-στοιχειοσύνολα
2. Ψαλίδισε τα υποψήφια στοιχειοσύνολα που περιέχουν μη συχνά στοιχειοσύνολα μεγέθους k
3. Υπολόγισε την υποστήριξη κάθε υποψήφιου (k+1)-στοιχειοσυνόλου διασχίζοντας τη βάση των δοσοληψιών
4. Σβήσε τα υποψήφια στοιχειοσύνολα που δεν είναι συχνά
5. k = k + 1

Στρατηγική apriori: Δημιουργία Στοιχειοσυνόλων

Για την παραγωγή υποψήφιων k-στοιχειοσυνόλων

$F_{k-1} \times F_1$

Επέκταση κάθε συχνού (k-1) στοιχειοσυνόλου με άλλα συχνά στοιχεία

$F_{k-1} \times F_{k-1}$

Συγχώνευση δύο συχνών (k-1) στοιχειοσυνόλου αν τα πρώτα k-2 στοιχεία τους είναι τα ίδια

Παρατηρήσεις

- Για να αποφύγουμε τη δημιουργία του ίδιου στοιχειοσυνόλου, κρατάμε κάθε στοιχειοσύνολο (λεξικογραφικά) ταξινομημένο
- Είναι δυνατόν να γίνουν απλοί έλεγχοι αν τα παραγόμενα πιθανά στοιχειοσύνολα είναι συχνά ελέγχοντας αν τα υποσύνολα τους είναι συχνά και έτσι να αποφύγουμε να υπολογίσουμε την υποστήριξή τους

Στρατηγική apriori: Υπολογισμός Υποστήριξης

Για κάθε νέο υποψήφιο k+1-στοιχειοσύνολο, πρέπει να υπολογίσουμε την υποστήριξή του

- Για να μειώσουμε τον αριθμό των πράξεων, σε κάθε βήμα, αποθηκεύουμε τα υποψήφια k+1-στοιχειοσύνολα σε ένα δέντρο κατακερματισμού
- Αντί να ταιριάζουμε κάθε δοσοληψία με κάθε υποψήφιο στοιχειοσύνολο, κατακερματίζουμε τα στοιχειοσύνολα της δοσοληψίας και ενημερώνουμε μόνο τους αντίστοιχους κάδους του δέντρου κατακερματισμού των συχνών στοιχειοσυνόλων

Παραγωγή Κανόνων

Δοθέντος ενός συχνού στοιχειοσυνόλου L, βρες όλα τα μη κενά υποσύνολα f $\subset$ L τέτοια ώστε ο κανόνας f $\rightarrow$ L – f ικανοποιεί τον περιορισμό της ελάχιστης εμπιστοσύνης

Η εμπιστοσύνη για τους κανόνες που παράγονται από το ίδιο στοιχειοσύνολο έχει μια αντι-μονότονη ιδιότητα

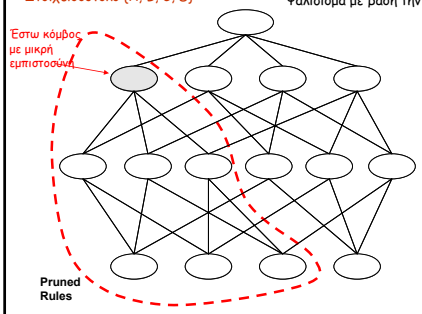
Για παράδειγμα L = {A,B,C,D}: $c(ABC \rightarrow D) \geq c(AB \rightarrow CD) \geq c(A \rightarrow BCD)$

Η εμπιστοσύνη είναι αντι-μονότονη σε σχέση με των αριθμό των στοιχείων στο RHS του κανόνα

Παραγωγή Κανόνων για τον Αλγόριθμο apriori

Πλέγμα Κανόνων για το Στοιχειοσύνολο {A, B, C, D}

Ψαλίδισμα με βάση την εμπιστοσύνη



Για κάθε συχνό στοιχειοσύνολο, ξεκινάμε με έναν κανόνα που έχει *μόνο k=1 στοιχείο στο δεξί μέρος του*

Υπολογίζουμε την εμπιστοσύνη

Παράγουμε κανόνες με **k+1** στοιχεία και υπολογίζουμε την εμπιστοσύνη τους

Για τον υπολογισμό της εμπιστοσύνης δεν χρειάζεται να διαπεράσουμε τη βάση

Αναπαράσταση Στοιχειοσυνόλων

Τα στοιχειοσύνολα που παράγονται είναι πολλά, κάποια ίσως περιττά – οδηγούν σε παραγωγή πολλών κανόνων

Ποια να κρατήσουμε;

Αντιπροσωπευτικά συχνά στοιχειοσύνολα:

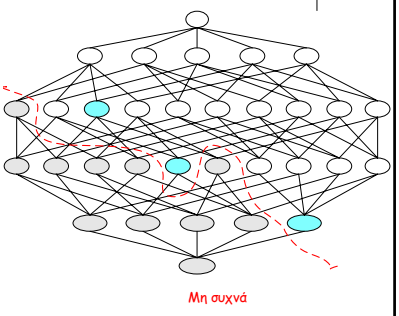
- Maximal συχνά
- Κλειστά συχνά

Αναπαράσταση Στοιχειοσυνόλων

Ένα στοιχειοσύνολο είναι maximal συχνό αν κανένα από τα άμεσα υπερσύνολά του δεν είναι συχνό

Προσφέρουν μια συνοπτική αναπαράσταση των συχνών στοιχειοσυνόλων: το μικρότερο σύνολο στοιχειοσυνόλων από το οποίο μπορούμε να πάρουμε όλα τα συχνά στοιχειοσύνολα – είναι τα υποσύνολά τους

Πρόβλημα: Δεν προσφέρουν καμιά πληροφορία για την υποστήριξη των υποσυνόλων τους



Αναπαράσταση Στοιχειοσυνόλων

Ένα στοιχειοσύνολο είναι **κλειστό (closed)** αν κανένα από τα άμεσα υπερσύνολα του δεν έχει την ίδια υποστήριξη με αυτό (δηλαδή, έχει μικρότερη)

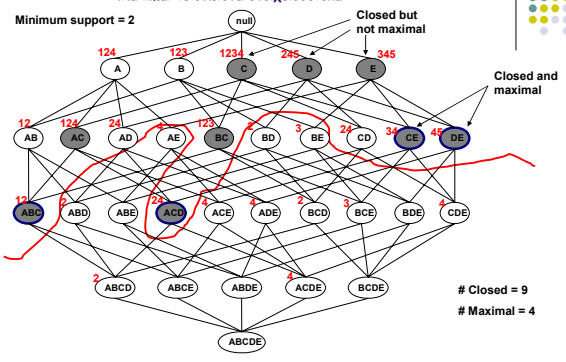
Ένα στοιχειοσύνολο είναι **κλειστό συχνό στοιχειοσύνολο** αν είναι κλειστό και η υποστήριξη του είναι μικρότερη ή ίση με minsup

Πάλι τα υποσύνολα τους μας δίνουν όλα τα συχνά υποσύνολα, τώρα όμως μπορούμε να υπολογίσουμε την υποστήριξη των υποσυνόλων τους

Πως: Η υποστήριξη ενός μη κλειστού στοιχειοσυνόλου πρέπει να είναι ίση με την μεγαλύτερη υποστήριξη ανάμεσα στα υπερσύνολά του

Αναπαράσταση Στοιχειοσυνόλων

Maximal vs Κλειστά στοιχειοσύνολα



Μέτρηση Ενδιαφέροντος

Στην αρχική διατύπωση του προβλήματος της εξόρυξης κανόνων συσχέτισης χρησιμοποιήθηκαν ως μέτρα μόνο η *υποστήριξη* και η *εμπιστοσύνη*

Μια σειρά μέτρων βασισμένα στη στατιστική

- $P(S \wedge B) = P(S) \times P(B)$ => Στατιστική ανεξαρτησία
- $P(S \wedge B) > P(S) \times P(B)$ => Positively correlated (θετική συσχέτιση)
- $P(S \wedge B) < P(S) \times P(B)$ => Negatively correlated (αρνητική συσχέτιση)

Μέτρηση Ενδιαφέροντος

Έστω ένας κανόνας, X → Y, η πληροφορία που χρειάζεται για τον υπολογισμό του ενδιαφέροντος του κανόνα μπορεί να υπολογιστεί από τον contingency table

Contingency table for X → Y

	Y	$\overline{Y}$	
X	f_{11}	f_{10}	f_{1+}
$\overline{X}$	f_{01}	f_{00}	f_{0+}
	f_{+1}	f_{+0}	|T|

f_{11}: support of X and Y
f_{10}: support of X and $\overline{Y}$
f_{01}: support of $\overline{X}$ and Y
f_{00}: support of $\overline{X}$ and $\overline{Y}$

Μέτρηση συχνότητας εμφάνισης

Χρησιμοποιείται για τον ορισμό διαφόρων μέτρων

Μέτρηση Ενδιαφέροντος

Μέτρα που λαμβάνουν υπ΄ όψιν τους στη στατιστική εξάρτηση

X → Y

$$Lift = \frac{P(Y \mid X)}{P(Y)}$$

$$Interest = \frac{P(X,Y)}{P(X)P(Y)}$$

$$PS = P(X,Y) - P(X)P(Y)$$

$$\phi - coefficient = \frac{P(X,Y) - P(X)P(Y)}{\sqrt{P(X)[1-P(X)]P(Y)[1-P(Y)]}}$$

