Apriori

A	B	C	D	E
1	0	0	1	0
1	0	0	1	1
0	0	1	0	0
1	0	1	1	1
1	0	1	0	1

Property 1: If an itemset S is large, then any proper subset of must be large.
Property 2: If an itemset S is NOT large, then any proper superset of S must Not be large.

item count
A 4
B 0
C 3
D 3
E 3
L = 3

L1 = {{A},{C},{D},{E}}
② counting step: L_2 = {{A,C},{A,D}, {A,E},{C,E}, {D,E}}

Large 2-itemset Generation (根据前缀区间)
① Join step / prune step
C_2 = {{A,C},{A,D},{A,E},{C,D},{C,E},{D,E}}

① Join step
C_3 = {{A,C,D},{A,C,E},{A,D,E}}
② prune step
C_3 = {{A,C,D},{A,C,E},{A,D,E}}
③ counting step (perfix)
L_3 = {{A,C,E},{A,D,E}} 前缀不相同,结束

TID	Items Bought
100	a, b, c, d, e, f, g, h
200	a, f, g
300	b, d, e, f, j
400	a, b, d, i, k
500	a, b, e, g

(ordered) Frequent Items

(Ordered) Frequent Items
a, b, d, e, f, g ①
a, f, g ②
b, d, e, f ③
a, b, d ④
a, b, e, g ⑤

FP-tree: Frequent Pattern Tree

to find all "large" (or frequent) itemsets with support at least a threshold (with support ≥ 3)

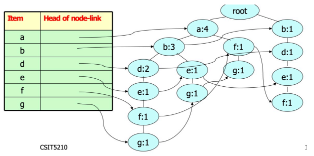
Item	Frequent
a	4
b	4
c	1
d	3
e	3
f	3
g	3
h	1
i	1
j	1
k	1

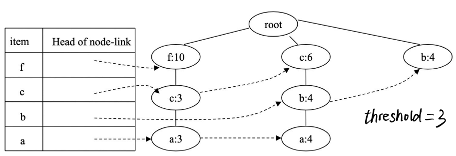
Support ≥ 3 ⟹

Item	Frequent
a	4
b	4
d	3
e	3
f	3
g	3

找到大于 threshold 的 item

建树





threshold = 3

Cond. FP-tree on "a" ('a':7)
{{f:3, c:3, a:3}, {c:4, b:4, a:4}}

item	frequent
a	7
c	7
b	4
f	3

得到 {a:3, c:3, f:3}, {a:4, b:4, c:4} (降序排列)

item	head of the node-link
c	7
b	4
f	3

root — C:7 — b:4, f:3

由于不是 single path, 继续递归

Cond. FP-tree on "af": 3
{{c:3, f:3}}

item	head
C	

root — C:3

Cond. FP-tree on "b": 8
{{c:4, b:4}, {b:4}}

item	head
c	

root — C:4 ; {b,c}: 4

Cond. FP-tree on "c": 9
{{f:3, c:3}, {c:6}}

item	head
f	

root — f:3 ; {f,c}: 3

Cond. FP-tree on "ab": 4

item	head
C	4

root — C:4

Cond. FP-tree on "f": 10
{{f:10}}
root

Cond. FP-tree on "ac": 7

item	head

root

frequent itemsets = {a}: 7, {a,f}: 3, {a,c,f}: 3, {a,b}: 4, {a,b,c}: 4, {a,c}: 7, {b}: 8, {b,c}: 4, {c}: 9, {c,f}: 3, {f}: 10

k-means clustering (original)

① 给出 k 个点的初始均值预测
② 把每个点分配给距离最近的 mean, 形成 cluster
③ 重新计算 mean, 重复 ②③ 直到 mean 不再改变 (收敛)

缺点: ① The result produced depend on the initial values for the means, and it happens that suboptimal partitions are found.
② 会出现没有点划分到初始预测的某群中

Sequential K-Means cluster

① 设立 k 个初始的假设, k 个 cluster 的数量 n_i 为 0
② 每次加入一个点后, 如果它属于某个 cluster, 就对该 cluster 的 mean 更新

