Derivations of Functional Dependencies

CSIT882: Data Management Systems

UNIVERSITY
OF WOLLONGONG
AUSTRALIA

Outline

- Derivations of functional dependencies
- Functional dependencies and keys
- Examples

Recap: Functional dependency

Let $R = (A_1, ..., A_n)$ be a relational schema (a header of relational table) and let X, Y be nonempty subsets of R

We say that functional dependency $X \rightarrow Y$ is valid in a relational schema R if ...

... for any relational table r with relational schema R, it is not possible that r has two rows that agree in the components for all attributes in set X yet disagree in one or more component for attributes in set Y

FD: relation between two sets

A functional dependency is a relation between two **sets** of attributes.

i.e., the value for a set of attributes determines the value for another set of attributes.

Examples:

$XY \rightarrow WZ$

$XW \rightarrow Z$

$Z \rightarrow XQ$

Derivations of functional dependencies - motivating example

Employee				
e#	ename	department	department-address	chairperson

Assume there are FDs in R, what can we infer?

e# → ename

e# → department

department → department-address

department → chairperson

chairperson → department

e# → department-address

e# → chairperson

chairperson → department-address

e# → ename, department

Inferring other FDs

Given $A \rightarrow B$ and $B \rightarrow C$ on a relation R,
what do we know about $A \rightarrow C$?

It's true that given a set F of functional dependencies, there are other functional dependencies that are **logically implied** by F.

- $F \models X \rightarrow Y$

Denotes that set of FDs F infers $X \rightarrow Y$ if all relation instances satisfying F also satisfies $X \rightarrow Y$.

- Example:

  $F = A \rightarrow B, B \rightarrow C$

  $F \models A \rightarrow C$

Usually, the schema designer will only specify the functional dependencies that are semantically obvious.

Derivations of functional dependencies

- It is always true that $A \rightarrow A$ (no matter what A means)
- It is always true that $A, B \rightarrow A$ (no matter what A, B mean)

- If it is true that $A \rightarrow B$ then it is true that $A, C \rightarrow B$
- If it is true that $A \rightarrow B, C$ then it is true that $A \rightarrow B$ and $A \rightarrow C$
- If it is true that $A \rightarrow B$ and $B \rightarrow C$ then it is true that $A \rightarrow C$

Are there any rules we can use to infer FDs?

Armstrong's Axioms

Let $R = (A_1, \ldots, A_n)$ be a relational schema, and let X, Y and Z be nonempty subsets of R

- Rule 1 (reflexivity)

  if $Y \subseteq X$, then $X \rightarrow Y$

- Rule 2 (augmentation)

  if $X \rightarrow Y$, then $XZ \rightarrow YZ$

- Rule 3 (transitivity)

  if $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

Above rules form a minimal and complete set of axioms

Practice

Given: $R = (A, B, C, G, H, I)$

$F = \{ A \rightarrow B, A \rightarrow C, CG \rightarrow H, CG \rightarrow I, B \rightarrow H \}$

These FDs can be inferred/derived:

$A \rightarrow H$

$AG \rightarrow I$

$CG \rightarrow HI$

Armstrong's Axiom

- Rule 1 (reflexivity)

  if $Y \subseteq X$, then $X \rightarrow Y$
- Rule 2 (augmentation)

  if $X \rightarrow Y$, then $XZ \rightarrow YZ$
- Rule 3 (transitivity)

  if $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

Solution

Given: $R = (A, B, C, G, H, I)$

$F = \{ A \rightarrow B, A \rightarrow C, CG \rightarrow H, CG \rightarrow I, B \rightarrow H \}$

$A \rightarrow H$

by transitivity from $A \rightarrow B$ and $B \rightarrow H$

$AG \rightarrow I$

by augmenting $A \rightarrow C$ to get $AG \rightarrow CG$

then transitivity with given $CG \rightarrow I$

$CG \rightarrow HI$

by augmenting $CG \rightarrow I$ to infer $CG \rightarrow CGI$,

then augmenting $CG \rightarrow H$ to infer $CGI \rightarrow HI$,

by a transitivity to get $CG \rightarrow HI$

Additional inference rules

Additional Rules we inferred from Armstrong's axioms.

