7/12/22

Query Optimization : কোন একটা query কে কত দ্রুত answer করা যায়, তাই নির্ণয় করা- query তে কোন operation এ বেশি সময় যায়, তার উপায়। যেমন: disk read - write.

- যত কম প্রয়োজন HDD block থেকে load করতে হবে, তত কম সময় লাগবে।

- Linear search করলে O(n) টা block খুজতে হবে, Binary search করলে log(n).

- যদি primary key বা candidate একটা column বরাবর খুঁজি, তখন

P.T.O.

$O(\log_2 n)$ time লাগবে

- Primary index এর clustering property থাকে। Secondary index এর নাই, এর মানে PI এর ফেরে খুব্বা

→ Primary index কিন্তু primary key না, এক্ষেত্রে কয়টা block খুঁজতে হবে?

- n_r = number of tuple
- b_r = " " blocks
- f_r = elements in a single block

- আমার কার্যপ্রণালী করছে, প্রমন ক'টা block আছে— তা কেমন বুঝব?

Solⁿ:

100 টা ব্যাংক 1,000,000 মানুষ

∴ 1 " " আছে 100,000 মানুষ

~~X~~

$V(A, r)$ = R relation এ A কতবার unique আছে।

$Sc(A, r) = \left\lceil \frac{n_r}{V(A, r)} \right\rceil$; $b_r = \left\lceil \frac{n_r}{f_r} \right\rceil$

↙
selection Cardinality

যদি binary search করি,

$\log(b_r) + \left\lceil \frac{Sc}{f_r} \right\rceil - 1$

↙
pointer is always at 1st point

Math : $f_{ac} = 20$

$V(b.name, ac) = 50$ (unique bname এর 50 টা)

$V(bal, ac) = 500$ (unique bal এর 500 টা)

$n_{ac} = 10,000$

Select bname = "P" from ac;

কয়টা ব্লক ঘোরা লাগবে এই query'র জন্য?

Soln :

$\frac{10000}{50} \times 20$

$\frac{10,000}{50} = 200$ data tuple

$\frac{200}{20} = 10$ blocks

$\therefore ans = 10 + \log_2\left(\frac{10000}{20}\right) = 10 + 9 = 19$ block ঘোরাতে হবে

• equality on key (?) এর primary key.

B+ tree এর, Fanout, F = 20

$\log_{\frac{F}{2}}\left(\frac{50,000}{20}\right)$ + $\frac{20}{20}$ + $\frac{10,000/500}{20}$

= 2 + 1 + 10

equality on key

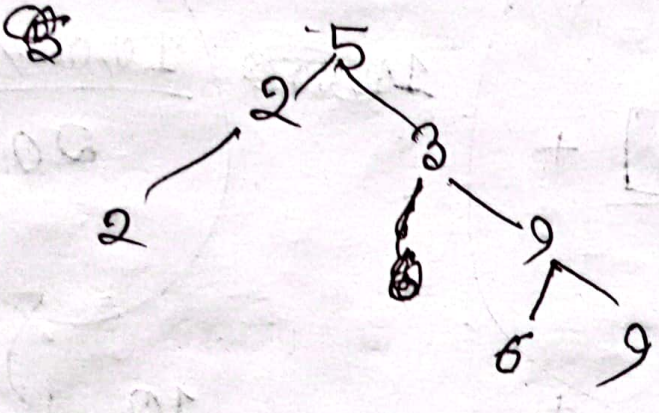
SC R

Background : Indexing এর ধারনা।

K-dimensional tree :

Database এর জন্য Indexing algorithm খুব দরকারি। যেমন: B+ tree.

→ Binary tree : 5, 2, 3, 2, 9, 6, 9



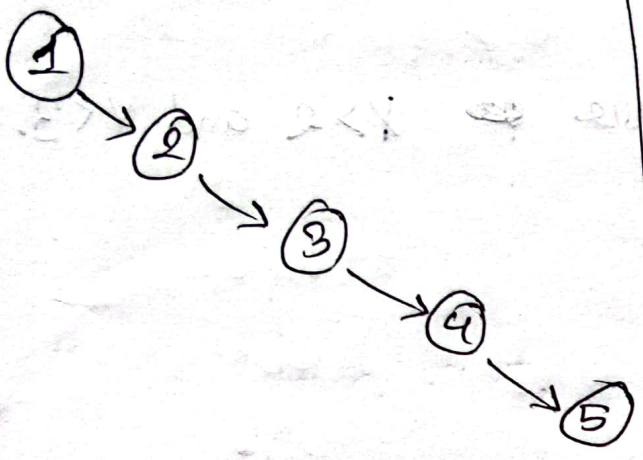
· Left এ ছোট
· Right এ বড়

এর অসুবিধা হলো :

1) Repeated data ভালভাবে handle করতে পারে না। (তবে এ অসুবিধা সমাধান counter ব্যবহার করা যেতে পারে।)

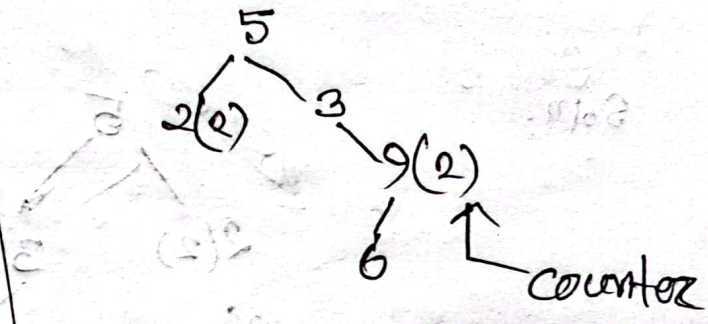
2) এটা balanced না হলে worst case এ it gets reduced to linear search.

