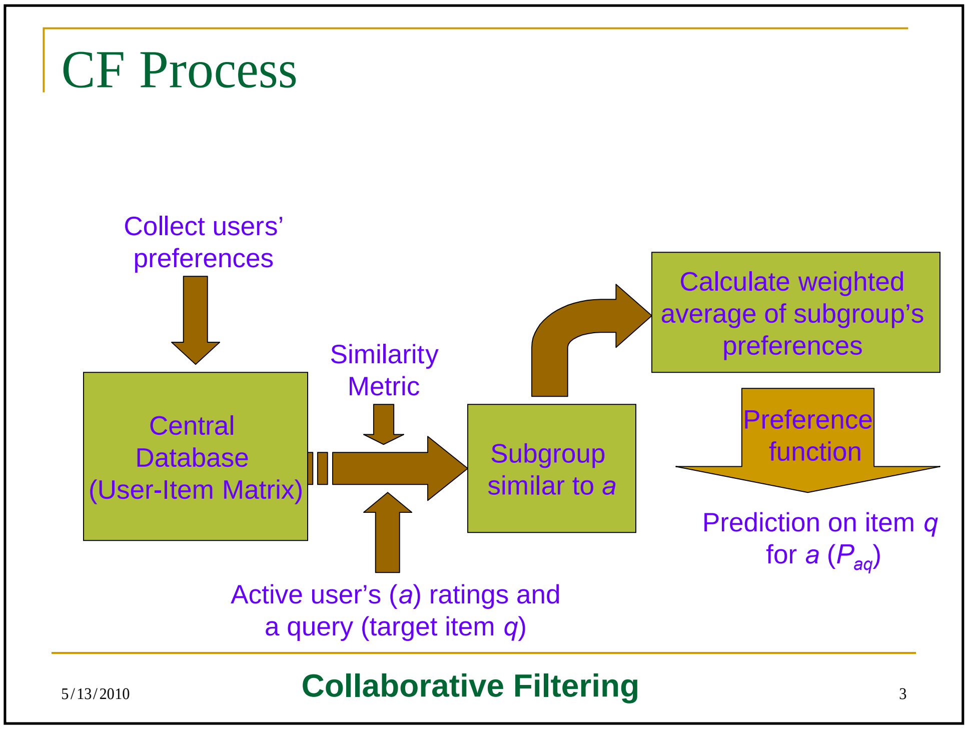
PRIVACY-PRESERVING COLLABORATIVE FILTERING (PPCF)

Collaborative Filtering (CF)