Γενικά έχουν προταθεί 21 τέτοια μέτρα

Άλλοι Μέθοδοι Υπολογισμού Συχνών Στοιχειοσυνόλων

Άλλοι Μέθοδοι Υπολογισμού Συχνών Στοιχειοσυνόλων

Ο Apriori από τους παλιότερους, αλλά:

Συχνά μεγάλο I/O επειδή κάνει πολλαπλά περάσματα στη βάση των δοσοληψιών

Κακή απόδοση όταν οι δοσοληψίες έχουν μεγάλο πλάτος

Άλλες μέθοδοι:

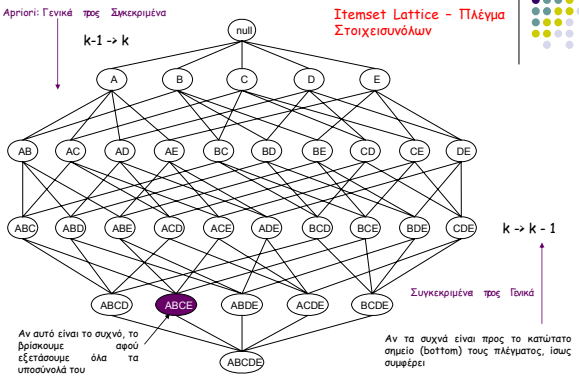
- Διαφορετικές διασχίσεις του πλέγματος των στοιχειοσυνόλων
- Αναπαράσταση Συνόλου Δοσοληψιών

Άλλοι Μέθοδοι Υπολογισμού Συχνών Στοιχειοσυνόλων

Apriori: Γενικά προς Συγκεκριμένα

k-1 -> k

Itemset Lattice - Πλέγμα Στοιχεισυνόλων



k -> k - 1

Συγκεκριμένα προς Γενικά

Αν αυτό είναι το συχνό, το βρίσκουμε αφού εξετάσουμε όλα τα υποσύνολά του

Αν τα συχνά είναι προς το κατώτατο σημείο (bottom) τους πλέγματος, ίσως συμφέρει

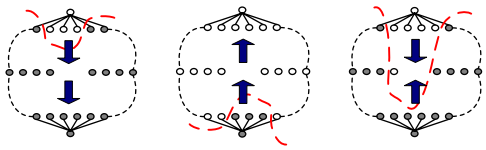
Άλλοι Μέθοδοι Υπολογισμού Συχνών Στοιχειοσυνόλων

Διάσχιση του Πλέγματος των Στοιχειοσυνόλων: Συγκεκριμένα-προς-Γενικά vs Γενικά-προς-Συγκεκριμένα

k -> k – 1 (συγκεκριμένο-προς-γενικό)

Ποιο χρήσιμο για τον εντοπισμό maximal συχνών στοιχειοσυνόλων σε πυκνές (δηλ, με μεγάλο πλάτος δοσοληψίες) όπου το συχνό στοιχειοσύνολο βρίσκεται κοντά στο κατώτατο σημείο του πλέγματος

Αν συχνό, δε χρειάζεται να ελέγξουμε κανένα από τα υποσύνολά του



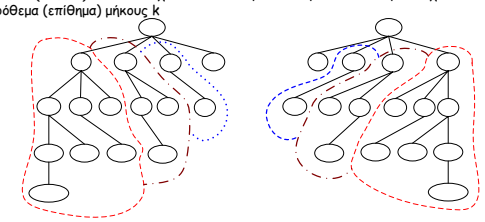
Άλλοι Μέθοδοι Υπολογισμού Συχνών Στοιχειοσυνόλων

Διάσχιση του Πλέγματος των Στοιχειοσυνόλων: Κλάσεις Ισοδυναμίας

Χωρισμός των στοιχειοσυνόλων του πλέγματος σε ξένες μεταξύ τους ομάδες (κλάσεις ισοδυναμίας) και εξέταση των στοιχειοσυνόλων ανά κλάσεις

Apriori: ορίζει τις κλάσεις με βάση το μήκος k των στοιχειοσυνόλων, πρώτα αυτά μήκους 1, μετά μήκους 2 κοκ

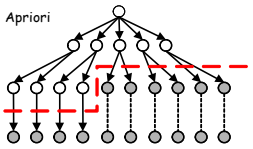
Prefix (Suffix): Δύο στοιχειοσύνολα ανήκουν στην ίδια κλάση αν έχουν κοινό πρόθεμα (επίθημα) μήκους k



Άλλοι Μέθοδοι Υπολογισμού Συχνών Στοιχειοσυνόλων

Διάσχιση του Πλέγματος των Στοιχειοσυνόλων: BFS vs DFS

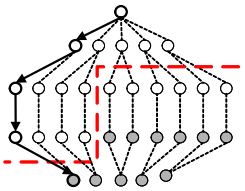
Apriori



BFS: Breadth-First-Search

Χρήσιμο για την εύρεση maximal συχνών στοιχειοσυνόλων γιατί τα εντοπίζει πιο γρήγορα από το BFS

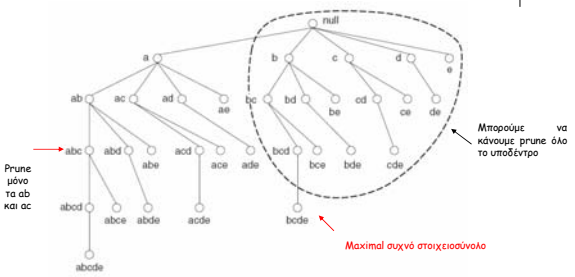
Μόλις εντοπιστεί το maximal, είναι δυνατόν να κλαδευτούν πολλά υποσύνολα του



DFS: Depth-First-Search

Άλλοι Μέθοδοι Υπολογισμού Συχνών Στοιχειοσυνόλων

Διάσχιση του Πλέγματος των Στοιχειοσυνόλων: BFS vs DFS



Prune μόνο τα ab και ac

Μπορούμε να κάνουμε prune όλο το υποδέντρο

Maximal συχνό στοιχειοσύνολο

Figure 6.22. Generating candidate itemsets using the depth-first approach.

Άλλοι Τρόποι Υπολογισμού

Αναπαράσταση της Βάσης Δεδομένων: Οριζόντια vs Κάθετη

Αυτό χρησιμοποιεί ο apriori

Horizontal Data Layout

TID	Items
1	A,B,E
2	B,C,D
3	C,E
4	A,C,D
5	A,B,C,D
6	A,E
7	A,B
8	A,B,C
9	A,C,D
10	B

Εναλλακτικά:

Για κάθε στοιχείο σε ποιες δοσοληψίες εμφανίζεται

Vertical Data Layout

A	B	C	D	E
1	1	2	2	1
4	2	3	4	3
5	5	4	5	6
6	7	8	9	
7	8	9		
8	10			
9				

Η υποστήριξη υπολογίζεται παίρνοντας τις τομές των TID-λιστών

null

Άλλοι Τρόποι Υπολογισμού

Η υποστήριξη υπολογίζεται παίρνοντας τις τομές των TID-λιστών

A
1
4
5
6
7
8
9

∧

B
1
2
5
7
8
10

→

AB
1
5
7
8

- Η υποστήριξη ενός k-στοιχειοσυνόλου υπολογίζεται παίρνοντας τις τομές των TID-λιστών δύο από τα (k-1)-ύπο-στοιχειοσύνολα του.
- Πλεονέκτημα: πολύ γρήγορος υπολογισμός της υποστήριξης
- Πρόβλημα, αν οι TID-λίστες είναι μεγάλες και δε χωρούν στη μνήμη

A

AC

ABD

Ο Αλγόριθμος FP-Growth

A

B AC

ABC

(a) Prefix tree

Αλγόριθμος FP-Growth

Χρησιμοποιεί μια συμπιεσμένη αναπαράσταση της βάσης με τη μορφή ενός FP-δέντρου

- Το δέντρο μοιάζει με prefix tree (trie)
- Ο αλγόριθμος κατασκευής διαβάζει μια δοσοληψία τη φορά, απεικονίζει τη δοσοληψία σε ένα μονοπάτι του FP-δέντρου
- Μερικά μονοπάτια μπορεί να επικαλύπτονται: όσο περισσότερα μονοπάτια επικαλύπτονται, τόσο καλύτερη συμπίεση

Μόλις το FP-δέντρο κατασκευαστεί, ο αλγόριθμος χρησιμοποιεί μια αναδρομική διαίρει-και-βασίλευε (divide-and-conquer) προσέγγιση για την εξόρυξη των συχνών στοιχειοσυνόλων

Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Το FP-δέντρο είναι ένα προθεματικό δέντρο

Επειδή έχουμε σύνολα, κάπως πρέπει να τα διατάξουμε ώστε να βρίσκουμε προθέματα

Δηλαδή δε μπορεί το ένα σύνολο να είναι {A, B} και το άλλο {B, C, A} γιατί χάνουμε το κοινό πρόθεμα AB (ή BA)