Insert : 1, 2, 3, 4, 5

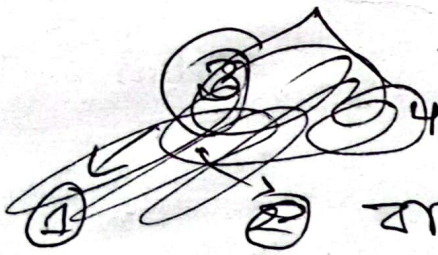


Using Counter :

5, 2, 3, 2, 1, 6, 1



এখানে, দেখা যাচ্ছে যে, যদি root value টা ঠিক মাঝে নেয়া যায়, তাহলে tree টা balanced হবে। এবং এতে দ্বিতীয় সমস্যার সমাধান হবে। [মানে median value কে root বানালে মোটামুটি balanced tree পাওয়া যাবে।]



1, 2, (4), 4, 10

median

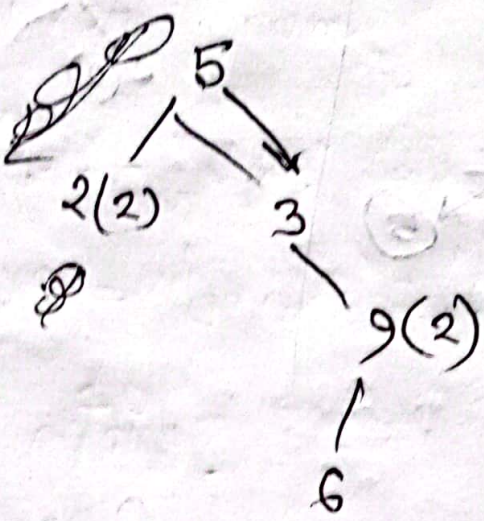
smaller bigger

Range

Query in Tree:

Select * from table where X > 2 and X < 3.

Soln:



Range query with multiple values, (X, Y):

दूसरे column

→ swap X, Y

↳ Grid

13/12/21

- T(webpage) = webpage Table

  V(Occurs, Url) = table এ Occur url আছে 10^6 টা।

- index - join :

i	j
1	1
2	2

$\bowtie$

- Index - Lookup : index অনুযায়ী B+ tree বানানো। তাই index ধরে search দিতে হবে।

  কিন্তু index কি?

- Table Join :
  - $\bowtie$: natural join
  - $\times$: Cartesian
  - ⟕ : Left outer join
  - ⟖ : Right Outer Join

• Left outer join কি?

- index - join করলে | A table এর url = B table এর url

- Query : page 9

Select author from
where author = "John" and
language = 'French' ;

Tables
Webpage
Occur
Word

কয়টা query লাগবে?

Soln. কিভাবে বিচার করি,

- Select author from webpage where author = John

Webpage	Occur	Word
• author • index • url	• url • index • url	• language • index

→ ans : = $\frac{10^9}{10^7}$ = 100 টা tuple

- Select author from record where language = "French"

Soln :

$$\frac{106}{100}$$

- Relational Algebra :

select author from webpage, Author where author = "John" and webpage.url = author.url.

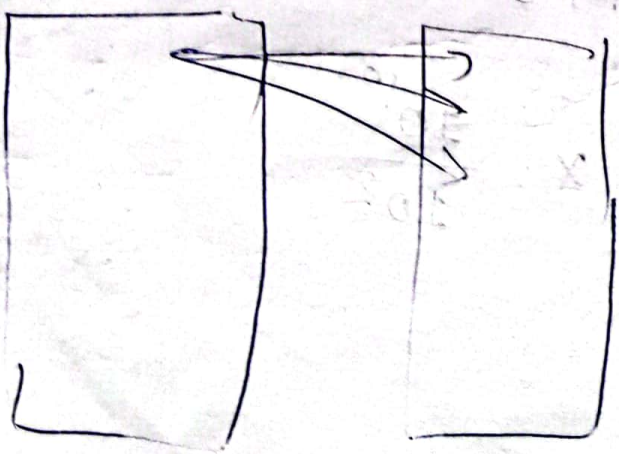
$$100 \text{ John} \times \frac{10^{10}}{10^{9}}$$

• main-memory-hash-join : Main memory তে কোন block নাই, তাই block access time ও নাই।