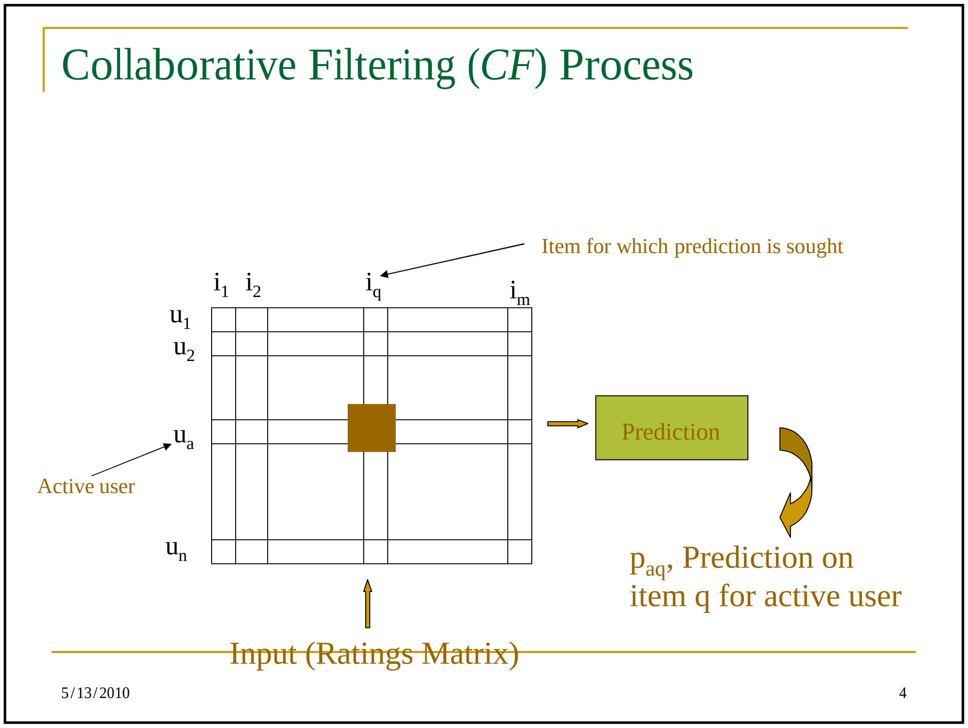
Problem: ***Information Overload***

Solution: ***Collaborative Filtering (CF)***



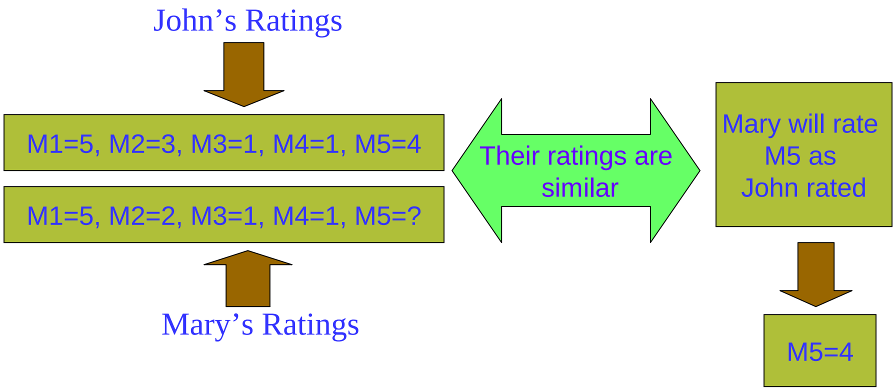
CF Process

Collect users' preferences
Central Database (User-Item Matrix)
Similarity Metric
Active user's (*a*) ratings and a query (target item *q*)
Subgroup similar to *a*
Calculate weighted average of subgroup's preferences
Preference function
Prediction on item *q* for *a* (P_{aq})



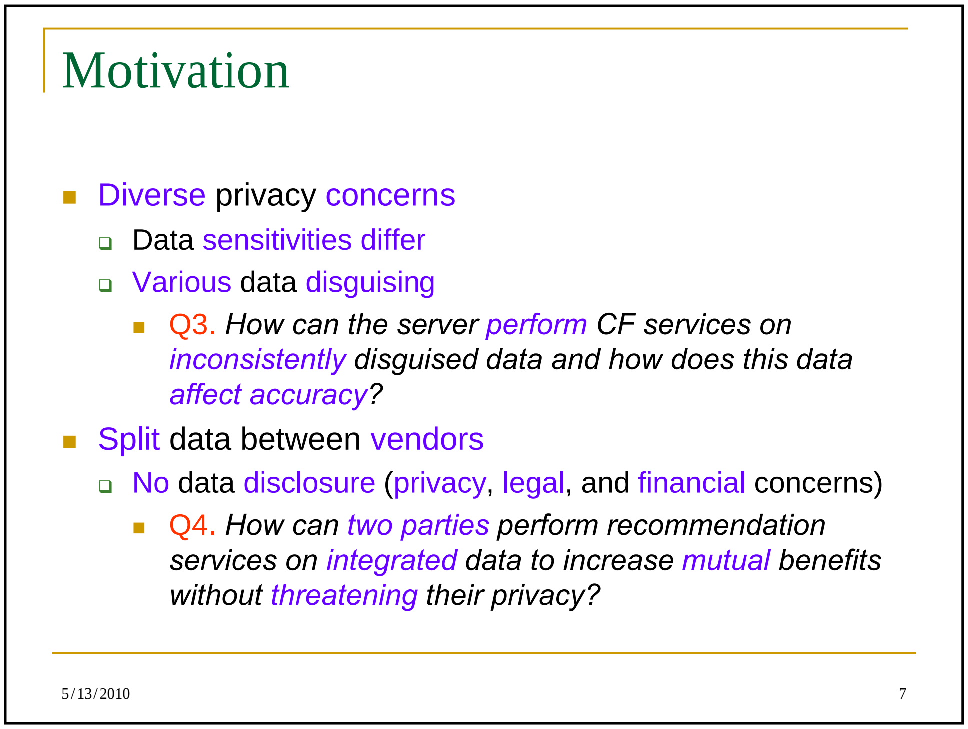
Collaborative Filtering (*CF*) Process

An Example


John’s Ratings
M1=5, M2=3, M3=1, M4=1, M5=4
M1=5, M2=2, M3=1, M4=1, M5=?
Mary’s Ratings
Their ratings are similar
Mary will rate M5 as John rated
M5=4