Άρα τα στοιχεία σε κάθε σύνολο πρέπει να ακολουθούν κάποια διάταξη, έστω τη λεξικογραφική (θα δούμε αργότερα αν κάτι άλλο συμφέρει καλύτερα)

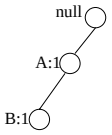
Αρχικά, το δέντρο κενό

Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Διάβασμα TID=1:



Κάθε κόμβος ετικέτα: ποιο στοιχείο και τη συχνότητα εμφάνισης (υποστήριξη) – πόσες δοσοληψίες φτάνουν σε αυτόν

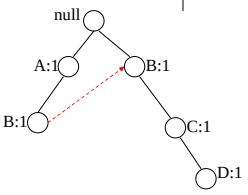
Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Διάβασμα TID=1:

Διάβασμα TID=2:



Κάθε κόμβος ετικέτα, ποιο στοιχείο και τη συχνότητα εμφάνισης (υποστήριξη) – πόσες δοσοληψίες φτάνουν σε αυτόν

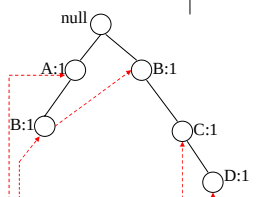
Επίσης, δείκτες μεταξύ των κόμβων που αναφέρονται στο ίδιο στοιχείο

Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Διάβασμα TID=1, 2:



Πίνακας Δεικτών

Item	Pointer
A	
B	
C	
D	
E	

Επίσης, κρατάμε δείκτες για να βοηθήσουν στον υπολογισμό των συχνών στοιχειοσυνόλων

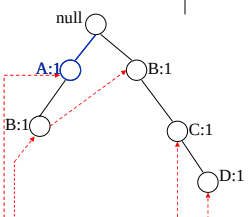
Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Διάβασμα TID=1, 2:

Διάβασμα TID=3



Πίνακας Δεικτών

Item	Pointer
A	
B	
C	
D	
E	

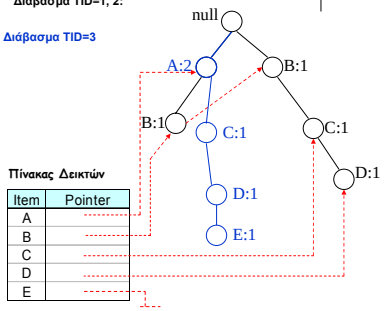
Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Διάβασμα TID=1, 2:

Διάβασμα TID=3



Πίνακας Δεικτών

Item	Pointer
A	
B	
C	
D	
E	

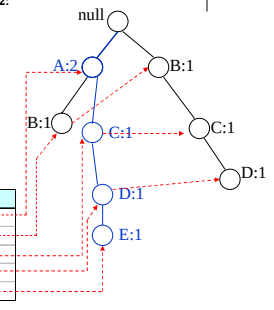
Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Διάβασμα TID=1, 2:

Διάβασμα TID=3



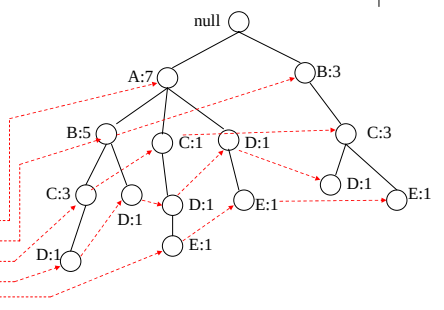
Πίνακας Δεικτών

Item	Pointer
A	
B	
C	
D	
E	

Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}



Header table

Item	Pointer
A	
B	
C	
D	
E	

Αλγόριθμος FP-Growth

Μέγεθος FP-δέντρου

Κάθε δοσοληψία αντιστοιχεί σε ένα μονοπάτι από τη ρίζα

Το μέγεθος του δέντρου συνήθως μικρότερο των δεδομένων, αν υπάρχουν κοινά προθέματα

Αν όλες οι δοσοληψίες τα ίδια δεδομένα, μόνο ένα κλαδί

Αν όλες διαφορετικές, ο χώρος μεγαλύτερος (γιατί αποθηκεύεται περισσότερη πληροφορία, όπως δείκτες μεταξύ των κόμβων αλλά και συχνότητες εμφάνισης)

Αλγόριθμος FP-Growth

Κατασκευή FP-δέντρου

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}

Το τελικό δέντρο, εξαρτάται από τη διάταξη: άλλη διάταξη -> άλλα προθέματα

(Συνήθως) Μικρότερο δέντρο, αν όχι λεξικογραφικά, αλλά με βάση τη συχνότητα εμφάνισης -> Αρχικά, διαβάζουμε όλα τα δεδομένα μια φορά ώστε να υπολογιστεί ο μετρητής υποστήριξης κάθε στοιχείου, και διατάσσουμε τα στοιχεία με βάση αυτό

Για το παράδειγμα, σ(A)=7, σ(B)=8, σ(C)=7, σ(D)=5, σ(E)=3

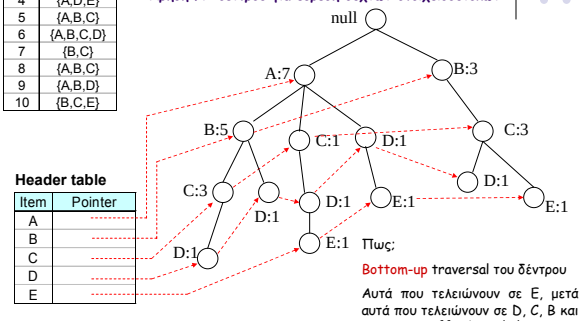
Άρα, διάταξη B,A,C,D,E

TID	Items
1	{B,A}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{B,A,C}
6	{B,A,C,D}
7	{B,C}
8	{B,A,C}
9	{B,A,D}
10	{B,C,E}

Αλγόριθμος FP-Growth

Χρήση FP-δέντρου για εύρεση συχνών στοιχειοσυνόλων

TID	Items
1	{A,B}
2	{B,C,D}
3	{A,C,D,E}
4	{A,D,E}
5	{A,B,C}
6	{A,B,C,D}
7	{B,C}
8	{A,B,C}
9	{A,B,D}
10	{B,C,E}



Header table

Item	Pointer
A	
B	
C	
D	
E	

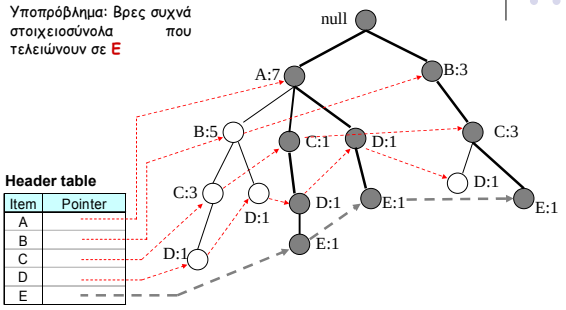
Πως;

Bottom-up traversal του δέντρου

Αυτά που τελειώνουν σε E, μετά αυτά που τελειώνουν σε D, C, B και τέλος A - suffix-based classes