$n_i = n_i + 1$ (某种中心点)
$m = m + \frac{1}{n_i}(x - m)$ (不同规模)

③ Forgetful Sequential K-means 越早加入 cluster 的点影响越小

$m_n = (1-a)^n m_0 + a \sum_{k=1}^{n} (1-a)^{n-k} x_k$ a 为遗忘系数

Hierarchical Clustering Methods

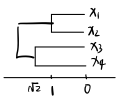
dendrogram
Distance between two cluster
① Single Linkage (最短距离)
② Complete Linkage (最长距离)
③ Group Averge Linkage (两个cluster之间两两平均距离)
④ Centroid Linkage (聚类中心之间的距离)
⑤ Median Clustering (取两个cluster的中心点中位数作为新的中心)

example: $x_1(1,2)$, $x_2(2,2)$, $x_3(5,2)$, $x_4(6,1)$

	x_1	x_2	x_3	x_4
x_1	0			
x_2	①	0		
x_3	4	3	0	
x_4	5√2	5	√2	0

(1.5, 2) (x_1, x_2), x_3, x_4

	(x_1x_2)	x_3	x_4
(x_1x_2)	0		
x_3	3.5	0	
x_4	√(85)/2	√2	0



Divisive Methods

M - 整块开始, 然后根据距离拆分小块
计算与其他点的距离, 将 D 最大的分离开 (重复为正数)
Monothetic = 值直接用
Chi - Squre Measure

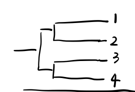
	A	B	C
1	1	1	0
2	1	1	0
3	1	0	1
4	0	0	1

B \ A	1	0
1	a=2	b=0
0	c=1	d=1

$$\chi^2_{AB} = \frac{(ad-bc)^2 N}{(a+b)(a+c)(b+d)(c+d)} = \frac{2^2 \times 4}{2\times3\times1\times2} = 1.33$$

Attr.	AB	AC	BC
a	2	1	0
b	0	1	2
c	1	2	2
d	1	0	0
N	4	4	4
χ^2	1.33	1.33	4

For Attr. A $\chi^2_{AB} + \chi^2_{AC} = 2.66$
For Attr. B $\chi^2_{AB} + \chi^2_{BC} = 5.33$
For Attr. C $\chi^2_{AC} + \chi^2_{BC} = 5.33$
所以根据 B 的取值 0, 1 将 1,2,3,4 分为 {1,2}, {3,4}.



Density-Based Clustering (DBSCAN)

基于密度的聚类方式 ε-neighborhood
给定 MinPts 和 ε,
N(o) = {a}, 如果点周围个数超过 MinPts, 为 Core points
① Core points ② border points ③ noise points
到所有一个core point

(11,13,5) X2 Y2 Z1
(12,11,21) X2 Y2 Z3
(11,17,27) X2 Y2 Z3
(13,14,28) X2 Y2 Z4
(22,37,36) X3 Y4 Z4
(24,31,27) X3 Y4 Z3
(25,35,21) X3 Y4 Z3
(29,34,4) X3 Y4 Z1
(35,5,4) X4 Y1 Z1
(36,6,5) X4 Y1 Z1

the grid size of each dimension is 10.
For size 1 density threshold 40% 10×4% = 4

For attr. X: x_1: 0, x_2: 4, x_3: 4, x_4: 2
For attr Y: Y_1: 2, Y_2: 4, Y_3: 0, Y_4: 4
For attr. Z: Z_1: 4, Z_2: 0, Z_3: 4, Z_4: 2

dense unit: $(x_2)(x_3)$ dense unit: $(Y_2)(Y_4)$ dense unit $(Z_1)(Z_3)$

For size 2
For attr XY
C_2 = {{x_2,Y_2}, {x_2,Y_4}, {x_3,Y_2}, {x_3,Y_4}}
L_2 = {{x_2,Y_2}, {x_3,Y_4}}
For attr. XZ
C_2 = {{x_2,Z_1}, {x_2,Z_3}, {x_3,Z_1}, {x_3,Z_3}}
L_2 = {}
. . . .
L_2 = {{X,Y}}

