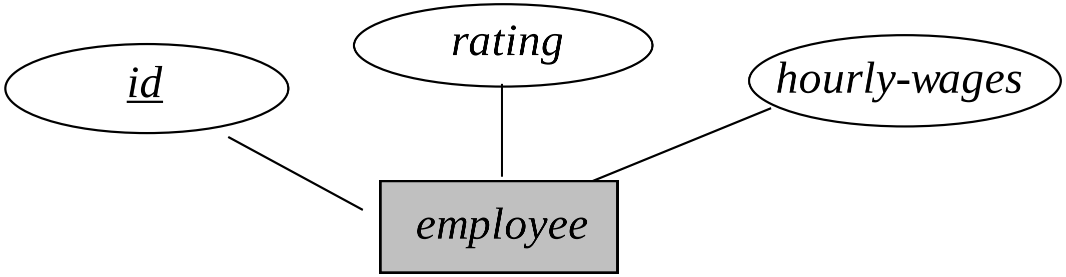
Decomposition and Functional Dependency

Outline

- Redundancy
- Decomposition
- Functional dependency
- Inference rules

Why Redundancy?

- An ER diagram can be directly converted to relational tables, which may contain redundancy, namely, repetition of the same information.



EMPLOYEE

id	*rating*	*hourly-wages*
1	B1	100
2	B1	100
3	B1	100
4	B2	200
5	B2	200
6	B2	200

- If we have the value for the attribute “rating”, we can uniquely determine the attribute “hourly-wages”.

Disadvantages of Redundancy

- Higher space consumption
- Higher update overhead
- Anomalous Insertion/update

EMPLOYEE

id	*rating*	*hourly-wages*
1	B1	100
2	B1	100
3	B1	100
4	B2	200
5	B2	200
6	B2	200

How to tackle this issue?

Solution: Decomposition

EMPLOYEE

id	*rating*	*hourly-wages*
1	B1	100
2	B1	100
3	B1	100
4	B2	200
5	B2	200
6	B2	200

EMPLOYEE

id	*rating*
1	B1
2	B1
3	B1
4	B2
5	B2
6	B2

SALARY

rating	*hourly-wages*
B1	100
B2	200

Basic Questions to Ask Before Using Decomposition

- Do we need to decompose a relation?

- What issues (if any) can a given decomposition cause?
  - May possibly lose information.
  - Need to join several tables together to obtain the complete information.

- How can we make sure that the first issue does not happen?
  - Fulfill lossless-join property (Can provide the same result after we join those decomposed tables.)
  - Fulfill dependency preserving property (discuss later)

Illegal Decomposition

EMPLOYEE

id	*rating*	*hourly-wages*
1	B1	100
2	B1	100
3	B1	100
4	B2	200
5	B2	200
6	B2	200

EMPLOYEE

id
1
2
3
4
5
6

SALARY

rating	*hourly-wages*
B1	100
B2	200

Violate the lossless-join property

Legal Decomposition

EMPLOYEE

id	*rating*
1	B1
2	B1
3	B1
4	B2
5	B2
6	B2

SALARY

rating	*hourly-wages*
B1	100
B2	200

- Checking the “legitimacy” of decomposition:
  - The new tables must have common attribute(s).
  - The common attribute(s) must be the candidate key of at least one new table.

Question 1

1. Consider three tables R, S, and T. Suppose that we decompose the table R into S and T. Is it an illegal decomposition? Explain your answer.

R

A	B	C
1	2	3
2	2	2
3	3	1

S

A	B
1	2
2	2
3	3

T

B	C
2	3
2	2
3	1

Question 2 and Question 3

2. Consider three tables R, S, and T. Suppose that we decompose the table R into S and T. Is it an illegal decomposition? Explain your answer.

R

A	B	C
1	2	4
2	3	5
3	1	8

⇨

S

A	B
1	2
2	3
3	1

T

B	C
2	4
3	5
1	8

3. Consider three tables R, S, and T. Suppose that we decompose the table R into S and T. Is it an illegal decomposition? Explain your answer.