Αλγόριθμος FP-Growth

Υποπρόβλημα: Βρες συχνά στοιχειοσύνολα που τελειώνουν σε **E**



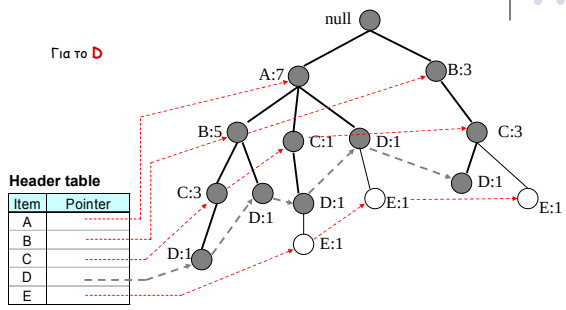
Header table

Item	Pointer
A	
B	
C	
D	
E	

Θα δούμε στη συνέχεια πως υπολογίζεται η υποστήριξη για τα πιθανά στοιχειοσύνολα

Αλγόριθμος FP-Growth

Για το **D**

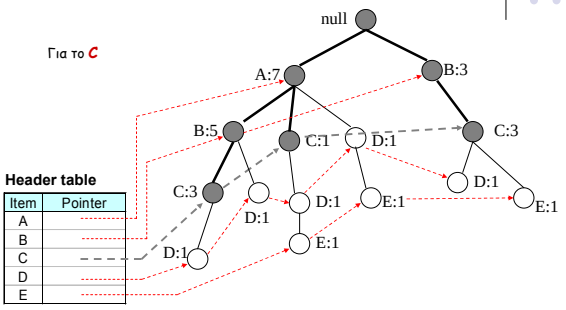


Header table

Item	Pointer
A	
B	
C	
D	
E	

Αλγόριθμος FP-Growth

Για το **C**

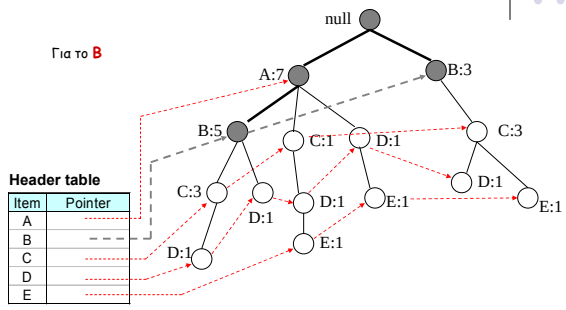


Header table

Item	Pointer
A	
B	
C	
D	
E	

Αλγόριθμος FP-Growth

Για το **B**

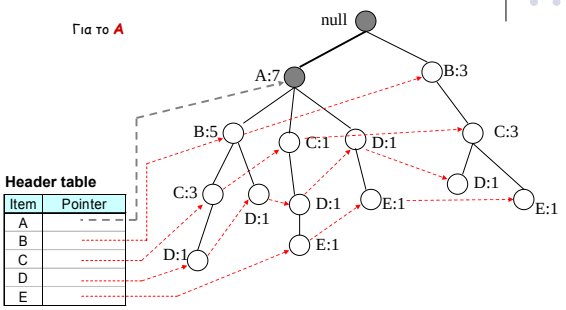


Header table

Item	Pointer
A	
B	
C	
D	
E	

Αλγόριθμος FP-Growth

Για το **A**



Header table

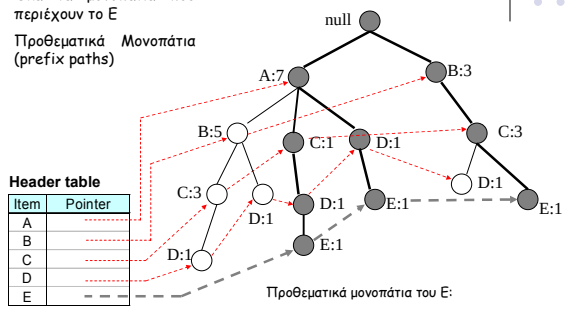
Item	Pointer
A	
B	
C	
D	
E	

Αλγόριθμος FP-Growth

Φάση 1

Όλα τα μονοπάτια που περιέχουν το E

Προθεματικά Μονοπάτια (prefix paths)



Header table

Item	Pointer
A	
B	
C	
D	
E	

Προθεματικά μονοπάτια του E:

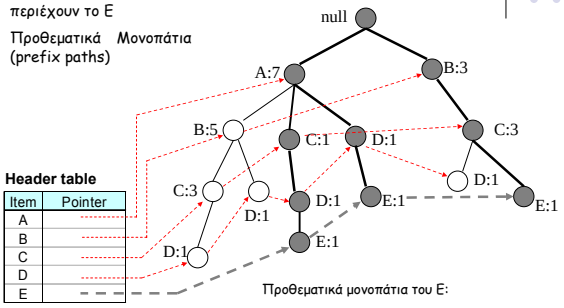
{E}, {D,E}, {C,D,E}, {A,D,E}, {A,C,D,E}, {C,E}, {B,C,E}

Αλγόριθμος FP-Growth

Φάση 1

Όλα τα μονοπάτια που περιέχουν το E

Προθεματικά Μονοπάτια (prefix paths)



Header table

Item	Pointer
A	
B	
C	
D	
E	

Προθεματικά μονοπάτια του E:

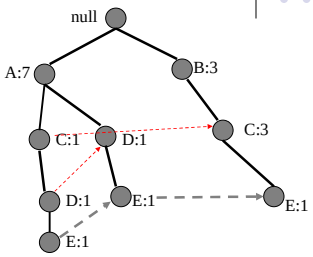
{E}, {D,E}, {C,D,E}, {A,D,E}, {A,C,D,E}, {C,E}, {B,C,E}

Αλγόριθμος FP-Growth

Φάση 1

Όλα τα μονοπάτια που περιέχουν το E

Προθεματικά Μονοπάτια (prefix paths)



Προθεματικά μονοπάτια του E:

{E}, {D,E}, {C,D,E}, {A,D,E}, {A,C,D,E}, {C,E}, {B,C,E}

Αλγόριθμος FP-Growth

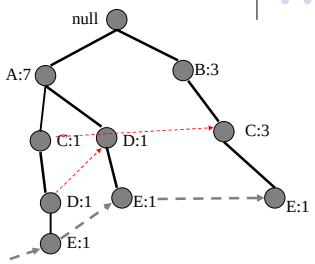
Έστω minsup = 2

Βρες την υποστήριξη του {E}

Πως;

Ακολούθησε τους συνδέσμους αθροίζοντας 1+1+1=3>2

Οπότε {E} συχνό



{E} συχνό άρα προχωράμε για DE, CE, BE, AE

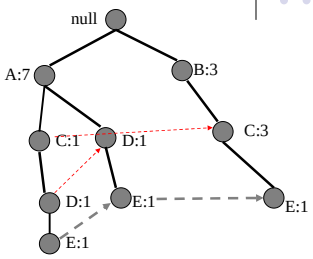
Αλγόριθμος FP-Growth

Μετατροπή των προθεματικών δέντρων σε FP-δέντρο υπό συνθήκες (conditional FP-tree)

Δύο αλλαγές

(1) Αλλαγή των μετρητών

(2) Περικοπή



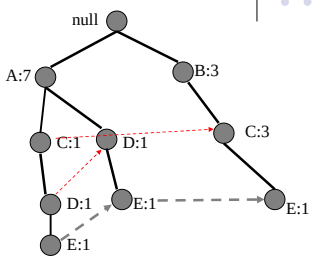
{E} συχνό άρα προχωράμε για DE, CE, BE, AE

Αλγόριθμος FP-Growth

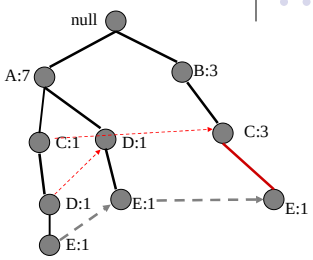
Αλλαγή μετρητών

Οι μετρητές σε κάποιους κόμβους περιλαμβάνουν δοσοληψίες που δεν έχουν το E

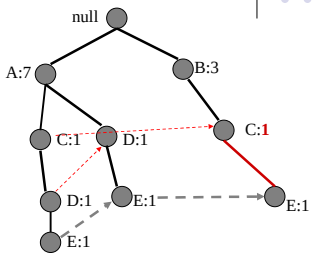
Πχ στο null->B->C->E μετράμε και την {B, C}



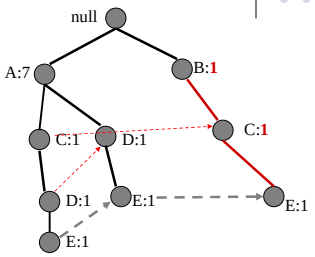
Αλγόριθμος FP-Growth



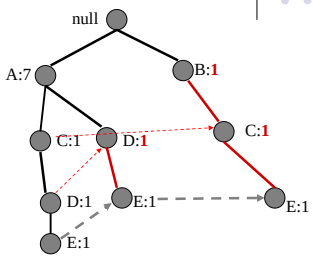
Αλγόριθμος FP-Growth



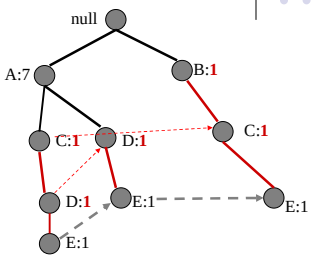
Αλγόριθμος FP-Growth



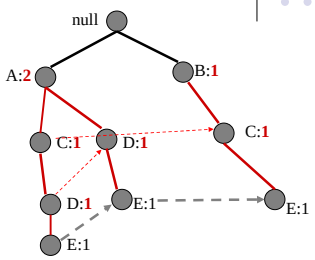
Αλγόριθμος FP-Growth



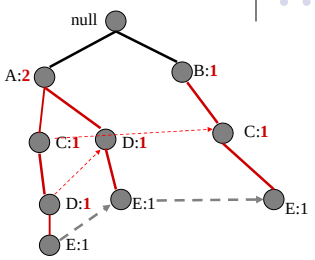
Αλγόριθμος FP-Growth



Αλγόριθμος FP-Growth



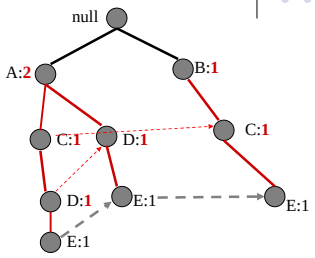
Αλγόριθμος FP-Growth



Αλγόριθμος FP-Growth

Περικοπή (truncate)

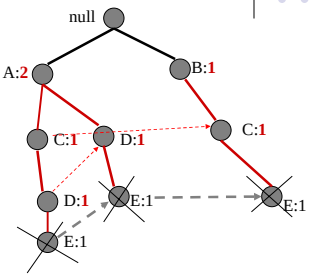
Σβήσε τους κόμβους του E



Αλγόριθμος FP-Growth

Περικοπή (truncate)

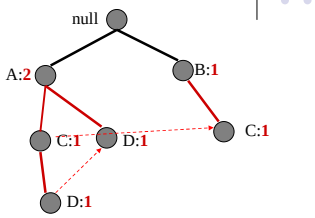
Σβήσε τους κόμβους του E



Αλγόριθμος FP-Growth

Περικοπή (truncate)