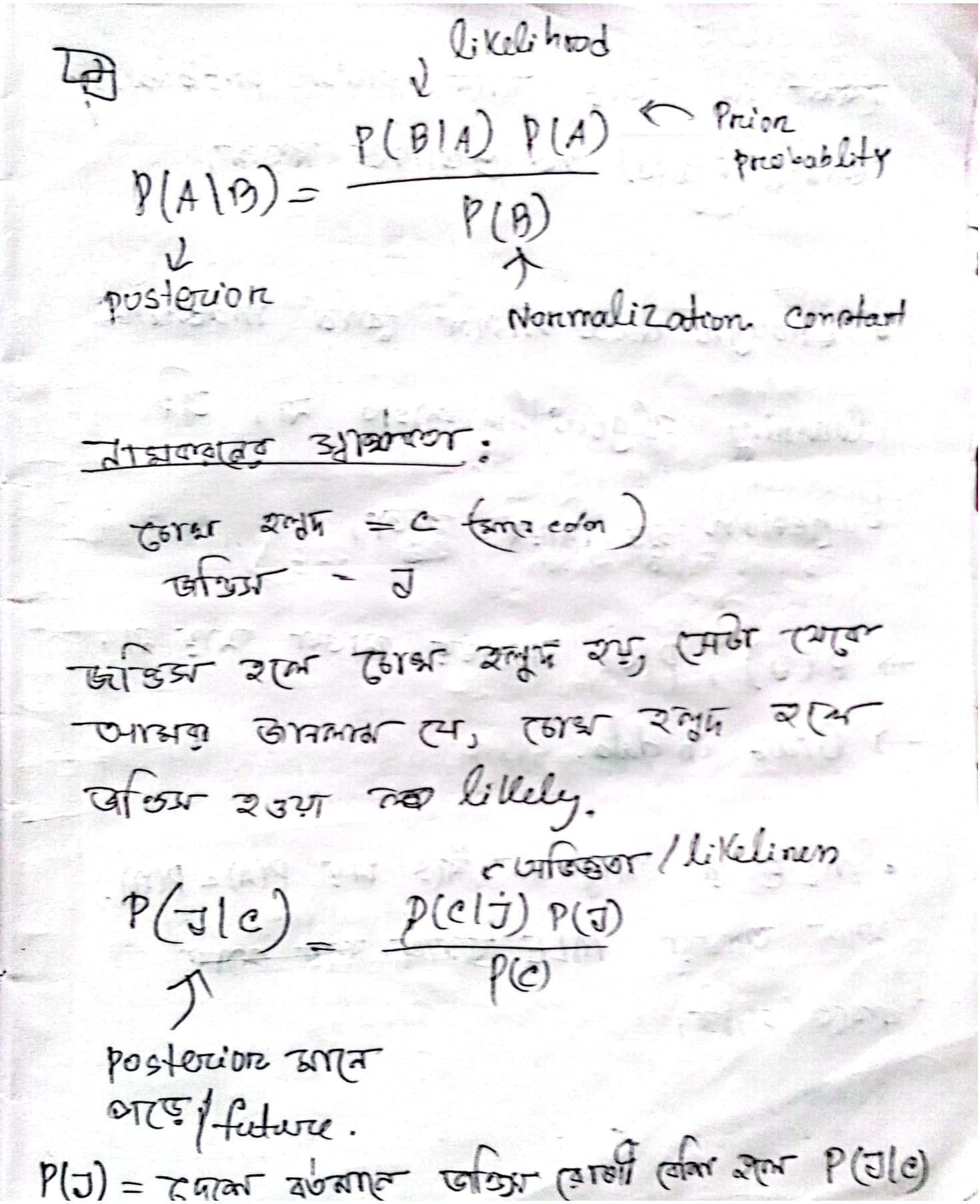
• select language from word where language = "French"

ans: $\frac{10^6}{100} = 10^4$

Final: ~~100~~ $100 \times \frac{10^3}{10} \times 10^4$



$10^0 + 10^2 \cdot 10^3 + 10^4 + 0 = ans.$



বায়েসের উপাত্তন:

চোখ হলুদ = C (Yellow eyes)

জন্ডিস = J

জন্ডিস হলে চোখ হলুদ হয়, সেটা থেকে আমরা জানলাম যে, চোখ হলুদ হলে জন্ডিস হওয়া কতো likely.

$$P(J|C) = \frac{P(C|J)\,P(J)}{P(C)}$$

C জন্ডিসের / likelihood

$P(J|C)$ ↑ posterior মানে পরে / future.

$P(J)$ = কোনো ব্যক্তির জন্ডিস হওয়ার সম্ভাবনা বলা হলে $P(J|C)$

বেশি হয়, তাই মানে, prior probability আমার $P(J|C)$ কে প্রভাবিত করবে,

- enough data থাকলে তবে machine learning algorithm লাগাত না, এই bayesian সূত্র দিয়েই হতো,

→ $P(J)$, $P(C)$ আগের থেকে জানা যায় না,

→ Bias in data.

- MLE র অনুমান করে নেয় $P(A) = P(B)$. মানে আমরা MLE ব্যবহার করে ~~করে~~ যাকি করতে পারি,

Skin detection:

$$T < \frac{P(c|s)}{P(c|ns)} \approx \frac{P(s|c)}{P(ns|c)}$$

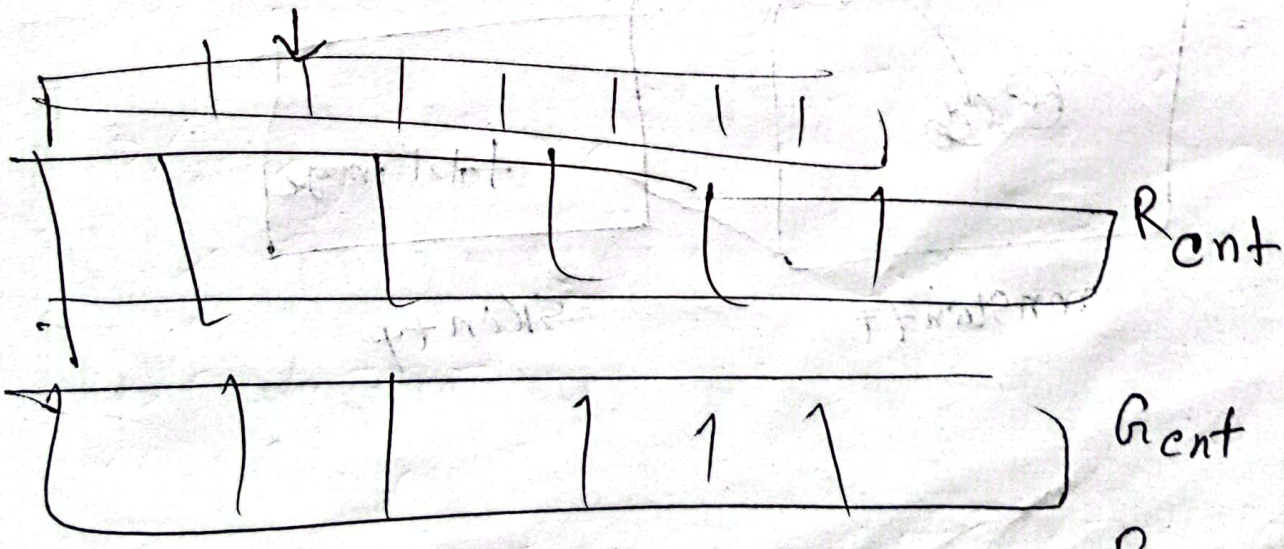
0.4

$$\frac{p(\text{skin হবার সম্ভাবনা} / \text{color})}{p(\text{No skin} \quad | \text{ color})} \rightarrow T$$