R

A	B	C
1	2	4
2	2	4
3	3	6

⇨

S

A	B
1	2
2	2
3	3

T

B	C
2	4
3	6

Functional Dependency

EMPLOYEE

id	*rating*	*hourly-wages*
1	B1	100
2	B1	100
3	B1	100
4	B2	200
5	B2	200
6	B2	200

- Why does redundancy exist in this table?
  - *rating* determines *hourly-wages*.
  - Once the tuple's *rating* is known, its *hourly-wages* is also decided.

- We have this functional dependency (also called a functional dependence (FD)).

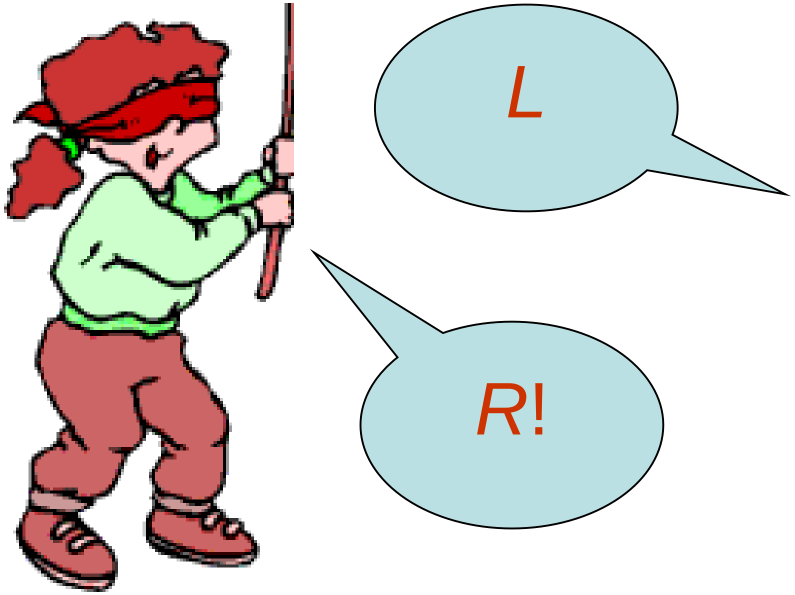
$$rating \rightarrow hourly\text{-}wages.$$

- The attribute value of *rating* can uniquely identify the attribute value of *hourly-wages*.

Functional Dependency

- Let L and R be two sets of attributes. $L \rightarrow R$ means that: if we know a tuple's L, then there is only a single possibility for the tuple's R!

- If we know L, we know R.



Correct Decomposition with Functional Dependency

EMPLOYEE

id	*rating*	*hourly-wages*
1	B1	100
2	B1	100
3	B1	100
4	B2	200
5	B2	200
6	B2	200

EMPLOYEE

id	*rating*
1	B1
2	B1
3	B1
4	B2
5	B2
6	B2

SALARY

rating	*hourly-wages*
B1	100
B2	200

rating → *hourly-wages.*

Where are FDs from?

- Common senses.
  - *HK-id* $\rightarrow$ *name*.
  - *country* $\rightarrow$ *capital*.
  - (*father*, *mother*) $\rightarrow$ *eldest-child*.

- Special constraints of the underlying application.
  - Suppose that every employee has his/her own office.
    - *emp-id* $\rightarrow$ *office-number*.
  - Suppose that every customer has his/her single account.
    - *cust-id* $\rightarrow$ *acc-id*.

- Explored by the inference rules.

Rule 0: A Candidate Key Determines All

- For example, a candidate key of EMPLOYEE is *id*.