Σβήσε τους κόμβους του E

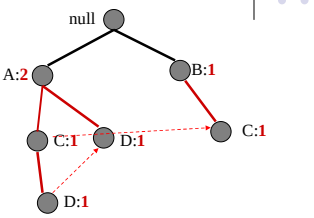


Αλγόριθμος FP-Growth

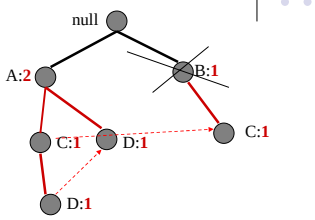
Πιθανή περαιτέρω περικοπή

Κάποια στοιχεία μπορεί να έχουν υποστήριξη μικρότερη της ελάχιστης

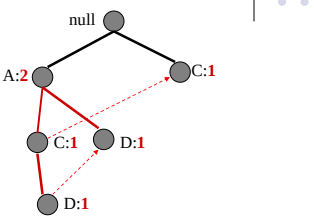
Πχ το B -> περικοπή



Αλγόριθμος FP-Growth



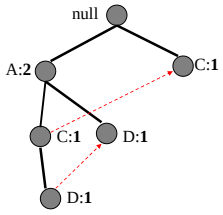
Αλγόριθμος FP-Growth



Αλγόριθμος FP-Growth

Υπο-συνθήκη FP-δέντρο για το E

Ο αλγόριθμος επαναλαμβάνεται για το {D, E}, {C, E}, {A, E}

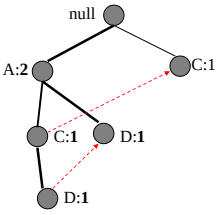


Αλγόριθμος FP-Growth

Φάση 1

Όλα τα μονοπάτια που περιέχουν το D (DE)

Προθεματικά Μονοπάτια (prefix paths)

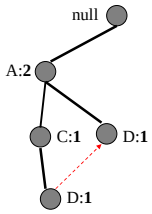


Αλγόριθμος FP-Growth

Φάση 1

Όλα τα μονοπάτια που περιέχουν το D (DE)

Προθεματικά Μονοπάτια (prefix paths)



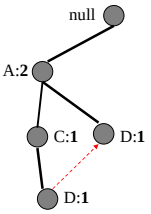
Αλγόριθμος FP-Growth

Βρες την υποστήριξη του {D, E}

Πως;

Ακολούθησε τους συνδέσμους αθροίζοντας 1+1=2≥2

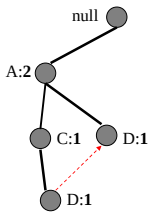
Οπότε {D, E} συχνό



Αλγόριθμος FP-Growth

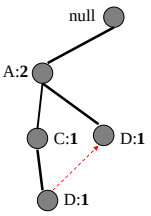
Κατασκεύασε το υπο-συνθήκη FP-δέντρο για το {D, E}

1. Αλλαγή υποστήριξης
2. Περικοπές κόμβων



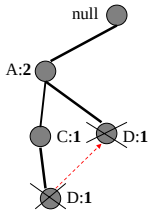
Αλγόριθμος FP-Growth

1. Αλλαγή υποστήριξης



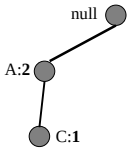
Αλγόριθμος FP-Growth

2. Περικοπές κόμβων



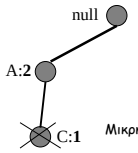
Αλγόριθμος FP-Growth

2. Περικοπές κόμβων



Αλγόριθμος FP-Growth

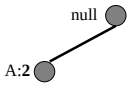
2. Περικοπές κόμβων



Μικρή υποστήριξη

Αλγόριθμος FP-Growth

Τελικό υπο-συνθήκη FP-δέντρο για το {D, E}



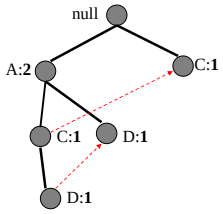
Υποστήριξη του A είναι ≥ minsup -> {A, D, E} συχνό

Αφού μόνο έναν κόμβο, επιστροφή στο επόμενο υποπρόβλημα

Αλγόριθμος FP-Growth

Υπο-συνθήκη FP-δέντρο για το E

Ο αλγόριθμος επαναλαμβάνεται για το ~~{D, E}~~, **{C, E}**, {A, E}

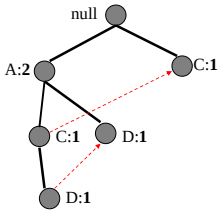


Αλγόριθμος FP-Growth

Βάση 1

Όλα τα μονοπάτια που περιέχουν το C (CE)

Προθεματικά Μονοπάτια (prefix paths)

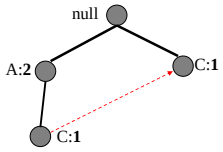


Αλγόριθμος FP-Growth

Βάση 1

Όλα τα μονοπάτια που περιέχουν το *C* (*CE*)

Προθεματικά Μονοπάτια (prefix paths)



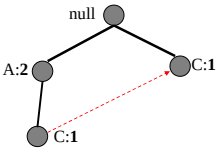
Αλγόριθμος FP-Growth

Βρες την υποστήριξη του {*C*, *E*}

Πως;

Ακολούθησε τους συνδέσμους αθροίζοντας 1+1=2≥2

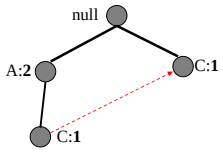
Οπότε {*C*, *E*} συχνό



Αλγόριθμος FP-Growth

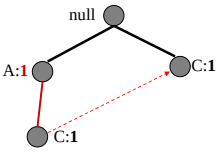
Κατασκεύασε το υπο-συνθήκη FP-δέντρο για το {*C*, *E*}

1. Αλλαγή υποστήριξης
2. Περικοπές κόμβων



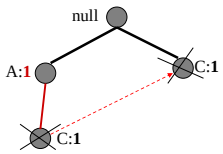
Αλγόριθμος FP-Growth

1. Αλλαγή υποστήριξης



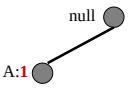
Αλγόριθμος FP-Growth

2. Περικοπή Κόμβων



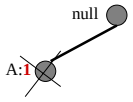
Αλγόριθμος FP-Growth

2. Περικοπή Κόμβων



Αλγόριθμος FP-Growth

2. Περικοπή Κόμβων



Αλγόριθμος FP-Growth

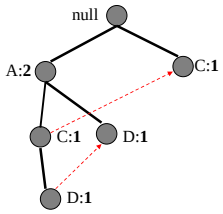
2. Περικοπή Κόμβων

Άρα, επιστροφή στο επόμενο υποπρόβλημα

Αλγόριθμος FP-Growth

Υπο-συνθήκη FP-δέντρο για το E

Ο αλγόριθμος επαναλαμβάνεται για το ~~{D, E}~~, ~~{C, E}~~, **{A, E}**

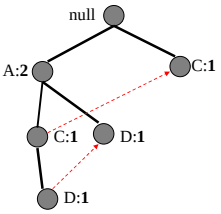


Αλγόριθμος FP-Growth

Βάση 1

Όλα τα μονοπάτια που περιέχουν το A (AE)

Προθεματικά Μονοπάτια (prefix paths)

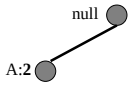


Αλγόριθμος FP-Growth

Βάση 1

Όλα τα μονοπάτια που περιέχουν το A (AE)

Προθεματικά Μονοπάτια (prefix paths)