subspace containing dense unit = {{X}, {Y}, {Z}, {X,Y}}

(2) identify clusters in each subspace containing dense units
For attr. X dense unit {x_2},{x_3} cluster 1 X[11,30]
For attr. Y dense unit {Y_2,Y_4} cluster 1: Y[11-20] cluster 2: Y[31-40]
For attr. Z dense unit {Z_1,Z_3} cluster 1: Z[1-10] cluster 2: Z[21-30]
For attr. XY dense unit (x_2,Y_2)(x_3,Y_4) cluster 1 {X[11-20], Y[11-20]} cluster 2 {X[21-30], Y[31-40]}

Entropy

$H(X) = -\sum_{x \in A} p(x) \log p(x)$

$H(X_1,\ldots,X_n) = -\sum_{x_1 \in A_1} \cdots \sum_{x_n \in A_n} p(x_1,\ldots,x_n) \log p(x_1,\ldots,x_n)$

X/Y	1	2
1	3/8	1/8
2	1/8	3/8

$H(X) = -\frac{1}{2}\log\frac{1}{2} + (-\frac{1}{2}\log\frac{1}{2}) = 1$
$H(Y) = -\frac{1}{2}\log\frac{1}{2} + (-\frac{1}{2}) \log\frac{1}{2} = 1$

$H(Y|X) = -\sum_{x \in A}\sum_{y \in B} p(x,y) \log p(y|x)$

X \ Y	1	2
1	1/4	0
2	1/4	1/2

(a) Calculate the conditional entropy of H(X|Y).
(b) Calculate H(X|Y) as
$-\sum_{x \in A}\sum_{y \in B} p(x, y) \log p(x|y)$
where A = {1, 2} and B = {1, 2}.

$H(Y|X) = \sum_{x \in A} p(x) H(Y|X=x)$

$H(X,Y) = H(X) + H(Y|X)$

(a) $H(X|Y) = \frac{1}{2}H(X|Y=1) + \frac{1}{2}H(X|Y=2)$
$= \frac{1}{2} \times 1 + \frac{1}{2} \times 0 = \frac{1}{2}$

(b) $H(X|Y) = -\frac{1}{4}\log\frac{1}{2} + (-\frac{1}{4}\log\frac{1}{2}) + 0 + \frac{1}{2}\log 1 = \frac{1}{2}$

Consider three dimensions (X, Y, Z). There are the following 10 three-dimensional data points.

(11, 13, 5), (12, 11, 21), (11, 17, 27), (13, 14, 38), (22, 37, 36), (24, 31, 27), (25, 35, 21), (29, 34, 4), (35, 5, 4), (36, 6, 5).

Suppose each dimension ranges from 1 to 40.
Assume that the grid size of each dimension is 10. For example, dimension X has 4 grids or units, namely X1, X2, X3 and X4, where X1, X2, X3 and X4 correspond to [1, 10], [11, 20], [21, 30] and [31, 40], respectively.

Consider the entropy-based subspace clustering. Let the entropy threshold be 2.0.
Find all subspaces containing good clusters.

(11,13,5) X2 Y2 Z1
(12,11,21) X2 Y2 Z3
(11,17,27) X2 Y2 Z3
(13,14,28) X2 Y2 Z4
(22,37,36) X3 Y4 Z4
(24,31,27) X3 Y4 Z3
(25,35,21) X3 Y4 Z3
(29,34,4) X3 Y4 Z1
(35,5,4) X4 Y1 Z1
(36,6,5) X4 Y1 Z1

For attr. X: x_1: 0, x_2: 0.4, x_3: 0.4, x_4: 0.2
$H(x) = -[0.4\log 0.4 + 0.4\log 0.4 + 0.2\log 0.2] = 1.52$