255^3 possible colors

Process:

Read image (RGB 0–255)



Cnt rgb

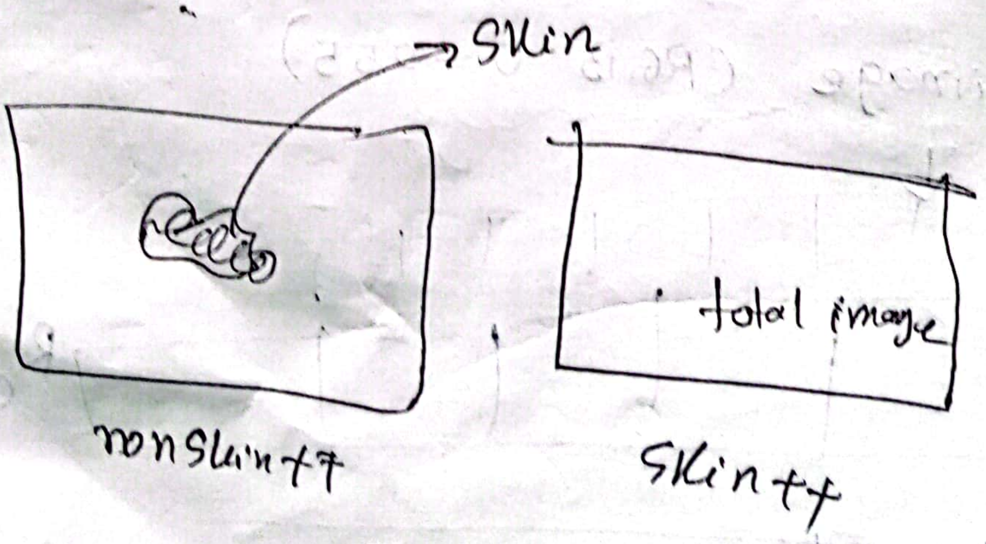
. ছবিতে বলা আছে . কোনটা skin, কোনটা non-skin.

↳ Mask কিভাবে বোঝে, ?

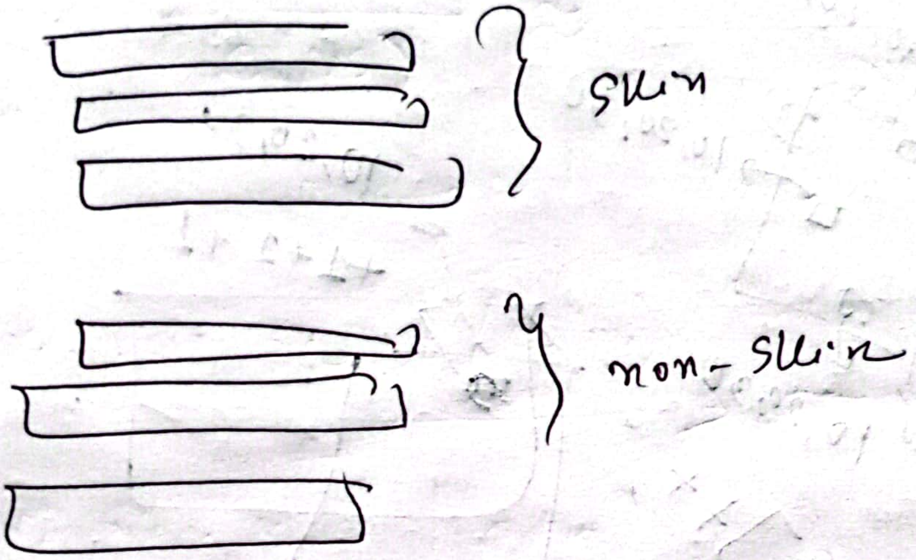
. আমরা করতে পারবো 223 225 >

⇕

equivalent file.



long double; ২০-২৫% বেশি হলে



⇓

Likelihood Ratio (২০-২৫%)

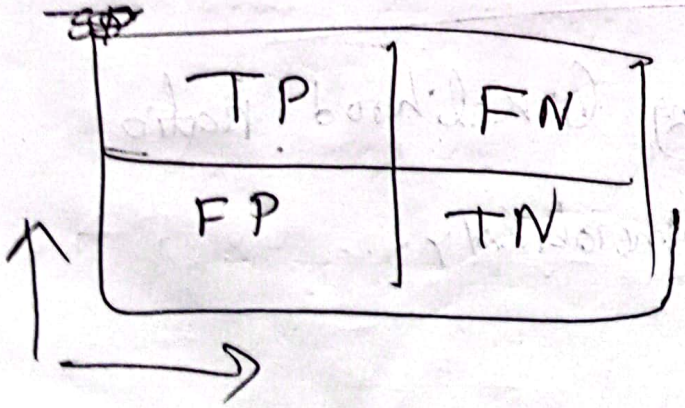
Test : image আসবে → color আসবে →
ওই color এর জন্য প্রাপ্য-ফলাফল হবে,

20/12/22

$$\frac{P(C|S)}{P(C|NS)} = \text{likelihood estimator}$$

Cross-validation :

- 10 fold CV তে dataset কে 10 ভাগ করা হয়।
তারপর Randomly 9 ভাগ নিয়ে train করব, 1 ভাগ নিয়ে test করব।
এভাবে 10 বার করব।



confusion matrix

#10 cv → accuracy → avg. accuracy
→ Confusion matrix → avg. confusion matrix

- Task : UCI machine learning repository → classification
  ↳ 2 class problem এর যেকোন dataset নিবে। UCI থেকে।
  করবে সেখানে Accuracy, confusion matrix বানাবে।

ছবি এর negative বানাবে → FN

23/12/22

A-Priori Algorithm:

- যদি কোনো profit maximizing and data correlation.