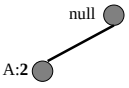


Αλγόριθμος FP-Growth

Βρες την υποστήριξη του {A, E}

Οπότε {A, E} συχνό

Δε χρειάζεται να φτιάξουμε υπο-συνθήκη FP-δέντρο για το {A, E}



Αλγόριθμος FP-Growth

Άρα για το E

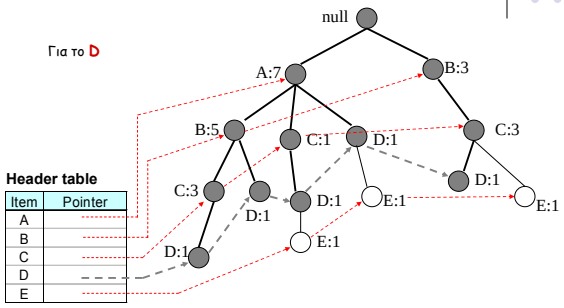
Έχουμε τα εξής συχνά στοιχειοσύνολα

{E} {D, E} {A, D, E} {C, E} {A, E}

Συνεχίζουμε για το D

Αλγόριθμος FP-Growth

Για το **D**



Header table

Item	Pointer
A	
B	
C	
D	
E	

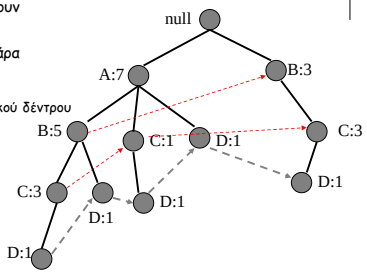
Αλγόριθμος FP-Growth

Φάση 1

Όλα τα προθεματικά μονοπάτια που περιέχουν το D

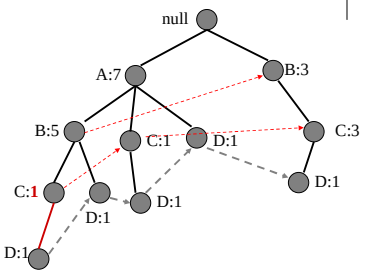
Υποστήριξη 5>2 -> άρα συχνό

Μετατροπή του προθεματικού δέντρου σε FP-δέντρο υπό συνθήκη



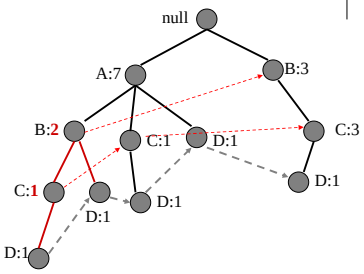
Αλγόριθμος FP-Growth

1. Αλλαγή υποστήριξης



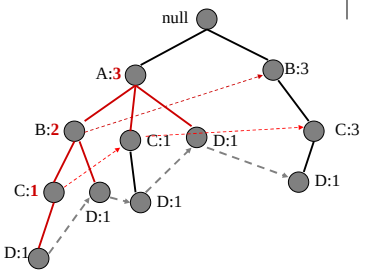
Αλγόριθμος FP-Growth

1. Αλλαγή υποστήριξης



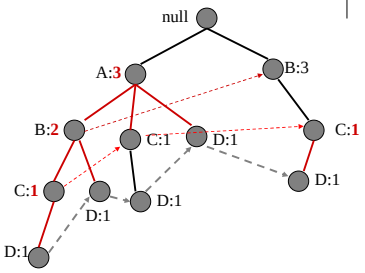
Αλγόριθμος FP-Growth

1. Αλλαγή υποστήριξης



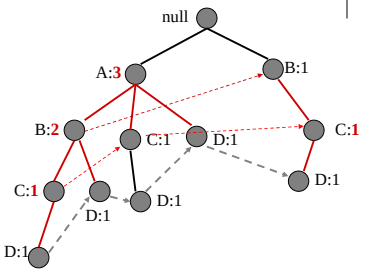
Αλγόριθμος FP-Growth

1. Αλλαγή υποστήριξης



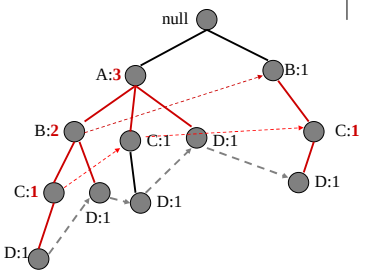
Αλγόριθμος FP-Growth

1. Αλλαγή υποστήριξης



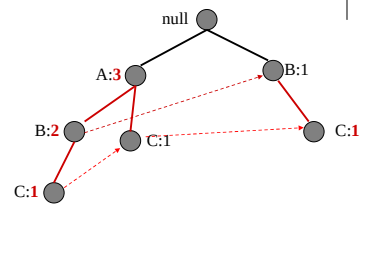
Αλγόριθμος FP-Growth

2. Περικοπή Κόμβων



Αλγόριθμος FP-Growth

2. Περικοπή Κόμβων

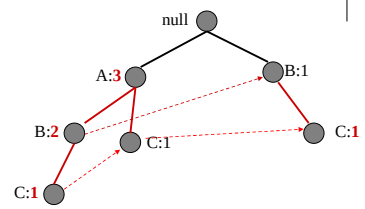


Αλγόριθμος FP-Growth

Προθεματικά δέντρα και υποσυνθήκη δέντρα

Για τα AD, BD και CD

κοκ



Αλγόριθμος FP-Growth

Παράδειγμα τεχνικής διαίρει-και-βασίλευε

Σε κάθε αναδρομικό βήμα, λύνεται και ένα υπο-πρόβλημα:

- Κατασκευάζεται το προθεματικό δέντρο
- Υπολογίζεται η νέα υποστήριξη για τους κόμβους του
- Περικόβονται οι κόμβοι με μικρή υποστήριξη

Επειδή τα υποπροβλήματα είναι ξένα μεταξύ τους, δεν δημιουργούνται τα ίδια συχνά στοιχειοσύνολα δυο φορές

Ο υπολογισμός της υποστήριξης είναι αποδοτικός – γίνεται ταυτόχρονα με τη δημιουργία των συχνών στοιχειοσυνόλων

Αλγόριθμος FP-Growth

Η απόδοση του FP-Growth εξαρτάται από τον παράγοντα συμπίεσης του συνόλου των δεδομένων (compaction factor)

Αν τα τελικά δέντρα είναι «θαμνώδη» (bushy) τότε δε δουλεύει καλά, αυξάνεται ο αριθμός των υποπροβλημάτων (οι αναδρομικές κλήσεις)

Επίδραση της Υποστήριξης

Κατανομή Υποστήριξης

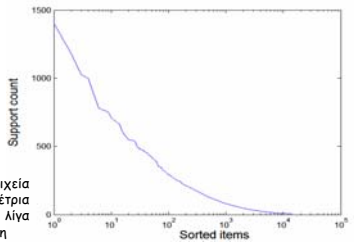
- Η απόδοση των αλγορίθμων εξαρτάται από τα δεδομένα εισόδου, πχ ο apriori από τον αριθμό των στοιχείων, το πλάτος των δοσοληψιών, ο FP-Growth από την τομή των δοσοληψιών
- Επίσης, από την τιμή της ελάχιστης υποστήριξης (minsup). Πως θα προσδιοριστεί μια κατάλληλη τιμή για το *minsup;*
  - Αν η τιμή είναι πολύ υψηλή, μπορεί να χαθούν στοιχειοσύνολα που περιέχουν ενδιαφέροντα σπάνια στοιχεία (πχ ακριβά προϊόντα)
  - Αν η τιμή είναι πολύ χαμηλή, οι μέθοδοι γίνονται ακριβοί γιατί ο αριθμός των υποψήφιων στοιχειοσυνόλων είναι πολύ μεγάλος
  - Ο αριθμός των συχνών στοιχειοσυνόλων γίνεται πολύ μεγάλος

Κατανομή Υποστήριξης

Επιπρόσθετα, η χρήση μόνο μίας ελάχιστης υποστήριξης μπορεί να μην αρκεί

Για πολλά πραγματικά δεδομένα η κατανομή της υποστήριξης δεν είναι ομοιόμορφη (skewed support distribution)

Support distribution of a retail data set



Τα περισσότερα στοιχεία έχουν μικρή ή μέτρια υποστήριξη και μόνο λίγα έχουν μεγάλη υποστήριξη

Κατανομή Υποστήριξης

Ομάδα	G1	G2	G3
Υποστήριξη	<1%	1%-90%	>90%
Αριθμός στοιχείων	1735	358	20

Παράξενοι κανόνες μεταξύ G1 και G3 (χαβιάρι και γάλα)

Cross-support patterns

Πολλαπλές Τιμές Υποστήριξης

Πολλαπλές Ελάχιστες Τιμές Υποστήριξης

MS(i): ελάχιστη υποστήριξη για το στοιχείο i

- Π.χ.: MS(Milk)=5%, MS(Coke) = 3%, MS(Broccoli)=0.1%, MS(Salmon)=0.5%
- MS({Milk, Broccoli}) = min (MS(Milk), MS(Broccoli)) = 0.1%

Πρόβλημα: Η υποστήριξη παύει να είναι αντιμονότονη:

- Έστω: Support(Milk, Coke) = 1.5% and Support(Milk, Coke, Broccoli) = 0.5%
- {Milk,Coke} είναι μη συχνό αλλά το {Milk,Coke,Broccoli} είναι συχνό

Λόγω του Broccoli που κατεβάζει το minsup

Πολλαπλές Τιμές Υποστήριξης

Multiple Minimum Support (Liu 1999)

- Ταξινόμησε τα στοιχεία με βάση την ελάχιστη τιμή υποστήριξης (σε αύξουσα διάταξη)
  - πχ.: MS(Milk)=5%, MS(Coke) = 3%, MS(Broccoli)=0.1%, MS(Salmon)=0.5%
  - Διάταξη: Broccoli, Salmon, Coke, Milk
- Τροποποίηση του Apriori έτσι ώστε:
  - L_1 : σύνολο συχνών στοιχειοσυνόλων
  - F_1 : σύνολο στοιχείων που η υποστήριξη τους είναι ≥ MS(1) όπου MS(1) είναι $\min_i(MS(i))$
  - C_2 : τα υποψήφια στοιχειοσύνολα μεγέθους 2 παράγονται από το F_1 αντί του L_1

Πολλαπλές Τιμές Υποστήριξης

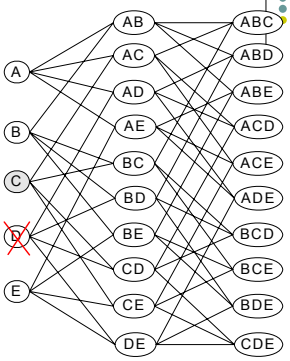
- Τροποποιήσεις στον Apriori (Βήμα Ψαλιδίσματος):
  - Στον παραδοσιακό Apriori,
    - Ένα υποψήφιο (k+1)-στοιχειοσύνολο δημιουργείται συγχωνεύοντας δυο συχνά k-στοιχειοσύνολα
    - Το υποψήφιο ψαλιδίζεται αν περιέχει ένα (οποιοδήποτε) μη συχνό k-στοιχειοσύνολο
  - Τροποποίηση βήματος ψαλιδίσματος:
    - Ψαλίδισε μόνο αν το υποσύνολο περιέχει το πρώτο στοιχείο

    πχ Candidate={Broccoli, Coke, Milk} (διατεταγμένα με βάση την μικρότερη ελάχιστη υποστήριξη)

    {Broccoli, Coke} και {Broccoli, Milk} είναι συχνά αλλά {Coke, Milk} είναι μη συχνό
      - Candidate δε σβήνεται γιατί το {Coke,Milk} δεν περιέχει το πρώτο ου στοιχείο, δηλαδή, Broccoli.