For attr. Y: Y_1: 0.2, Y_2: 0.4, Y_3: 0, Y_4: 0.4
$H(Y) = -[0.2\log 0.2 + 0.4\log 0.4 + 0.4\log 0.4] = 1.52$

For attr. Z: Z_1: 0.4, Z_2: 0, Z_3: 0.4, Z_4: 0.2
$H(Z) = -[0.4\log 0.4 + 0.4\log 0.4 + 0.2\log 0.2] = 1.52$

L_1 = {{X}, {Y}, {Z}}

For Size 2
C_2 = {{X,Y}, {X,Z}, {Y,Z}}
For attr. XY

	Y_1	Y_2	Y_3	Y_4
X_1				
X_2		0.4		
X_3				0.4
X_4	0.2			

$H(X,Y) = -[0.2\log 0.2 + 0.4\log 0.4 + 0.4\log 0.4] = 1.52$

For attr. XZ

	Z_1	Z_2	Z_3	Z_4
X_1				
X_2	0.1		0.2	0.1
X_3	0.1		0.2	0.1
X_4	0.2			

$H(X,Z) = -[0.1\log 0.1 + 0.1\log 0.1 + \ldots + 0.2\log 0.2] = 2.72$

. . . . L_2 = {X,Y}
Subset containing all good clustering
{{X}, {Y}, {Z}, {X,Y}}

Naive Bayes Classifier

$$P(A|B) = \frac{P(A,B)}{P(B)} = \frac{P(B|A) \cdot P(A)}{P(B)}$$

$P(X,Y,Z|A) = P(X|A) \cdot P(Y|A) \cdot P(Y|Z)$

判断

Race	Income	Child	Insurance
White	high	no	?

P(Yes | Race=white, Income=high, child=no)
P(No | Race=white, Income=high, child=no)
= P(Race=white, Income=high, child=no | Yes)
P(Race=white, Income=high, child=no)

分母出 child 就是看 child=no 的选项

Decision Tree

$H(X) = -p(x)\log p(x)$

① ID3 $Gain(A,T) = Info(T) - Info(A,T)$
② C4.5 $Gain(A,T) = (Info(T) - Info(A,T)) / SplitInfo(A)$
③ CART $(Info(T) = 1 - \sum_j p_j^2$ 用这公式代替 H(X))

$SplitInfo(A) = -\sum_{v \in A} P(v) \log P(v)$

Race	Income	Child	Insurance
B	H	N	Y
W	H	Y	Y
W	L	Y	Y
W	L	Y	Y
B	L	N	N
B	L	N	N
B	L	N	N
W	L	N	N

For attr. child Info(T) = 1
$Info(T_{no}) = -\frac{1}{5}\log\frac{1}{5} + \frac{4}{5}\log\frac{4}{5} = 0.72$
$Info(T_{yes}) = 0$
$Info(child, T) = \frac{5}{8}\times 0.72 + \frac{3}{8}\times 0 = 0.45$
Gain(child, T) = 0.55 (ID3)
$SplitInfo(child) = -[\frac{3}{8}\log\frac{3}{8} + \frac{5}{8}\log\frac{5}{8}] = 0.95$
Gain(child, T) = 0.58 (C4.5)

Bayes Belif Network

① 求单种情况 P(HD)
P(HD) = P(HD | A=Y, B=Y) P(A=Y, B=Y)
+ P(HD | A=Y, B=N) P(A=Y, B=N)
+ ... } 4种情况

② 由子求父的情况 HD → BP
$$P(HD=Yes | BP=Yes) = \frac{P(BP=Yes | HD=Yes) \cdot P(HD=Yes)}{P(BP=Yes)}$$
P(BP | Yes) 由 ① 中方法得到