DataMining Book 2nd edition.pdf
page 236/265 ★★

↳ All possible combination এর frequency বের করব, যে combination এর frequency বেশি, সেগুলোকেই কোনো ব্যবহার।

But complexity is n!

ঘ.

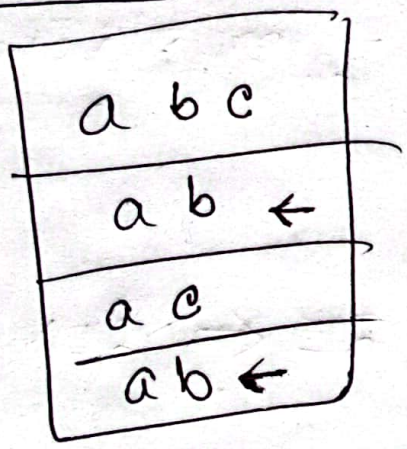
দোকানে 5 টা item থাকলে সবগুলো

Total algorithm Complexity = $5! = 120$

যদি 100 টা থাকে, তবে $100!$ তখন

অনেক হবে।

Possible Solution:



unique

a, b, c

a
b
c
ab
ac
bc
:

abc	1
ab	2
ac	1

Book Solution

- Support Count.

- Satisfies the customer. Its not about picking the absolute max profit.

↳ Association prior Knowledge.

. confidence, $A \Rightarrow B$ মানে, A কিনলে আর B ও কিনবে, তার নিশ্চয়তা কতো।

. $P(A \Rightarrow B) = \frac{SC(A \cup B)}{SC(A)}$ → SC = support count

$= \frac{\text{একসাথে কিনা}}{\text{শুধু A কিনা}}$

∴ A কেনায় আর B কেনার নিশ্চয়তা = $\frac{\text{একসাথে A, B কিনা}}{\text{শুধু A কিনা}}$

→ Confidence vs Probablity ঃ

সম্ভাব্য হচ্ছে, $\sum P = 1$

$\sum C \neq 1$

→ Likelihood vs Probablity

Likelihood হচ্ছে অতীতের ঘটনা থেকে আজকে

Probablity হচ্ছে Ponterion.

#

$I1 \wedge I2 \Rightarrow I5 = \frac{s\_c(I1 \wedge I2 \Rightarrow I5)}{s\_c(I1 \wedge I2)}$

$I1\,I2^*, I5^*, I1\,I2\,I5 = 4+$

$I1, I2^*$

$\Rightarrow sc(I1$

$I1 \wedge I5 \Rightarrow I2 = \frac{(I1 \wedge I5 \wedge I2) \geq c}{(I1 \wedge I5) \geq c}$

$= \frac{2}{2} = 100\%$

∴ We have 100% confidence on $I1 \wedge I5 \Rightarrow I2$

Coding approach :

[sc, length,

(a) File data] → algorithm → File output

(book example)

(b) generic N. + সকল query র জন্য confidence %. calculate করতে হবে।

Recovery System

We try to make our system as volatile as possible.

- having multiple copy in diff geographical location
- Part by part

But

- Defer মানে delay করা, Defer করার in database.

ACID:

ACID = Atomicity, Consistency, Isolation, Durability

Atomicity: কোন একটা transaction পুরাটা হবে, নাহলে হবেই না।

A gives B 50.
B gets 50 from A } both একসাথে।

Consistency: দিনশেষে আয় ব্যয় এর হিসাব মিলতে হবে। (accounting এর মতো)

Isolation: ৩-৪ টি transaction, parallely চলে যে output আসে, ঠিক সেরকম ভাবে serially চালালেও একই output আসে,

"A ; B কে 100 টাকা দিবে এবং C কে A এর 10% দিবে" — একে parallely করা সম্ভব না।

Durability: কোন সমস্যা হলেও database ঠিক থাকবে।

- Redo is idempotent operation. কারণ
  ↳ বারবার 950 ই বসবে।
  ↳ one commit, one redo.