- Rule 4 (union)

  if $X \rightarrow Y$ and $X \rightarrow Z$, then $X \rightarrow YZ$

- Rule 5 (pseudotransitivity)

  if $X \rightarrow Y$ and $WY \rightarrow Z$ then $WX \rightarrow Z$

- Rule 6 (decomposition)

  if $X \rightarrow Y$ and $Z \subseteq Y$, then $X \rightarrow Z$

Proving additional rules

Let's try prove Rule 5 (pseudotransitivity)

if $X \rightarrow Y$ and $WY \rightarrow Z$ then $WX \rightarrow Z$

Armstrong's Axiom

- Rule 1 (reflexivity)

  if $Y \subseteq X$, then $X \rightarrow Y$
- Rule 2 (augmentation)

  if $X \rightarrow Y$, then $XZ \rightarrow YZ$
- Rule 3 (transitivity)

  if $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

Solution

Let's try prove Rule 5 (pseudotransitivity)

if $X \rightarrow Y$ and $WY \rightarrow Z$ then $WX \rightarrow Z$

Step 1: $XW \rightarrow WY$ (augmentation)

Step 2: $WY \rightarrow Z$ (given)

Step 3: $XW \rightarrow Z$ (transitivity of S1&2)

Armstrong's Axiom

- Rule 1 (reflexivity)

  if $Y \subseteq X$, then $X \rightarrow Y$
- Rule 2 (augmentation)

  if $X \rightarrow Y$, then $XZ \rightarrow YZ$
- Rule 3 (transitivity)

  if $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

Proving additional rules

Let's try prove Rule 6 (decomposition)

if $X \rightarrow Y$ and $Z \subseteq Y$, then $X \rightarrow Z$

Armstrong's Axiom

- Rule 1 (reflexivity)

  if $Y \subseteq X$, then $X \rightarrow Y$

- Rule 2 (augmentation)

  if $X \rightarrow Y$, then $XZ \rightarrow YZ$

- Rule 3 (transitivity)

  if $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

Solution

Let's try prove Rule 6 (decomposition)

if $X \rightarrow Y$ and $Z \subseteq Y$, then $X \rightarrow Z$

Step 1: $Y \rightarrow Z$ (reflexivity)

Step 2: $X \rightarrow Y$ (given)

Step 3: $X \rightarrow Z$ (transitivity of S1&2)

Armstrong's Axiom

- Rule 1 (reflexivity)

  if $Y \subseteq X$, then $X \rightarrow Y$

- Rule 2 (augmentation)

  if $X \rightarrow Y$, then $XZ \rightarrow YZ$

- Rule 3 (transitivity)

  if $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

Outline

- Derivations of functional dependencies
- Functional dependencies and keys
- Examples

Recap: Keys

Superkey

Superkey is a nonempty subset X of relational schema $R = (A_1, ..., A_n)$ such that for any two rows t1, t2 in a relational table defined over R, $t1[X] \neq t2[X]$.

If X is a superkey in R then $X \rightarrow (A_1, ..., A_n)$

Minimal key

Minimal key is a superkey K with an additional property such that removal of any attribute from K will cause K not to be a superkey

Recap: Keys

Examples:

- A set of attributes **`{snum}`** is a minimal key in a relational schema **`STUDENT={snum, first-name, last-name, date-of-birth}`**
- A set of attributes **`{snum, last-name}`** is a superkey in a relational schema **`STUDENT={snum, first-name, last-name, date-of-birth}`**
- A set of attributes **`{snum, code, enrolment-date, enrolment-time}`** is a minimal key in a relational schema **`ENROLMENT={snum, code, enrolment-date, enrolment-time}`**
- A set of attributes **`{bldg#, room#}`** is a minimal key in a relational schema **`ROOM={bldg#, room#, area}`**
- A set of attributes **`{p#, manufacturer, price}`** is a superkey in a relational schema **`PART={p#, name, price, manufacturer}`**
- A set of attributes **`{p#, manufacturer}`** is a superkey in a relational schema **`PART={p#, name, price, manufacturer}`**
- A set of attributes **`{p#}`** is a minimal key in a relational schema **`PART={p# name, price, manufacturer}`**

Recap: Keys

Primary key

Primary key is an arbitrarily selected minimal key.