- Then, *id* determines any combination of the attributes.
  - $id \rightarrow id$
  - $id \rightarrow rating$
  - $id \rightarrow hourly\text{-}wages$
  - $id \rightarrow rating, hourly\text{-}wages$
  - $id \rightarrow id, rating, hourly\text{-}wages$

EMPLOYEE

id	*rating*	*hourly-wages*
1	B1	100
2	B1	100
3	B1	100
4	B2	200
5	B2	200
6	B2	200

- If we know the id of a tuple, we can know that there is only one possibility for the value(s) of any attribute(s) in this tuple.

Rule 1: Reflexivity

- Let **R** be a relation schema. **X** and **Y** are the subsets of attributes in **R**. If **Y** ⊆ **X**, we have **X** → **Y**.

- Example:
  - (*id*, *rating*) → *id*
  - (*id*, *rating*) → *rating*
  - (*id*, *rating*) → (*id*, *rating*)
  - (*id*, *hourly-wages*) → *id*
  - (*id*, *rating*, *hourly-wages*) → (*id*, *rating*)

EMPLOYEE

id	*rating*	*hourly-wages*
1	B1	100
2	B1	100
3	B1	100
4	B2	200
5	B2	200
6	B2	200

Rule 2: Union (Combining)

- Let **R** be a relation schema. **X**, **Y**, and **Z** are the subsets of attributes in **R**. If **X** → **Y** and **X** → **Z**, we have **X** → **YZ**.

BANK

creditcard-no	*cust-id*	*cust-name*	*cust-city*	*branch-id*	*acc-id*
40101342...	1	Brown	HK	B1	A1
40101343...	2	William	NY	B1	A2
40101344...	1	Brown	HK	B2	A2

- Suppose that we have the following fds.
  - *cust-id* → *cust-name*
  - *cust-id* → *cust-city*

- We have the following fd based on this rule.
  - *cust-id* → (*cust-name*, *cust-city*)

Rule 3: Transitivity

- Let **R** be a relation schema. **X**, **Y**, and **Z** are the subsets of attributes in **R**. If **X** → **Y** and **Y** → **Z**, we have **X** → **Z**.

BANK

creditcard-no	*cust-id*	*cust-name*	*cust-city*	*branch-id*	*acc-id*
40101342...	1	Brown	HK	B1	A1
40101343...	2	William	NY	B1	A2
40101344...	1	Brown	HK	B2	A2

- Suppose that we have the following fds.
  - *creditcard-no* → *cust-id*
  - *cust-id* → *cust-name*

- We have the following fd based on this rule.
  - *creditcard-no* → *cust-name*

Rule 4: Augmentation

- Let **R** be a relation schema. **X**, **Y**, and **Z** are the subsets of attributes in **R**. If **X** → **Y**, we have **XZ** → **YZ**.

BANK

creditcard-no	*cust-id*	*cust-name*	*cust-city*	*branch-id*	*acc-id*
40101342...	1	Brown	HK	B1	A1
40101343...	2	William	NY	B1	A2
40101344...	1	Brown	HK	B2	A2

- Suppose that we have the following fd.
  - *creditcard-no* → *cust-id*

- We have the following fd based on this rule.
  - (*creditcard-no*, *branch-id*) → (*cust-id*, *branch-id*)

Rule 5: Splitting

- Let **R** be a relation schema. **X**, **Y**, and **Z** are the subsets of attributes in **R**. If **X** → **YZ**, we have **X** → **Y** and **X** → **Z**.

BANK

creditcard-no	*cust-id*	*cust-name*	*cust-city*	*branch-id*	*acc-id*
40101342...	1	Brown	HK	B1	A1
40101343...	2	William	NY	B1	A2
40101344...	1	Brown	HK	B2	A2

- Suppose that we have the following fd.
  - *creditcard-no* → (*cust-id*, *cust-name*)

- We have the following fd based on this rule.
  - *creditcard-no* → *cust-id*
  - *creditcard-no* → *cust-name*

Question 4

Consider the following table, which only contains the following three records.