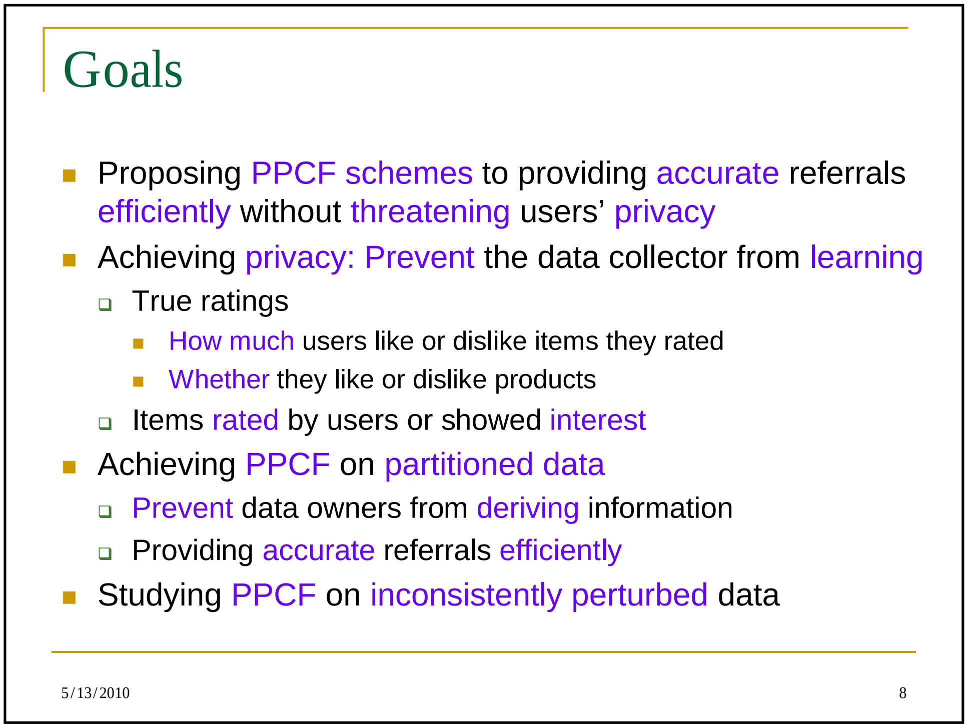
Motivation

- CF has disadvantages
  - Most important: Serious threat to individual privacy
  - Privacy risks: severe & many
  - Vulnerable E-commerce sites
  - Customer data: Valuable asset
  - False data contribution
  - Privacy measures: Key to CF’s success
    - Q1. *How can customers contribute their preferences for CF purposes without greatly compromising their privacy?*
    - Q2. *How can the server provides accurate referrals estimated from perturbed data without exposing users’ privacy?*



Motivation

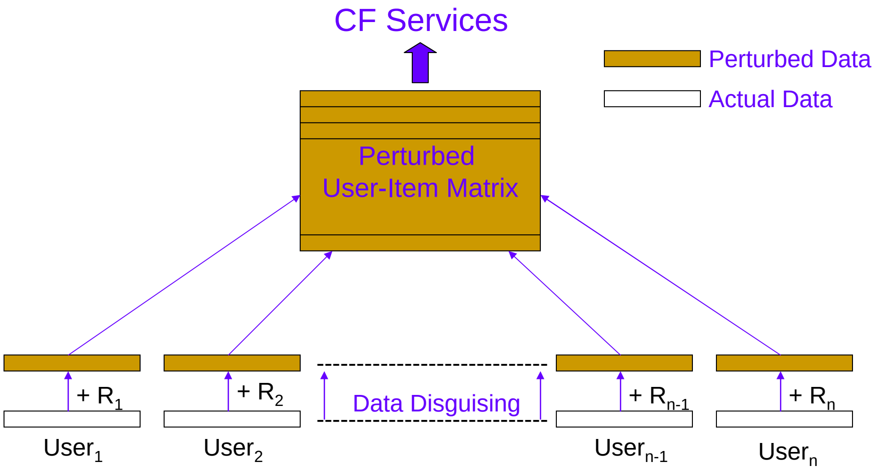
- Diverse privacy concerns
  - Data sensitivities differ
  - Various data disguising
    - Q3. *How can the server perform CF services on inconsistently disguised data and how does this data affect accuracy?*
- Split data between vendors
  - No data disclosure (privacy, legal, and financial concerns)
    - Q4. *How can two parties perform recommendation services on integrated data to increase mutual benefits without threatening their privacy?*



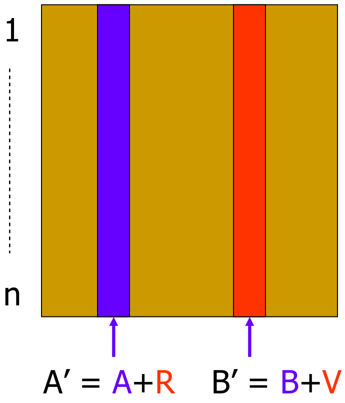
Goals

- Proposing PPCF schemes to providing accurate referrals efficiently without threatening users' privacy
- Achieving privacy: Prevent the data collector from learning
  - True ratings
    - How much users like or dislike items they rated
    - Whether they like or dislike products
  - Items rated by users or showed interest
- Achieving PPCF on partitioned data
  - Prevent data owners from deriving information
  - Providing accurate referrals efficiently
- Studying PPCF on inconsistently perturbed data