Immediate vs Deferred

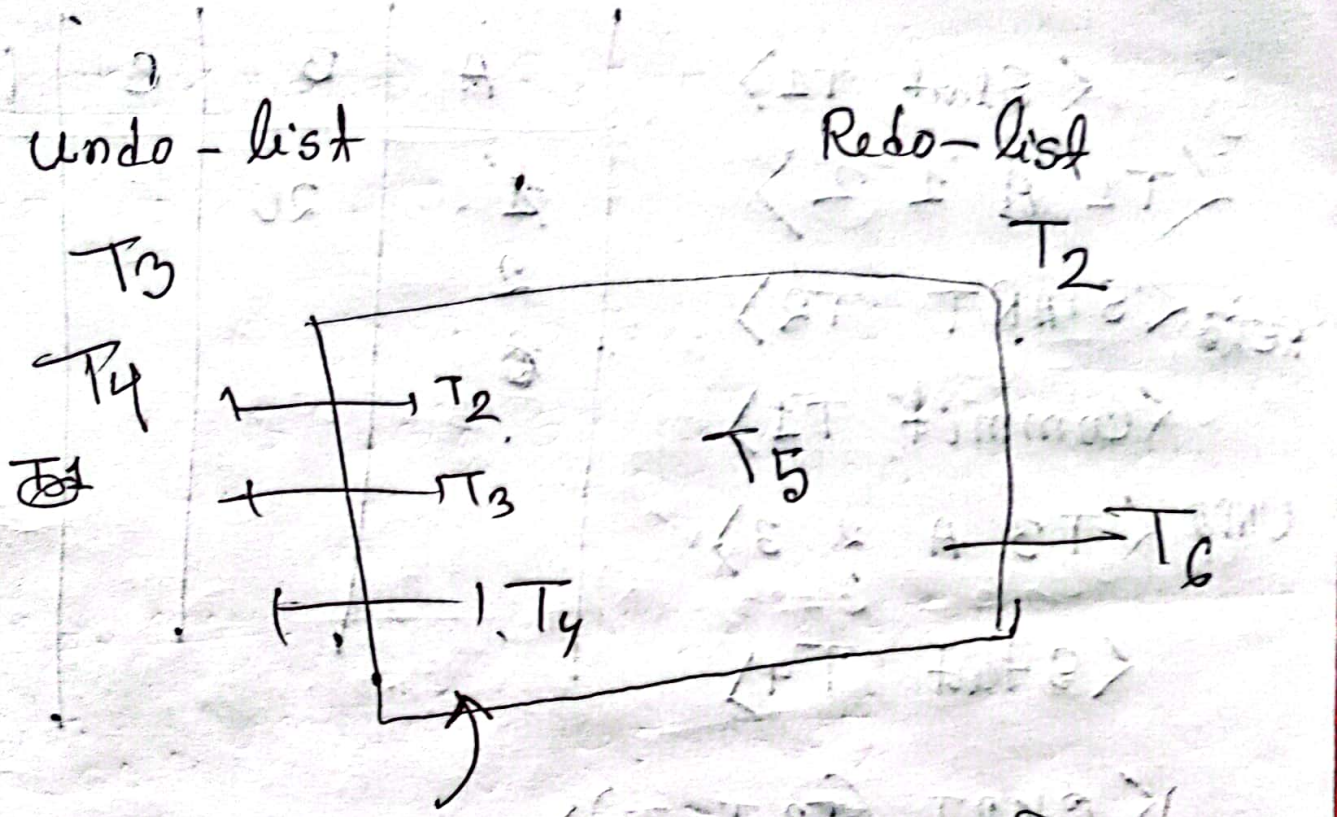
Immediate dataset change: প্রতি transaction এর আগে তথ্য ঠিক করি।

UNDO

- সকল undo কোর্স করে সকল redo করব।

Recovery: Log & checkpoint

↳ কোর্স না হলে undo.



জল্লাদের হাতের থাকলেই নিরাপদ,
অন্যে touch করলেও একটা নিরাপদ না।

→ Traceback

<Start T1>

<T1 A 1 2>

REDO <START T2>

<Commit T1>

UNDO <T3 A 2 3>

<start T4>

<CKPT (T2, T3, T4)>

<T2 B 10 20>

<Commit T2>

<START T5>

<T5 D 1000 2000>

<T4 C 100 200>

<COMMIT T5>

<START T6>

<END CKPT>

A	B	C	D
~~1~~	20		
2			
~~2~~			

<T6 D 2000 3000>

Query Optimization Question Solve :-

$Sc = \frac{n_R}{V(R,A)}$

$ans = \frac{Sc}{f_R}$

a) $\sigma_{B=25}(R)$

$Sc(R,B) = \frac{n_R}{V(R,B)} = \frac{100}{10}$

$= 10$

∴ Number of tuples = 10.

b) $\sigma_{(B=25) \text{ and } (C=80)}(R)$

$Sc(R,B) = \frac{n_R}{V(R,B)}$

$Sc(R,C) = \frac{n_R}{V(R,C)}$

∴ $Sc(R,B)$ and $Sc(R,C) = \frac{n_R}{V(R,B) \times V(R,C)}$

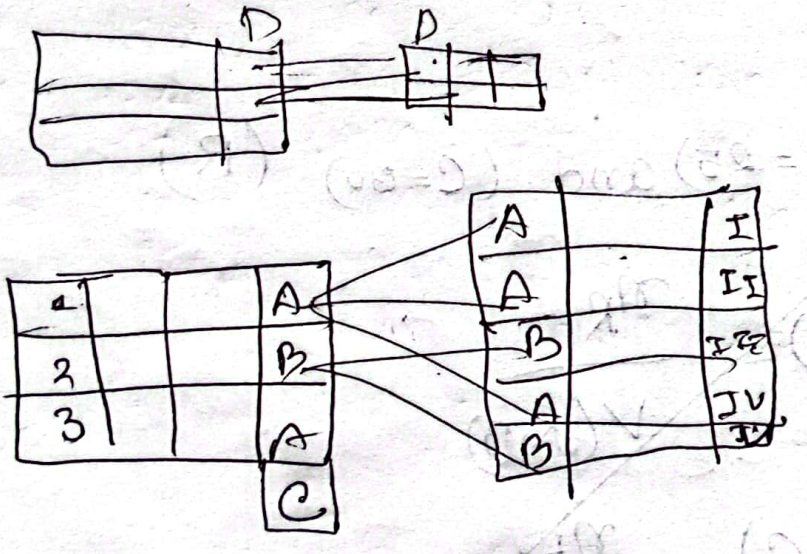
$= \frac{100 \times 10}{10 \times 1} = 1000$

b) $T(\sigma_{B=25 \,\&\, C=30}(R)) = \frac{n_R}{V(R,B)\, V(R,C)} = \frac{100}{10 \times 1}$

$= 10$

c) $T(\sigma_{B>25}) = \frac{n_R}{3} = \frac{100}{3} = 33$

d) $T(\sigma_{B>25 \,\&\, B=15})(R) = \frac{n_R}{V(B>25) \cdot V(B)}$ $= \frac{100}{10 \times 3}$



A	1201	H
M	1202	M
S	1203	M
F	1204	M

H	10%
M	90%

Natural Join

A	1201	H	H	10%
M	1202	M	H	10%
S	1203	M	H	10%
F	1204	M	H	10%
A	1201	H	M	90%
M	1202	M	M	90%
S	1203	M	M	90%
F	1204	M	M	90%

b)