Candidate key

Candidate key is any other minimal key which is not primary

Examples:

- A set of attributes `{snum}` and a set of attributes `{first-name, last-name, date-of-birth}` are the candidate keys in a relational schema `STUDENT={snum, first-name, last-name, date-of-birth}`
- A candidate key `{snum}` can be selected by a database designer as a primary key

Functional dependencies and keys

Let $R = (A_1, \ldots, A_n)$ be a relational schema, and let X, Y be nonempty subsets of R such that $X \cup Y = R$

- If $X \rightarrow Y$ is valid in R, then X is a key
- If X is a key, then $X \rightarrow Y$ is valid in R

Outline

- Derivations of functional dependencies
- Functional dependencies and keys
- Examples

Derivations of functional dependencies

Given $R = (A, B, C)$ and $F = \{A \rightarrow B, A \rightarrow C\}$

Is it true that $A \rightarrow BC$?

Solution

Given $R = (A, B, C)$ and $F = \{A \rightarrow B, A \rightarrow C\}$

Is it true that $A \rightarrow BC$?

$A \rightarrow C$ (given)

$AB \rightarrow BC$ (augmentation) (S1)

$A \rightarrow B$ (given)

$A \rightarrow AB$ (augmentation) (S2)

=> $A \rightarrow BC$ (transitivity of S1 and S2)

$A \cup BC = R$

=> A is a minimal key

Derivations of functional dependencies

Given $R = (A, B, C)$ and $F = \{A \rightarrow B, B \rightarrow C\}$

Is it true that $A \rightarrow C$?

Solution

Given $R = (A, B, C)$ and $F = \{A \rightarrow B, B \rightarrow C\}$

Is it true that $A \rightarrow C$?

$A \rightarrow B, B \rightarrow C$ (given)
$A \rightarrow C$ (transitivity)

$A \rightarrow B, A \rightarrow C \Rightarrow A \rightarrow BC$ (union)

$A \cup BC = R$
=> A is a minimal key

Derivations of functional dependencies

Given $R = (A, B, C)$ and $F = \{A \rightarrow BC\}$

Is it true that $A \rightarrow B$ and $A \rightarrow C$?

Solution

Given $R = (A, B, C)$ and $F = \{A \rightarrow BC\}$

Is it true that $A \rightarrow B$ and $A \rightarrow C$?

$BC \rightarrow C$ (reflexivity) (S1)
$A \rightarrow BC$ (given) (S2)
$A \rightarrow C$ (transitivity of S1 and S2)

$BC \rightarrow B$ (reflexivity) (S3)
$A \rightarrow BC$ (given) (S4)
$A \rightarrow B$ (transitivity of S3 and S4)

Derivations of functional dependencies

Given $R = (A, B, C, D)$ and
$F = \{A \rightarrow B, A \rightarrow C, BC \rightarrow D\}$

Is it true that $A \rightarrow D$?

Solution

Given $R = (A, B, C, D)$ and

$F = \{A \rightarrow B, A \rightarrow C, BC \rightarrow D\}$

Is it true that $A \rightarrow D$?

$A \rightarrow BC$ (union)

$BC \rightarrow D$ (given)

=> $A \rightarrow D$ (transitivity)

$A \rightarrow BCD$ (union)

$A \cup BCD = R$

=> A is a minimal key

References

- **Elmasri R., Navathe S. B., *Fundamentals of Database Systems*, chapter 10.2**
- **R. Ramakrishnan, J. Gehrke *Database Management Systems*, chapters 19.2, 19.3**