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

Determine whether the following fds hold for this table.

a) $A \rightarrow C$

b) $A \rightarrow D$

c) $A \rightarrow CD$

d) $A \rightarrow DE$

e) $E \rightarrow F$

f) $D \rightarrow EF$

Prove FDs

- To prove an FD is correct, we need to use Rules 0-5 to show the correctness.

- Example: Consider R(A, B, C, D, E) with these FDs, $A \rightarrow B$, $B \rightarrow D$, and $DE \rightarrow C$. Prove of disprove $AE \rightarrow C$.

  We have: $A \rightarrow D$ (Transitivity rule).
  $AE \rightarrow DE$ (Augmentation rule).
  $AE \rightarrow C$ ($DE \rightarrow C$)

Disprove FDs

- To disprove an FD, we need to find some counterexample to verify it is incorrect.

- Example: Consider R(A, B, C, D, E) with these FDs, $A \rightarrow B$, $B \rightarrow D$, and $DE \rightarrow C$. Prove of disprove $A \rightarrow C$.

A	B	C	D	E
2	4	3	1	3
2	4	6	1	4

Question 5 and Question 6

5. Consider the table R(A, B, C, D). Suppose that we have the following FDs.

$$A \rightarrow C \text{ and } D \rightarrow B$$

   Prove or disprove $AD \rightarrow CB$.

6. Consider the table R(A, B, C, D, E, F, G). Suppose that we have the following FDs.

$$DEF \rightarrow G,\ A \rightarrow C, \text{ and } C \rightarrow DE$$

   Prove or disprove $AF \rightarrow G$.

Question 7 and Question 8

7. Consider the table R(A, B, C, D, E, F, G). Suppose that we have the following FDs.

$$DE \rightarrow FG,\ A \rightarrow D,\ A \rightarrow E, \text{ and } A \rightarrow C$$

   Prove or disprove $A \rightarrow CFG$.

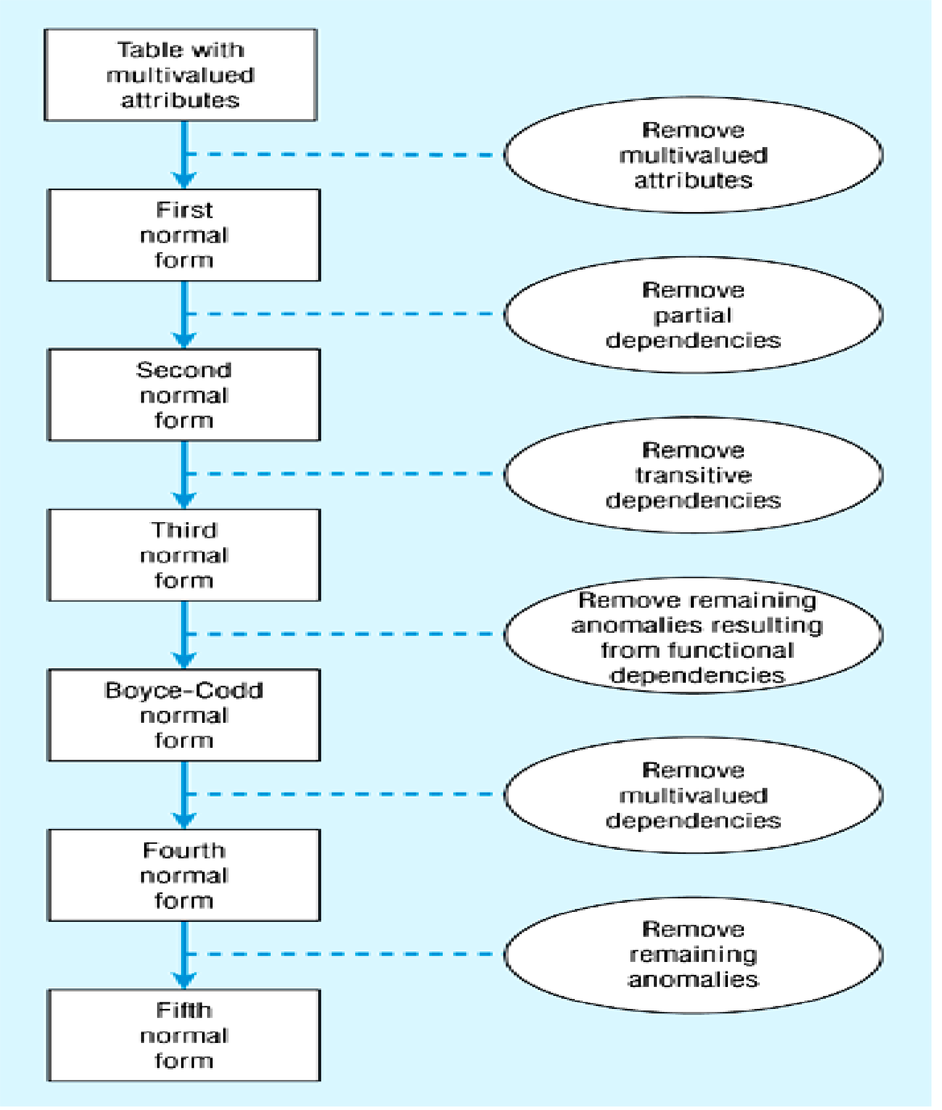
8. Consider the table R(A, B, C, D, E, F, G). Suppose that we have the following FDs.