RPT



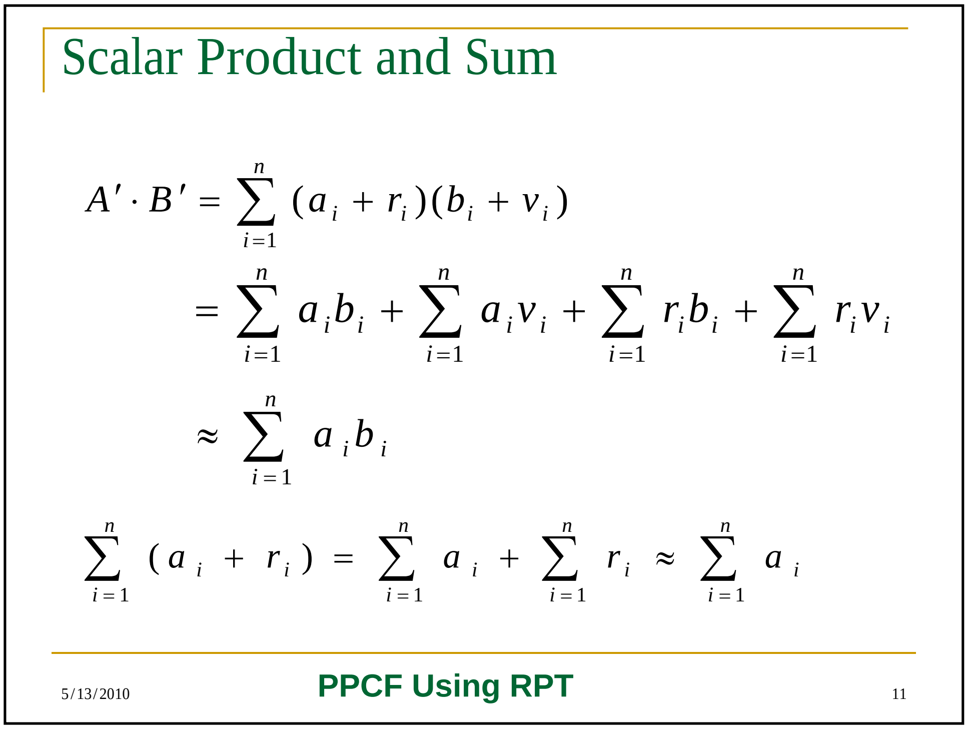
Two Building Blocks



A′ = A+R = (a_1+r_1, …, a_n+r_n)
B′ = B+V = (b_1+v_1, …, b_n+v_n)
R & V: independent
Random values drawn from a distribution with μ = 0
How to compute:

$$A \cdot B = \sum_{i=1}^{n} a_i b_i$$

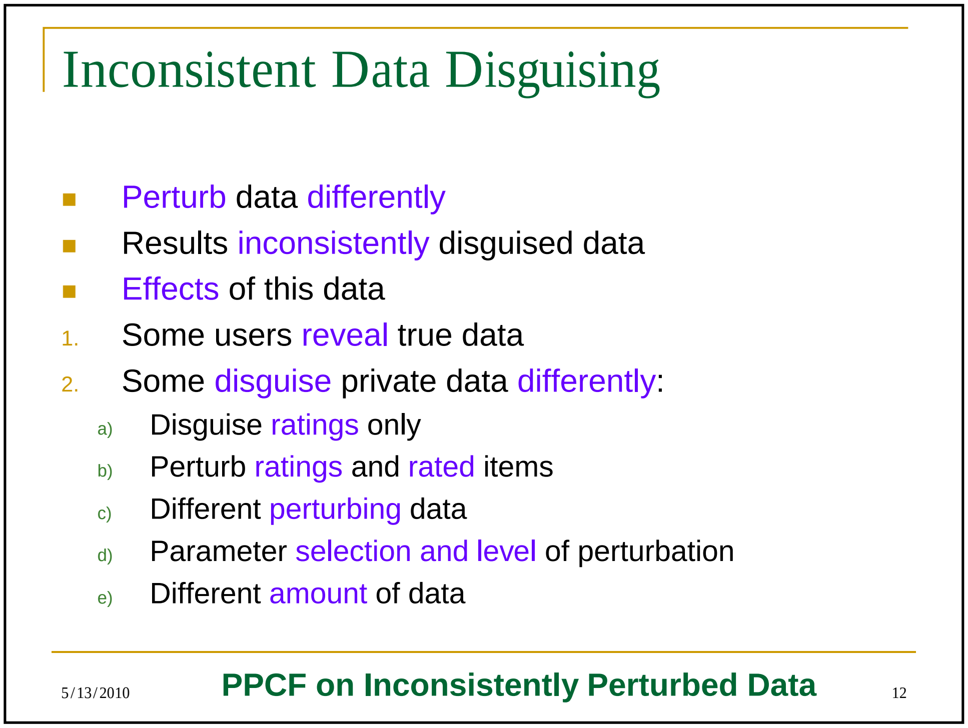
$$SUM = \sum_{i=1}^{n} a_i$$



Scalar Product and Sum

$$
\begin{aligned}
A' \cdot B' &= \sum_{i=1}^{n} (a_i + r_i)(b_i + v_i) \\
&= \sum_{i=1}^{n} a_i b_i + \sum_{i=1}^{n} a_i v_i + \sum_{i=1}^{n} r_i b_i + \sum_{i=1}^{n} r_i v_i \\
&\approx \sum_{i=1}^{n} a_i b_i
\end{aligned}
$$