A	1201	H
M	1202	M
S	1203	M
F	1204	M

Covid

H	10%	Y
M	90%	N
M	90%	Tie

Closed Join

A	1201	H	10%	Y
M	1202	M	90%	N
M	1202	M	90%	Tie
S	1203	M	90%	N
S	1203	M	90%	Tie
F	1204	M	90%	N
F	1204	M	90%	Tie

3x2

$\frac{4 \times 3}{2} = 6$

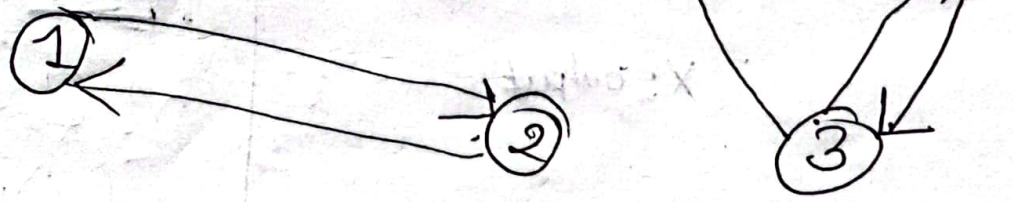
Transaction

4/1/22

Abort = undo

Conflict Serializable : Parallel কে serial বা serial কে parallel করার পর উভয় একই থাকলে,

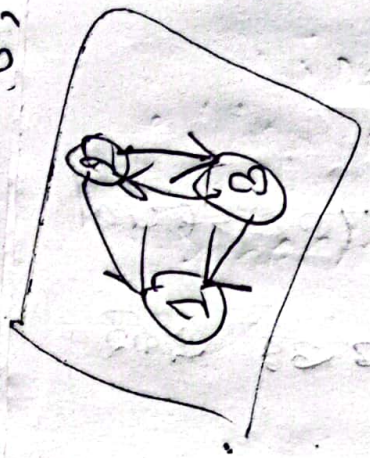
- Precedence graph এ cycle না থাকলে conflict serializable.



File :

p2 (x)

p	1	a
w	2	b
	3	c



1:
2: 1, 3
3: 1, 2

p2(x) p3(Y) w3(x) p1(Y) w2(x) w1(Y) w1(x)

parser(p2(x)

1:

2:

3: for (—) {

□ = findNext(p2(x), j)

X: output

r(2)

r2, r1, r3, r2

w3

r2

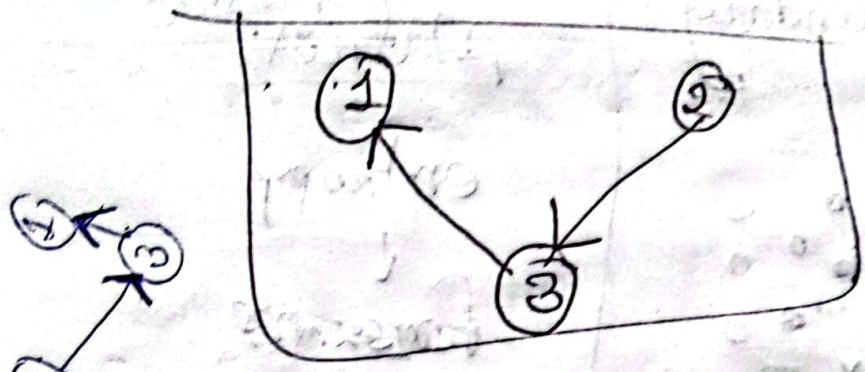
r(1)

op()

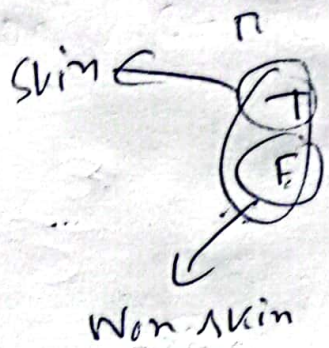
• P2(X) P3(Y) W3(X) R1(Y)

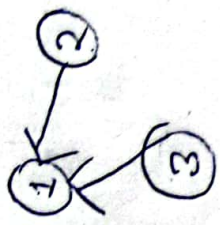
~~R1(X)~~

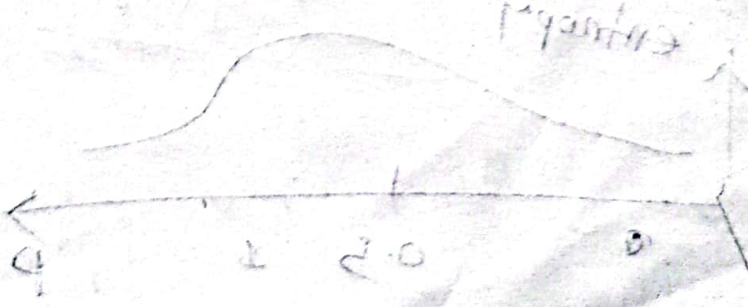
P W2(X) W1(Y) W1(



Apriori
UCI
Conflict serializable
Maze







11/1/23

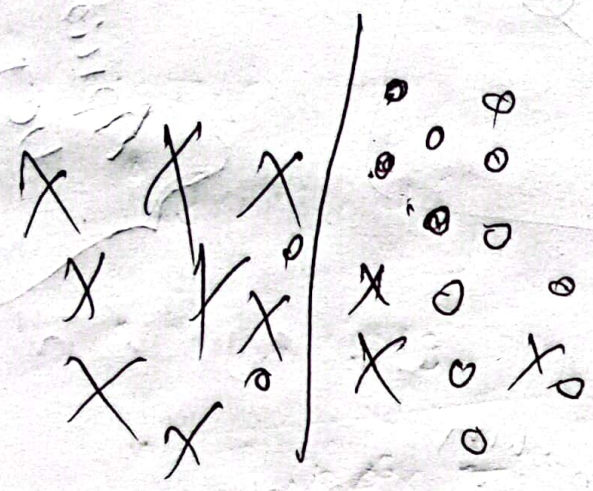
Decision Tree

↳ Deterministic

- Supervised approach.