③ D E → HD → BP 上下都有
$$P(A|B,D,E) = \frac{P(B|A,D,E)\,P(A|D,E)}{P(B|D,E)}$$
P(HD=Yes | BP=Yes, D=Yes, E=Yes)
$$= \frac{P(BP | HD=Yes, D=Yes, E=Yes)\, P(HD=Yes | D=Yes, E=Yes)}{P(BP | D=Yes, E=Yes)}$$
$$= \frac{P(BP | HD=Yes, D=Yes, E=Yes)\, P(HD=Yes | D=Yes, E=Yes)}{\sum_{x \in \{Yes, No\}} P(BP=Yes | HD=x) \cdot P(HD=x | D=Yes, E=Yes)}$$

比较 P(Yes) ... 和 P(No) ... 哪个大

E = Yes, BP = High, CP = Yes (例题)

E → HD → BP, CP

P(HD=Yes | E=Yes, BP=High, CP=Yes)
$$= \frac{P(BP=High, CP=Yes | HD=Yes, E=Yes) \cdot P(HD=Yes | E=Yes)}{P(BP=High, CP=Yes | E=Yes)}$$
$$= \frac{0.85 \times 0.8 \times 0.25}{P(BP=H, CP=Y | HD=Yes) \cdot P(HD=Yes, E=Yes) + P(BP=H, CP=Y | HD=No) \cdot P(HD=No, E=Yes)}$$
$$= \frac{0.85 \times 0.8 \times 0.25}{0.85 \times 0.8 \times 0.25 + 0.2 \times 0.6 \times 0.45}$$

Data Cube

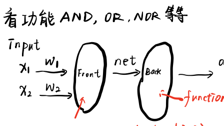
Gain({view for "sc", top view}, {top view})
从这两个视角 "sc", top view 比 只从 top view 所节省的时间

Greedy Algorithm
即在上面的基础之间选取最大的

	1st Choice	2nd Choice	3rd Choice
a			
b	50×5 = 250		
c	25×5 = 125	25×2 = 50	25×1 = 25
d	80×2 = 160	30×2 = 60	30×2 = 60
e	70×3 = 210	20×3 = 60	20×2+10 = 50
f	60×2 = 120	60+10 = 70	
g	99×1 = 99	49×1 = 49	49×1 = 49
h	90×1 = 90	40×1 = 40	30×1 = 30

Neuron Network

看功能 AND, OR, NOR 等等



net = $w_1x_1 + w_2x_2 + b$ (b = bias)

activate function
① Threshold function / Step function, Harder function
$y = \begin{cases} 1 & \text{if } net \ge 0 \\ 0 & \text{if } net < 0 \end{cases}$
② Linear function / Identity function
y = net
③ Rectifier Function / ReLU
$y = \begin{cases} net & \text{if } net \ge 0 \\ 0 & \text{if } net < 0 \end{cases}$
④ sigmoid Function
$y = \frac{1}{1+e^{-net}}$
⑤ tanh Function
$y = \frac{e^{2net} - 1}{e^{2net} + 1}$

(x_1,x_2) = (00)
(x_1,x_2) = (01)
(x_1,x_2) = (10)
(x_1,x_2) = (11)
(x_1,x_2) = (00)

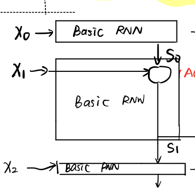
① $net = w_1x_1 + w_2x_2 + b = 0.1$
y = 1
$w_1 = w_1 + \alpha(d-y)x_1 = 0.1$
$w_2 = w_2 + \alpha(d-y)x_2 = 0.1$
$b = b + \alpha(d-y) = -0.4$
⋮

x_1	x_2	d
0	0	0
0	1	0
1	0	0
1	1	1

(AND)

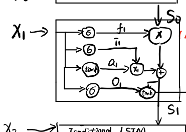
RNN

① Basic RNN ② Traditional LSTM ③ GRU



$S_t = \tanh(W \cdot [X_t, S_{t-1}] + b)$
$y_t = S_t$
Error = $y_t - y$

LSTM



$f_t = \sigma(W_f[X_t, y_{t-1}] + b_f)$
$I_t = \sigma(W_I[X_t, y_{t-1}] + b_I)$
$a_t = \tanh(W_a[X_t, y_{t-1}] + b_a)$
$S_t = f_t \cdot S_{t-1} + I_t \cdot a_t$
$O_t = \sigma(W_o[X_t, y_{t-1}] + b_o)$
$y_t = O_t \cdot \tanh(S_t)$
$\sigma(x)$ = sigmoid function
Error = $y_t - y$ (原数据)