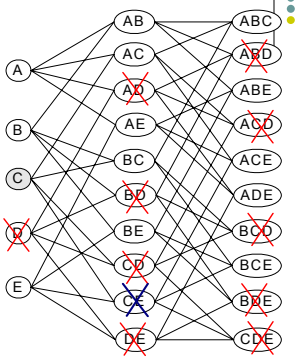
Πολλαπλές Τιμές Υποστήριξης

Item	MS(I)	Sup(I)
A	0.10%	0.25%
B	0.20%	0.26%
C	0.30%	0.29%
D	0.50%	0.05%
E	3%	4.20%



Πολλαπλές Τιμές Υποστήριξης

Item	MS(I)	Sup(I)
A	0.10%	0.25%
B	0.20%	0.26%
C	0.30%	0.29%
D	0.50%	0.05%
E	3%	4.20%



Ακολουθιακά Δεδομένα

Ακολουθιακά Δεδομένα

Μέχρι στιγμής, οι δοσοληψίες *σύνολα από στοιχεία*, δεν έχει σημασία η σειρά εμφάνισης των στοιχείων σε κάθε δοσοληψία – επίσης, *σύνολα από δοσοληψίες*, δεν έχει σημασία η σειρά εμφάνισης κάθε δοσοληψίας

Ωστόσο, πολλά δεδομένα στο «Καλάθι της νοικοκυράς» περιέχουν χρονική πληροφορία, πχ ποιες δοσοληψίες κάνει ένας πελάτης σε μια συγκεκριμένη χρονική περίοδο

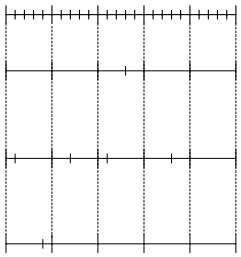
Επίσης, γεγονότα που είναι αποτελέσματα επιστημονικών πειραμάτων

Δηλαδή, σχέση διάταξης είτε χρονική, είτε χωρική

Sequence Database
(Ακολουθιακή Βάση Δεδομένων):

Object	Timestamp	Events
A	10	2, 3, 5
A	20	6, 1
A	23	1
B	11	4, 5, 6
B	17	2
B	21	7, 8, 1, 2
B	28	1, 6
C	14	1, 8, 7

Ταξινόμηση με βάση τη χρονόσημα -> ακολουθία



Γεγονότα (Events) (~στοιχεία) σχετιζόμενα με Αντικείμενα (Objects) (~δοσοληψίες) και πότε αυτά εμφανίζονται

Παραδείγματα Ακολουθιακών Δεδομένων

- Ακολουθία προσπελάσεων Web:
  - ‹ {Homepage} {Electronics} {Digital Cameras} {Canon Digital Camera} {Shopping Cart} {Order Confirmation} {Return to Shopping} ›
- Ακολουθία γεγονότων που οδήγησαν σε πυρηνικό ατύχημα στο 3-mile Island: (http://stellar-one.com/nuclear/staff_reports/summary_SOE_the_initiating_event.htm)
  - ‹ {clogged resin} {outlet valve closure} {loss of feedwater} {condenser polisher outlet valve shut} {booster pumps trip} {main waterpump trips} {main turbine trips} {reactor pressure increases}›
- Ακολουθία βιβλίων δανεισμού από βιβλιοθήκη:
  - ‹{Fellowship of the Ring} {The Two Towers} {Return of the King}›

Τυπικός Ορισμός Ακολουθίας

Μια ακολουθία (sequence) είναι μια *διατεταγμένη λίστα* από στοιχεία (elements) (~transactions)

$s = \langle e_1\ e_2\ e_3\ \dots \rangle$

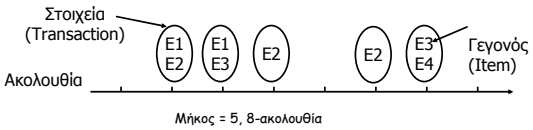
Κάθε στοιχείο αποτελείται από μια συλλογή από γεγονότα (events) (~items)

$e_i = \{i_1, i_2, \dots, i_k\}$

Κάθε στοιχείο αντιστοιχεί σε μια συγκεκριμένη χρονική στιγμή ή τοποθεσία

Μήκος (length) μιας ακολουθίας, $|s|$, είναι ο αριθμός των στοιχείων της ακολουθίας

Μια k-ακολουθία είναι μια ακολουθία που περιέχει k γεγονότα (items)



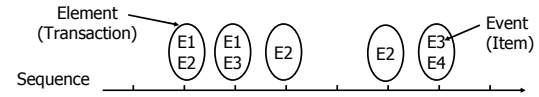
Μήκος = 5, 8-ακολουθία

Παραδείγματα Ακολουθιακών Δεδομένων

Sequence Database	Sequence	Element (Transaction)	Event (Item)
Customer	Purchase history of a given customer	A set of items bought by a customer at time t	Books, diary products, CDs, etc
Web Data	Browsing activity of a particular Web visitor	A collection of files viewed by a Web visitor after a single mouse click	Home page, index page, contact info, etc
Event data	History of events generated by a given sensor	Events triggered by a sensor at time t	Types of alarms generated by sensors
Genome sequences	DNA sequence of a particular species	An element of the DNA sequence	Bases A,T,G,C

Χρονική διάταξη

Χωρική διάταξη



Τυπικός Ορισμός Υπο-ακολουθίας

Μια ακολουθία $\langle a_1\ a_2\ \dots\ a_n \rangle$ περιέχεται σε μια άλλη ακολουθία (είναι υπο-ακολουθία της) $\langle b_1\ b_2\ \dots\ b_m \rangle$ ($m \geq n$) αν υπάρχουν ακέραιοι $i_1 < i_2 < \dots < i_n$ τέτοιοι ώστε $a_1 \subseteq b_{i1}$, $a_2 \subseteq b_{i2}$, …, $a_n \subseteq b_{in}$

Παραδείγματα

Ακολουθία Δεδομένων	Υπο-ακολουθία	Περιέχεται;
< {2,4} {3,5,6} {8} >	< {2} {3,5} >	Ναι
< {1,2} {3,4} >	< {1} {2} >	Όχι
< {2,4} {2,4} {2,5} >	< {2} {4} >	Ναι

Έστω D ένα σύνολο που περιέχει μια ή περισσότερες ακολουθίες.

Η υποστήριξη (support) μιας ακολουθίας w ορίζεται ως το ποσοστό των ακολουθιών στο D που *περιέχουν* το w

Σύνολο **5** ακολουθιών

Object	Timestamp	Events
A	1	1,2,4
A	2	2,3
A	3	5
B	1	1,2
B	2	2,3,4
C	1	1,2
C	2	2,3,4
C	3	2,4,5
D	1	2
D	2	3, 4
D	3	4, 5
E	1	1, 3
E	2	2, 4, 5

Παραδείγματα:

‹ {1,2} › s=60%

eline

10

Object A:

2
3
5

ct B:

4
5
6

ct C:

Εύρεση Ακολουθιακών Προτύπων

Έστω D ένα σύνολο που περιέχει μια ή περισσότερες ακολουθίες.