↳ decision boundary

↳



Metric Matrix

entropy

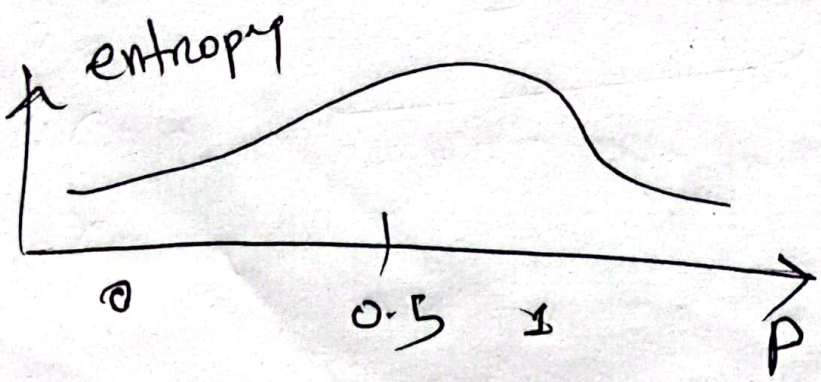
↓

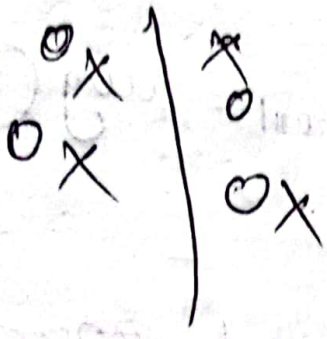
বিশুদ্ধতা পরিমাপ

1

$$\text{Entropy} = \sum_i -P_i \log_2 P_i$$

If $P_i = 0.5$, entropy সর্বোচ্চ মান ধারণ করে





দুই পাশে 50%, 50% তিনিস থাকাই সবচেয়ে chaotic scenario. তাই entropy is highest.

$$E_{Parent} = -\frac{13}{30}\log\left(\frac{13}{30}\right) - \frac{17}{30}\log\left(\frac{17}{30}\right)$$

$$E_{Child} = avg(E_0, E_1)$$

$$E_0 = -\frac{2}{17}\log\left(\frac{2}{17}\right) - \frac{15}{17}\log\left(\frac{15}{17}\right)$$

$$E_1 = -\frac{1}{13}\log(1/13) - \frac{12}{13}\log(1/13)$$

Info gain = $E_{parent} - avg(E_{child})$

UCI dataset থেকে তথ্য entropy বাহির।

90	10

- Sampling
- Pruning
- Index

Parts

Cross Validation
Confusion Matrix
.

Task. 5টি assignment পত্র ব্যবহার

18/1/22

I/O Parallelism

Round Robin :

Partitioning

Point Query	Range Query
• Hashed partition	• Round Robin Partition • Range Partition

- Skew & binning
- Parallelism — query (Inter query)
  query (Intra query)

⇒ exam a algorithm তোমার আসবে। ***

- Fragment-and-replicate join নিয়ে আসবে। ***

---

3 Partition → Query → 4 Parallelism

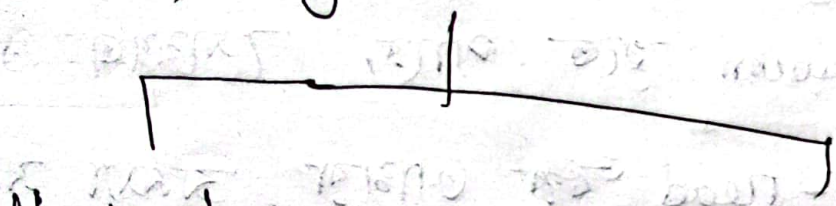
e.g. Join ← e.g. sort (2)

Pipeline Independent

24/1/23

Distributed Database

- Replication ; Segmentation



- Centralized Scheme (name server)
- Naming convention (table name + segment + ip)

TC : Transaction Coordinator, locally

TM : Transaction manager, TM

Two phase commit protocol

Phase 1) আমার এই query করার জন্য কে কে রাজি আছো? (Ready, No, Abort)

phase 2) কাজটা করা, (বা

Phase 2 তে কাজ করার সময় কোন system down হতে পারে, সেক্ষেত্রে তার log file read করে আমরা সমস্যা সমাধান করব, Log can be: (Manager'n log) বা globally viewable log

<Commit T>: Operation শেষ করেছিল কিন্তু আমাকে Reply টুকু পাঠাতে পারে নাই। So we just take its result.

<abort T>: Redo.

<Ready T>: এক্ষেত্রে, System coordinator এর log দেখব, এটা local log. তাই এটা দেখে সিদ্ধান্ত নিতে পারব যে কতখানি কাজ হয়েছে।

এখন, coordinator / Local log fail করলে কি করব?

<Commit T> : কাজ শেষ

<abort T> : Undo

<Ready T> : Undo

<If none of the above, then wait.

Network Bully algorithm : Coordinator মারা গেলে election করে নতুন coordinator নির্বাচিত হবে।

Timeout এর মধ্যে coordinator Reply না দিলে broadcast করব, "Recruiting new coordinator." তারপর system কার কয় আছে count করব। We select the best system (e.g. highest priority) to do our query.

~~Join strategy~~:

Dhaka		Sylhet		Join
5GB	+	5GB	=	10GB

1 কারণ 5GB data transfer করা costly.

2-phase protocol

4টা process

A	B	C	D
Replicate	4	seg	segment

- Logfile hardcoded

Question Pattern: 4 math + 6 theory (14)

Apriori: confidence, support count বের করা

Naive bayes:

Decision tree: এক level, 2 level বের করা; Entropy বের করা।

Serialization: একাধিক serial নিয়ে, কিভাবে করবে,

Def: (what is transaction)