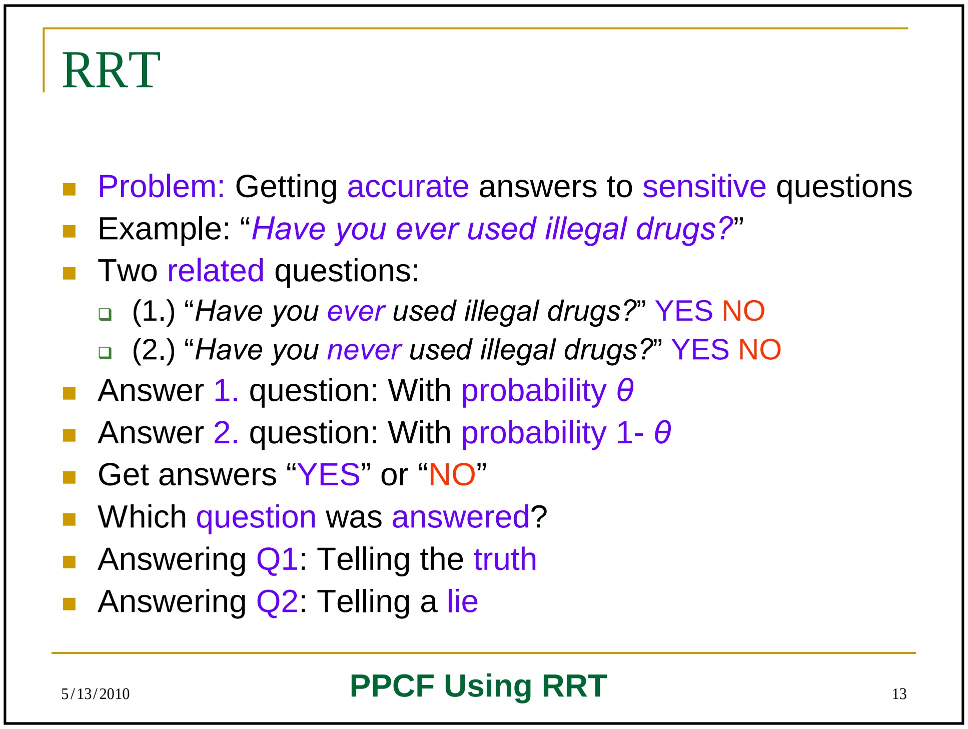
$$\sum_{i=1}^{n} (a_i + r_i) = \sum_{i=1}^{n} a_i + \sum_{i=1}^{n} r_i \approx \sum_{i=1}^{n} a_i$$



Inconsistent Data Disguising

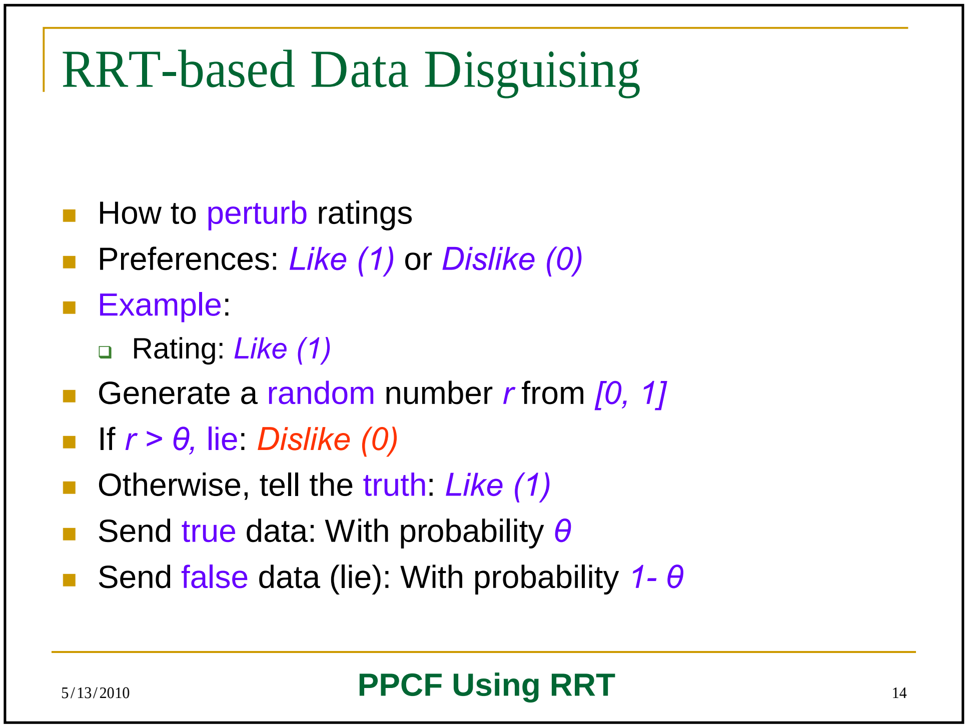
- Perturb data differently
- Results inconsistently disguised data
- Effects of this data