$$E \rightarrow FG,\ A \rightarrow E, \text{ and } C \rightarrow A$$

   Prove or disprove $A \rightarrow CFG$.

Other Ways of Eliminating Redundancy

- Normalization
  - Extended reading: 1NF, 2NF, 3NF, …



Solution to Question 1

- Yes. It violates the lossless-join property.

$S \bowtie_{S.B=T.B} T =$

A	B	C
1	2	3
1	2	2
2	2	3
2	2	2
3	3	1

$\neq R$

Solutions to Questions 2 & 3

2. No. It satisfies the lossless-join property.

$S \bowtie_{S.B=T.B} T =$

A	B	C
1	2	4
2	3	5
3	1	8

$= R$

3. No. It satisfies the lossless-join property.

$S \bowtie_{S.B=T.B} T =$

A	B	C
1	2	4
2	2	4
3	3	6

$= R$

Solution to Question 4

a. Yes
b. Yes
c. Yes
d. Yes
e. No
f. Yes

Solution to Question 5

Answer:

1) $A \rightarrow C$
2) $AD \rightarrow CD$ (Augmentation 1))
3) $AD \rightarrow D$ (Reflexivity)
4) $D \rightarrow B$
5) $AD \rightarrow B$ (Transitivity 3) and 4))
6) $AD \rightarrow BCD$ (Union 2) and 5))
7) $BCD \rightarrow CB$ (Reflexivity)
8) $AD \rightarrow CB$ (Transitivity 6) and 7))

Solution to Question 6

Answer:

1) $A \rightarrow C$
2) $C \rightarrow DE$
3) $A \rightarrow DE$ (Transitivity 1) and 2))
4) $AF \rightarrow DEF$ (Augmentation 3))
5) $DEF \rightarrow G$
6) $AF \rightarrow G$ (Transitivity 4) and 5))

Solution to Question 7

Answer:

1) $A \rightarrow D$
2) $A \rightarrow E$
3) $A \rightarrow DE$ (Union 1) and 2))
4) $DE \rightarrow FG$
5) $A \rightarrow FG$ (Transitivity 3) and 4))
6) $A \rightarrow C$
7) $A \rightarrow CFG$

Solution to Question 8

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

$E \rightarrow FG$ holds $A \rightarrow E$ holds $C \rightarrow A$ holds

$A \rightarrow CFG$ does not hold

There are two cases for C, F, and G when A=1

- $A = 1 \rightarrow C = 2, F = 1, G = 2$
- $A = 1 \rightarrow C = 3, F = 1, G = 2$