GRU

$r_t = \sigma(W_r[X_t, y_{t-1}] + b_r)$
$a_t = \tanh(W_a[X_t, r_t \cdot y_{t-1}] + b_a)$
$u_t = \sigma(W_u[X_t, y_{t-1}] + b_u)$
$y_t = (1-u_t) \cdot y_{t-1} + u_t \cdot a_t$
Error = $y_t - y$

DataStream

FP = False Positive (Frequent item 中不正确)
FN = False Negative (Infrequent item 中不正确的)

ε - deficient synosis

满足三个条件

① 没有 fn (看看三个表中 Infrequency items 是否有不对的
② 估计的frequency与真实的frequency的差别不超过 εN —— 针对 Frequency (不同Algo 与 ideal algo 而比较)
③ 所有的项目中真实frequency小于 (S-ε)N 的都被判定为 Infrequency item. → 针对 Ideal Algo 的 Infrequency

Sticky Sampling algorithm.

$t = \lceil 1/\varepsilon \ln(s^{-1}\delta^{-1}) \rceil$

$S = 0.5$ $\varepsilon = 0.35$

$\delta = 0.5$

Data No.	r (Sampling rate)
1 ~ 2t	1
2t+1 - 4t	2
4t ~ 8t	4

(ii) $I_4, I_1, I_3, I_2, I_2, I_1, I_4, I_1$

the storage is

I_4 : 2 I_2 : 2
I_1 : 3
I_3 : 1

(iii) ① $r_1 = 2$ new sampling rate

② updates items in the storage in ascending order of item no.

For item I_1

$(I_1, 3) \xrightarrow{T} (I_1, 2) \xrightarrow{H} (I_1, 2) \rightarrow$ done

For item I_2

$(I_2, 2) \xrightarrow{H} (I_2, 2)$

For item I_3

$(I_3, 1) \xrightarrow{T} (I_3, 0) \rightarrow$ delete

For item I_4

$(I_4, 2) \xrightarrow{T} (I_4, 1) \xrightarrow{T} (I_4, 0) \rightarrow$ delete

result = $(I_1, 2)$ $(I_2, 2)$

(IV) I_1, I_4, I_5, -

假定第一个插入未成功

则 result = $(I_1, 3)(I_2, 2)(I_5, 1)$

(C) $y = 32 - 16 = 16$

$r_2 = 4$

(iii) $\varepsilon N = 0.35 \times 16 = 5.6$ $SN = 0.5 \times 16 = 8$

output = Get a list of items where $f + \varepsilon N \geq SN$

$(I_1, 5), (I_2, 4), (I_3, 2)$

∴ result = (I_1, I_2)

S: empty list
→ will contain (e, f)
Estimated frequency

When data e arrives,
- If e exists in S, increment f in (e, f)
- If e does not exist in S, add entry (e, 1) with prob. 1/r (where r: sampling rate)

Just after r changes,
- For each entry (e, f),
  - Repeatedly toss a coin with P(head) = 1/r until the outcome of the coin toss is head
  - If the outcome of the toss is tail,
    - Decrement f in (e, f)
  - If f = 0, delete the entry (e, f)

[Output] Get a list of items where
$f + \varepsilon N >= sN$

Lossing counting Algorithm

$S = 0.5$ $\varepsilon = 0.25$

$I_4, I_1, I_3, I_2, I_2, I_1, I_4, I_1, I_1, I_4, I_5, I_7, I_4, I_9$ ...