1. Some users reveal true data
2. Some disguise private data differently:
   a) Disguise ratings only
   b) Perturb ratings and rated items
   c) Different perturbing data
   d) Parameter selection and level of perturbation
   e) Different amount of data



RRT

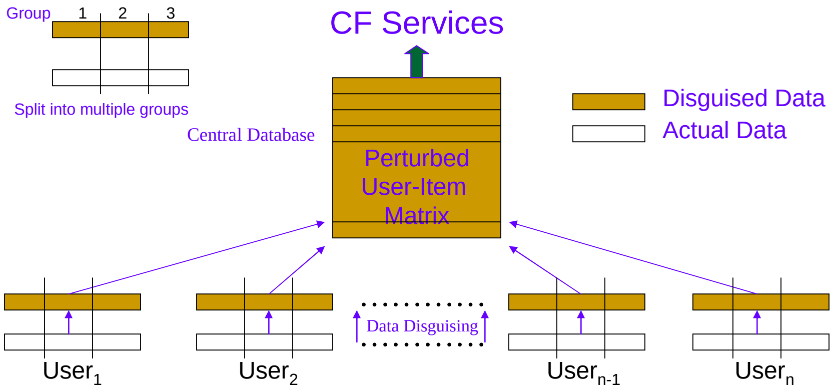
- Problem: Getting accurate answers to sensitive questions
- Example: "*Have you ever used illegal drugs?*"
- Two related questions:
  - (1.) "*Have you ever used illegal drugs?*" YES NO
  - (2.) "*Have you never used illegal drugs?*" YES NO
- Answer 1. question: With probability θ
- Answer 2. question: With probability 1- θ
- Get answers "YES" or "NO"
- Which question was answered?
- Answering Q1: Telling the truth
- Answering Q2: Telling a lie



RRT-based Data Disguising

- How to perturb ratings
- Preferences: *Like (1)* or *Dislike (0)*
- Example:
  - Rating: *Like (1)*
- Generate a random number *r* from *[0, 1]*
- If $r > \theta$, lie: *Dislike (0)*
- Otherwise, tell the truth: *Like (1)*
- Send true data: With probability θ
- Send false data (lie): With probability *1-* θ

PPCF Using RRT



RRT-based Schemes

$U_1 = (1, 1, 0, -, 0, 1, -, 0, 1, 0, -, 1), j = 12$

One-group:

1	1	0	-	0	1	-	0	1	0	-	1

j-group: $\boxed{1}\ \boxed{1}\ \boxed{0}\ \boxed{-}\ \boxed{0}\ \boxed{1}\ \boxed{-}\ \boxed{0}\ \boxed{1}\ \boxed{0}\ \boxed{-}\ \boxed{1}$

Multi-group or *M*-group, $1 < M < j$

Two-group:

1	1	0	-	0	1

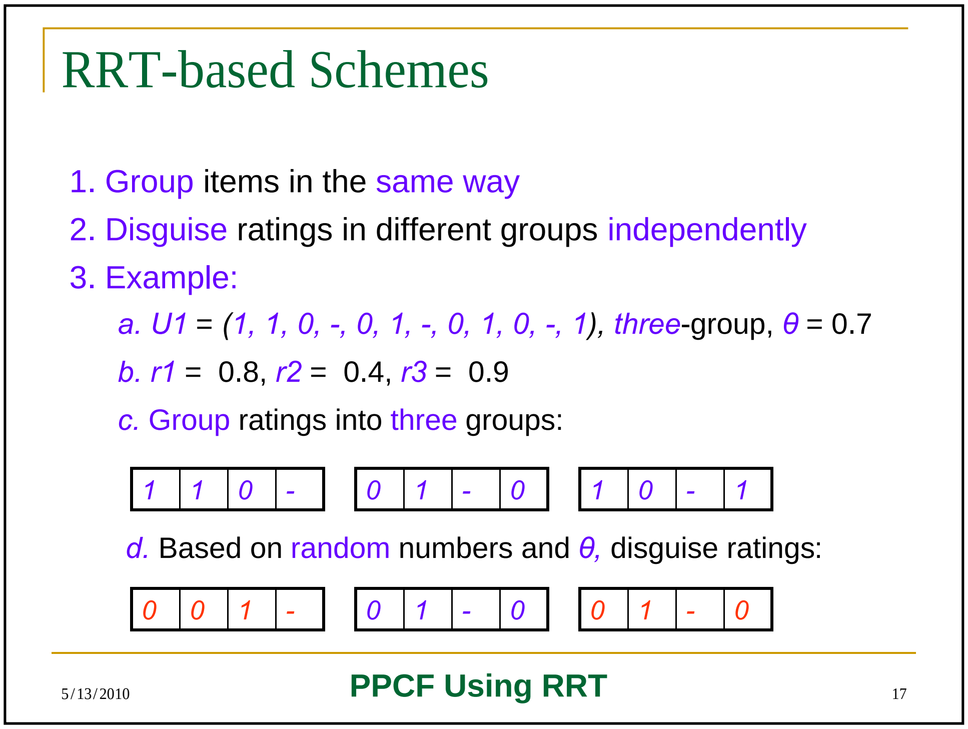
-	0	1	0	-	1

Three-group:

1	1	0	-

0	1	-	0

1	0	-	1



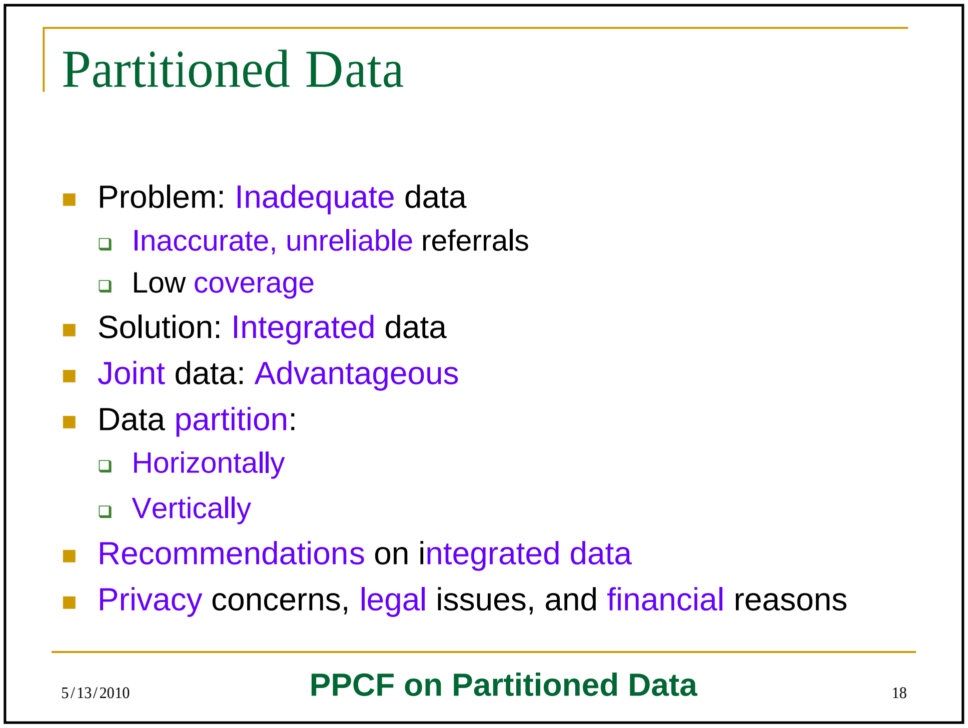
RRT-based Schemes

1. Group items in the same way
2. Disguise ratings in different groups independently
3. Example:
   - *a. U1* = (*1, 1, 0, -, 0, 1, -, 0, 1, 0, -, 1*), *three*-group, *θ* = 0.7
   - *b. r1* = 0.8, *r2* = 0.4, *r3* = 0.9
   - *c.* Group ratings into three groups:

1	*1*	*0*	-	*0*	*1*	-	*0*	*1*	*0*	-	*1*

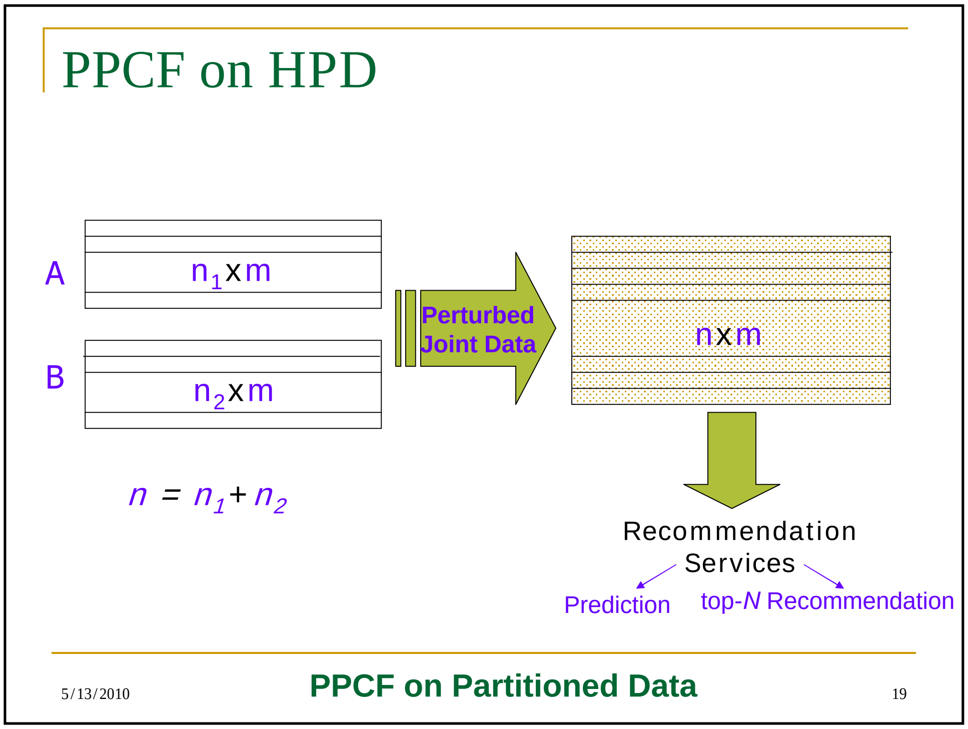
d. Based on random numbers and *θ,* disguise ratings:

0	*0*	*1*	-	*0*	*1*	-	*0*	*0*	*1*	-	*0*



Partitioned Data

- Problem: Inadequate data
  - Inaccurate, unreliable referrals
  - Low coverage
- Solution: Integrated data
- Joint data: Advantageous
- Data partition:
  - Horizontally
  - Vertically
- Recommendations on integrated data
- Privacy concerns, legal issues, and financial reasons



PPCF on HPD

A

$n_1 x m$

B

$n_2 x m$

Perturbed Joint Data

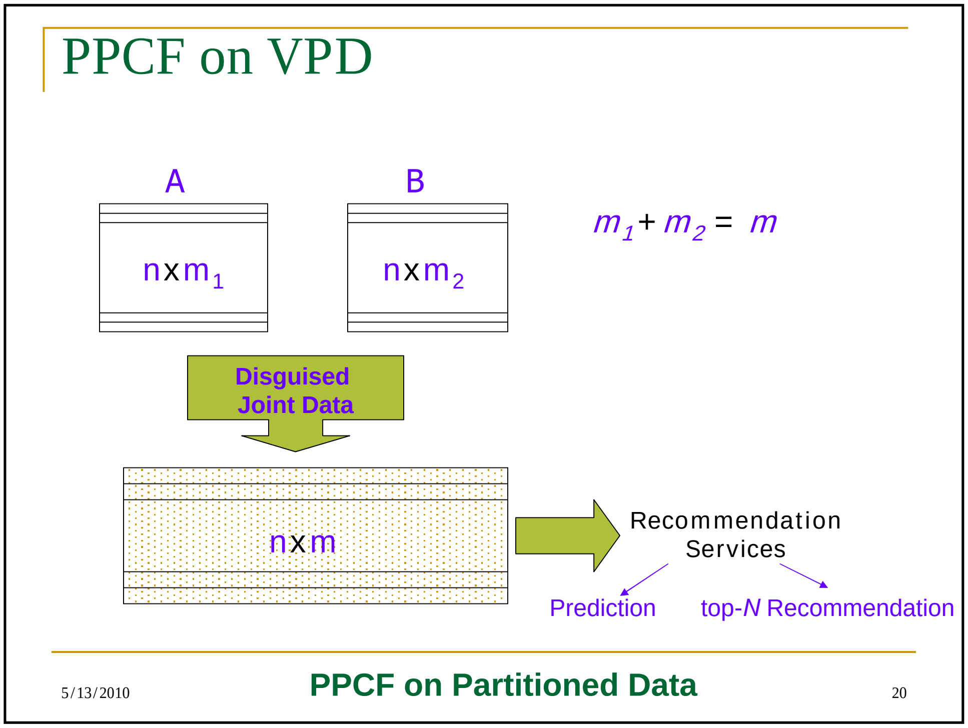
$n x m$

$n = n_1 + n_2$

Recommendation Services

Prediction

top-*N* Recommendation



PPCF on VPD

A

B

$n x m_1$

$n x m_2$

$m_1 + m_2 = m$

Disguised Joint Data

$n x m$

Recommendation Services

Prediction

top-*N* Recommendation