Η υποστήριξη (support) μιας ακολουθίας w ορίζεται ως το ποσοστό των ακολουθιών στο D που *περιέχουν* το w

Σύνολο 5 ακολουθιών

Object	Timestamp	Events
A	1	1,2,4
A	2	2,3
A	3	5
B	1	1,2
B	2	2,3,4
C	1	1, 2
C	2	2,3,4
C	3	2,4,5
D	1	2
D	2	3, 4
D	3	4, 5
E	1	1, 3
E	2	2, 4, 5

Παραδείγματα:

‹ {3}{5}› s=80%
‹{1,2}{5}› s=40%
‹{5}{1,2}› s=0%

Επίσης, οποιαδήποτε ακολουθία με μήκος μεγαλύτερο του 2

Εύρεση Ακολουθιακών Προτύπων

Ένα ακολουθιακό πρότυπο (*sequential pattern*) είναι μια συχνή υπο-ακολουθία (δηλαδή μια ακολουθία με υποστήριξη ≥ *minsup*)

Object	Timestamp	Events
A	1	1,2,4
A	2	2,3
A	3	5
B	1	1,2
B	2	2,3,4
C	1	1, 2
C	2	2,3,4
C	3	2,4,5
D	1	2
D	2	3, 4
D	3	4, 5
E	1	1, 3
E	2	2, 4, 5

Minsup = 50%

Παραδείγματα συχνών υπο-ακολουθιών

‹ {1,2} › s=60%
‹ {2,3} › s=60%
‹ {2,4}› s=80%
‹ {3} {5}› s=80%
‹ {1} {2} › s=80%
‹ {2} {2} › s=60%
‹ {1} {2,3} › s=60%
‹ {2} {2,3} › s=60%
‹ {1,2} {2,3} › s=60%

Εύρεση Ακολουθιακών Προτύπων

Ορισμός Προβλήματος Εξόρυξης Ακολουθιακών Προτύπων (Sequential Pattern Mining)

Είσοδος:
Μια βάση από ακολουθίες
Ένα ελάχιστο κατώφλι υποστήριξης, *minsup*

Πρόβλημα:
Βρες όλες τις υπο-ακολουθίες με υποστήριξη ≥ *minsup*

Εύρεση Ακολουθιακών Προτύπων

Έστω μια ακολουθία: ‹{a b} {c d e} {f} {g h i}›
Παραδείγματα υπο-ακολουθίας:
‹{a} {c d} {f} {g} ›, ‹ {c d e} ›, ‹ {b} {g} ›, etc
Ο αριθμός τους είναι εκθετικός

Πόσες k-υποακολουθίες μπορεί να εξαχθούν από μια n-ακολουθία;

‹{a b} {c d e} {f} {g h i}› n = 9

Παράδειγμα για k = 4

{a}{c}{f}{g}
{a}, {f}, {h i}
{c d e} {g}

$$\binom{n}{k} = \binom{9}{4} = 126$$

$$\sum_{i=1,9} \binom{9}{i} = 511$$

Εύρεση Ακολουθιακών Προτύπων

Brute-Force Μέθοδος

Απαρίθμηση όλων των πιθανών υπο-ακολουθιών και υπολογισμός της υποστήριξης τους

Εύρεση Ακολουθιακών Προτύπων

- Έστω n γεγονότα: $i_1, i_2, i_3, \ldots, i_n$
- Υποψήφιες 1-υπο-ακολουθίες:
  - $\langle\{i_1\}\rangle, \langle\{i_2\}\rangle, \langle\{i_3\}\rangle, \ldots, \langle\{i_n\}\rangle$
- Υποψήφιες 2-υπο-ακολουθίες:
  - $\langle\{i_1, i_2\}\rangle, \langle\{i_1, i_3\}\rangle, \ldots, \langle\{i_1\}\{i_1\}\rangle, \langle\{i_1\}\{i_2\}\rangle, \ldots, \langle\{i_n\}\{i_n\}\rangle$
- Υποψήφιες 3-υπο-ακολουθίες:
  - $\langle\{i_1, i_2, i_3\}\rangle, \langle\{i_1, i_2, i_4\}\rangle, \ldots, \langle\{i_1, i_2\}\{i_1\}\rangle, \langle\{i_1, i_2\}\{i_2\}\rangle, \ldots,$
  - $\langle\{i_1\}\{i_1, i_2\}\rangle, \langle\{i_1\}\{i_1, i_3\}\rangle, \ldots, \langle\{i_1\}\{i_1\}\{i_1\}\rangle, \langle\{i_1\}\{i_1\}\{i_2\}\rangle, \ldots$

Παρατήρηση

Ο αριθμός των υποψήφιων ακολουθιών είναι πολύ μεγαλύτερος από τον αριθμό των υποψήφιων στοιχειοσυνόλων για δύο κυρίως λόγους:

- Ένα στοιχείο μπορεί να εμφανιστεί μόνο μια φορά σε ένα στοιχειοσύνολο, ενώ ένα γεγονός μπορεί να εμφανιστεί περισσότερο από μια φορά σε μια ακολουθία
  Πχ το στοιχειοσύνολο {i1, i2} -> <{i1,i1}> <{i1},{i1}>, κλπ
- Η διάταξη δεν έχει σημασία στα στοιχειοσύνολα, αλλά έχει στις ακολουθίες
  Πχ το στοιχειοσύνολο {i1, i2} -> <{i1}{i2}> <{i2}{i1}>, κλπ

Apriori για ακολουθίες

Μια k-ακολουθία πρέπει να περιέχει όλες τις πιθανές k-1 υπο-ακολουθίες της

Οπότε, παρόμοιος αλγόριθμος

Generalized Sequential Pattern (GSP)

- **Βήμα 1**:
  - Κάνε το πρώτο πέρασμα στη βάση των ακολουθιών D και παρήγαγε όλες τις συχνές ακολουθίες ενός στοιχείου
- **Βήμα 2**:
  Επανέλαβε μέχρι να μην παράγονται νέες συχνές ακολουθίες
  - **Δημιουργία Υποψηφίων - Candidate Generation**:
    - Συγχώνευση συχνών ακολουθιών που βρέθηκαν στο (k-1)*th* πέρασμα φια δημιουργία υποψηφίων ακολουθιών με k στοιχεία
  - **Ψαλίδισμα Υποψηφίων - Candidate Pruning**:
    - Ψαλίδισε τις k-ακολουθίες που περιέχουν μη συχνές Prune candidate (*k-1*)-υπο-ακολουθίες
  - **Υπολογισμός Υποστήριξης - Support Counting**:
    - Κάνε ένα νέο πέρασμα στη βάση D για τον υπολογισμό της υποστήριξης των νέων υποψηφίων
  - **Υπολογισμός Υποψηφίων - Candidate Elimination**:
    - Διώξε τις υποψήφιες *K*-ακολουθίες που η πραγματική τους υποστήριξη είναι μικρότερη του minsup

Δημιουργία υποψηφίων

- Βάση (k=2):
  Συγχώνευση δύο συχνών 1-ακολουθιών <$\{i_1\}$> and <$\{i_2\}$> θα παράξει δυο υποψήφιες 2-ακολουθίες: <$\{i_1\}$ $\{i_2\}$> and <$\{i_1\ i_2\}$>
- Γενική περίπτωση (k>2):
  Συνθήκη για συγχώνευση: Μια συχνή (*k-1*)-ακολουθία w_1 συγχωνεύεται μα μια άλλη συχνή (*k-1*)-ακολουθία w_2 για να παραχθεί μια υποψήφια *k*-ακολουθία αν η υπο-ακολουθία που παίρνουμε αν *αφήσουμε το πρώτο γεγονός της* w_1 είναι το ίδιο με την υπο-ακολουθία που παίρνουμε αν *αφήσουμε το τελευταίο γεγονός της* w_2
  Το **αποτέλεσμα** μετά τη συγχώνευση είναι η ακολουθία w_1 επεκταμένη με το τελευταίο γεγονός της w_2.
  - Αν τα τελευταία δύο γεγονότα της w_2 ανήκουν στο ίδιο στοιχείο τότε το τελευταίο γεγονός της w_2 γίνεται μέρος του τελευταίου στοιχείου της w_1
  - Αλλιώς το τελευταίο γεγονός της w_2 γίνεται ένα διαφορετικό στοιχείο appended στο τέλος της w_1

Παραδείγματα Δημιουργίας Υποψηφίων

- Συγχώνευση των ακολουθιών
  w_1=<{ } {2 3} {4}> και w_2 =<{2 3} {4 □}>
  μας δίνει την υποψήφια ακολουθία < {1} {2 3} {4 5}> γιατί τα 2 τελευταία γεγονότα της w_2 (4 και 5) ανήκουν στο ίδιο στοιχείο
- Συγχώνευση των ακολουθιών
  w_1=<{ } {2 3} {4}> και w_2 =<{2 3} {4} { □}>
  μας δίνει την υποψήφια ακολουθία < {1} {2 3} {4} {5}> γιατί τα 2 τελευταία γεγονότα της w_2 (4 και 5) δεν ανήκουν στο ίδιο στοιχείο
- Δε χρειάζεται να συγχωνεύσουμε τις ακολουθίες
  w_1 =<{ } {2 6} {4}> και w_2 =<{1} {2} {4 □}>
  για να πάρουμε το υποψήφιο < {1} {2 6} {4 5}> συγχώνευση w_1 με <{2 6} {4 □}>

Παραδείγματα Δημιουργίας Υποψηφίων

Frequent 3-sequences

< {1} {2} {3} >
< {1} {2 5} >
< {1} {5} {3} >
< {2} {3} {4} >
< {2 5} {3} >
< {3} {4} {5} >
< {5} {3 4} >

Candidate Generation

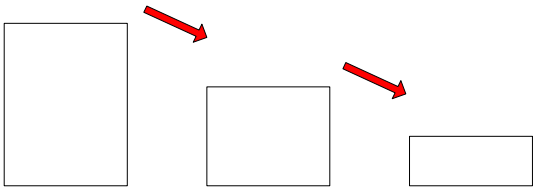
< {1} {2} {3} {4} >
< {1} {2 5} {3} >
< {1} {5} {3 4} >
< {2} {3} {4} {5} >
< {2 5} {3 4} >

Εύρεση Ακολουθιακών Προτύπων

Ψαλίδισμα υποψηφίων

Μια υποψήφια k-ακολουθία σβήνεται αν έχει τουλάχιστον μια μη συχνή (*k-1*)-υπο-ακολουθία

Εύρεση Ακολουθιακών Προτύπων



Τέλος