(a) $w = \lceil \frac{1}{\varepsilon} \rceil = 4$

(b)(i) D:

	$f + \Delta$
$(I_4, 1, 0)$	1
$(I_1, 1, 0)$	1
$(I_3, 1, 0)$	1
$(I_2, 1, 0)$	1

(ii) Since $f + \Delta$ of all entries ≤ 1

remove all entries

∅

(c)(i)

	$f + \Delta$
$(I_2, 1, 1)$	2
$(I_1, 2, 1)$	3
$(I_4, 1, 1)$	2

$b_{current}$ 为第几批

D $(I_1, 2, 1)$

(e) $\varepsilon N = 0.25 \times 12 = 3$

$SN = 0.5 \times 12 = 6$

∴ I_1 is output

(d)(i)

	$f + \Delta$
$(I_1, 3, 1)$	4
$(I_4, 1, 2)$	3
$(I_5, 1, 2)$	3
$(I_7, 1, 2)$	3

$(I_1, 3, 1)$

D: Empty set
- Will contain (e, f, Δ)

When data e arrives,
- If e exists in D,
  - Increment f in (e, f, Δ)
- If e does not exist in D,
  - Add entry (e, 1, $b_{current}$-1)

Remove some entries in D whenever
N ≡ 0 mod w
(i.e., whenever it reaches the bucket boundary)
The rule of deletion is:
(e, f, Δ) is deleted if
$f + \Delta <= b_{current}$

[Output] Get a list of items where
$f + \varepsilon N >= sN$

Space Saving

S = 0.5

data stream

$I_4, I_1, I_3, I_2, I_2, I_2, I_1$

1. D: Empty set
   - Will contain (e, f, Δ) Max. possible error in
2. $p_e = 0$
3. When data e arrives,
   - If e exists in D,
     - Increment f in (e, f, Δ)
   - If e does not exist in D,
     - If the size of D = M
       - $p_e \leftarrow \min_{e \in D} \{f + \Delta\}$
       - Remove all entries e where $f + \Delta \leq p_e$
     - Add entry (e, 1, p_e)
4. **[Output]** Get a list of items where
   $f + \Delta >= sN$

(a) 10-th

$(I_4, 3, 0)$
$(I_1, 4, 0)$
$(I_3, 1, 0)$
$(I_2, 2, 0)$

$P_e = 0$

(b) 加入 I_5 后 D ≥ m 3, 就删除

$P_e \leftarrow 1$

remove $(I_3, 1, 0)$

$(I_4, 3, 0)$
$(I_1, 4, 0)$
$(I_2, 2, 0)$
$(I_5, 1, 1)$

$P_e = 1$

(C) find $\min\{f + \Delta\}$

$(P_e \leftarrow 2$

$(I_4, 3, 0)$
$(I_1, 4, 0)$
$(I_7, 1, 2)$

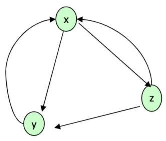
$P_e = 2$.

(d) $SN = 0.5 \times 12 = 6$

no output

HITs Algorithm

Authority   hub



normalized.

adjacency matrix

	x	y	z
x	0	1	1
y	1	0	0
z	1	1	0

Auth(x) = Hub(y) + Hub(z)
Auth(y) = Hub(x) + Hub(z)
Auth(z) = Hub(x)

$$\begin{pmatrix} h(x) \\ h(y) \\ h(z) \end{pmatrix} = M \begin{pmatrix} a(x) \\ a(y) \\ a(z) \end{pmatrix}$$

$\vec{h} = M\vec{a} = MM^T\vec{h}$

$\vec{a} = M^T\vec{h} = M^TM\vec{a}$

Hub(x) = Auth(y) + Auth(z)
Hub(y) = Auth(x)
Hub(z) = Auth(x) + Auth(y)

Google PageRank

	x	y	z
x	0	1	0.5
y	0.5	0	0.5
z	0.5	0	0

(M)

① $\vec{r} = M\vec{r}$

② 改进 (防止自环)

$\vec{r} = 0.8 \times M\vec{r} + \vec{c}$

$\vec{c} = \begin{pmatrix} 0.2 \\ 0.2 \\ 0.2 \end{pmatrix}$