Database Management Systems

Madhavan Mukund

https://www.cmi.ac.in/~madhavan

Sai University

Lecture 12, 27 September 2023

Functional dependencies

- $A_1, A_2, \ldots, A_k \rightarrow B_1, B_2, \ldots B_m$
  - LHS atributes uniquely fix RHS attributes
  - Must hold for every instance — semantic property of attributes
- Need not correspond to superkeys
  - `dept_name` $\rightarrow$ `building`
  - `dept_name` $\rightarrow$ `budget`
- Use to identify sources of redundancy, guide decomposition

ID	*name*	*salary*	*dept_name*	*building*	*budget*
22222	Einstein	95000	Physics	Watson	70000
12121	Wu	90000	Finance	Painter	120000
32343	El Said	60000	History	Painter	50000
45565	Katz	75000	Comp. Sci.	Taylor	100000
98345	Kim	80000	Elec. Eng.	Taylor	85000
76766	Crick	72000	Biology	Watson	90000
10101	Srinivasan	65000	Comp. Sci.	Taylor	100000
58583	Califieri	62000	History	Painter	50000
83821	Brandt	92000	Comp. Sci.	Taylor	100000
15151	Mozart	40000	Music	Packard	80000
33456	Gold	87000	Physics	Watson	70000
76543	Singh	80000	Finance	Painter	120000

Boyce-Codd Normal Form (BCNF)

- Relational schema R, set of functional dependencies F
- R is in BCNF if, for every $\alpha \rightarrow \beta \in F^+$, one of the following holds
  - $\alpha \rightarrow \beta$ is trivial (i.e., $\beta \subseteq \alpha$)
  - α is a superkey for R
- $\alpha \rightarrow \beta \in F^+$ is a BCNF violation for R if neither of the following holds
  - $\alpha \rightarrow \beta$ is trivial (i.e., $\beta \subseteq \alpha$)
  - α is a superkey for R
- To fix this, decompose R as
  - $\alpha \cup \beta$
  - $R \setminus (\beta \setminus \alpha)$

$\nearrow$ α is the overlap

Dependency preservation

- Advisor(student_id,faculty_id,dept_name)
- Each faculty member is in only one department
- Students can be across multiple departments
- Each student has at most one advisor in each department
- BCNF decomposition is (student_id,faculty_id), (faculty_id,dept_name)
- Functional dependencies
  - faculty_id → dept_name
  - student_id,dept_name → faculty_id
- Need join to check second dependency

Third normal form (3NF)

- R is in 3NF if, for every $\alpha \to \beta \in F^+$, one of the following holds
  - $\alpha \to \beta$ is trivial (i.e., $\beta \subseteq \alpha$)
  - α is a superkey for R
  - Each attribute A in $\beta \setminus \alpha$ is contained in some candidate key for R
- BCNF is a stricter condition than 3NF
- Priorities
  - Lossless decomposition
  - BCNF
  - Dependency preservation

stud, fac, dept

key

stud, dept → fac

$\boxed{\text{fac → dept}}$

Why 3 ?

1NF = data is "simple"

2NF ≈ BCNF

Computing the closure of a set of attributes

- Iterative algorithm — $\alpha = \{A_1, A_2, \ldots, A_k\}$, check if B is in closure α^+

  Initialize α^+ to $\{A_1, A_2, \ldots, A_k\}$
  repeat
      for each $\beta \rightarrow \gamma$ in F
          if $\beta \subseteq \alpha^+$, add γ to A^+
      end
  until no change in α^+

Semantic condition

$\alpha \rightarrow B$

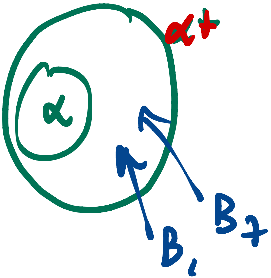
Syntactic condition

$B \in \alpha^+$

- Need to establish correctness
  - Soundness — if B is included in α^+, then $\underbrace{A_1, A_2, \ldots, A_K}_{\alpha} \rightarrow B$ is indeed a functional dependency
  - Completeness — if $A_1, A_2, \ldots, A_K \rightarrow B$ is a functional dependency, then B will be added to α^+

$\alpha \to \beta$

$\underbrace{A_1 A_2 \ldots A_k}_{\alpha} \to B_1 \ldots B_m$



Each $B_i \in \alpha^+$

$A \to B, C$ vs $A \to B$
$A \to C$

$a \in A$
only one row
with $A = a$

$\alpha \to B_i$ hold
for
any $B_i \in \alpha^+$

$\alpha \to A_i$

Canonical basis

- Given functional dependecies F, closure F^+ set of all implied dependencies
  - For each subset of attributes α, compute attribute closure α^+
  - Can split rules and combine right hand sides
    $\alpha \rightarrow B, C$ iff $\alpha \rightarrow B$, $\alpha \rightarrow C$

Canonical basis

- Given functional dependecies F, closure F^+ set of all implied dependencies
  - For each subset of attributes α, compute attribute closure α^+
  - Can split rules and combine right hand sides
    $\alpha \rightarrow B, C$ iff $\alpha \rightarrow B$, $\alpha \rightarrow C$
- Could have an alternative set of dependencies G such that $G^+ = F^+$

Canonical basis

- Given functional dependecies F, closure F^+ set of all implied dependencies
  - For each subset of attributes α, compute attribute closure α^+
  - Can split rules and combine right hand sides
    $\alpha \rightarrow B, C$ iff $\alpha \rightarrow B$, $\alpha \rightarrow C$
- Could have an alternative set of dependencies G such that $G^+ = F^+$
- Canonical basis for $F+$ is G such that
  - $G^+ = F^+$
  - For any proper subset $H \subset G$, $H^+ \neq F^+$

Extraneous attributes

- Computing a canonical basis — eliminate redundant rules

Extraneous attributes

- Computing a canonical basis — eliminate redundant rules
- Look for extraneous attributes
  - Can $A, B \rightarrow C$ be replaced by $A \rightarrow C$?
  - Can $A \rightarrow C, D$ be replaced by $A \rightarrow C$?

Extraneous attributes

- Computing a canonical basis — eliminate redundant rules
- Look for extraneous attributes
  - Can $A, B \rightarrow C$ be replaced by $A \rightarrow C$?
  - Can $A \rightarrow C, D$ be replaced by $A \rightarrow C$?
- $A \rightarrow C$ is stronger than $A, B \rightarrow C$
  - Check that $A \rightarrow C$ is already in F^+

Check if $C \in A^+$

Extraneous attributes

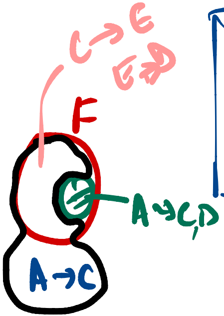
- Computing a canonical basis — eliminate redundant rules
- Look for extraneous attributes
  - Can $A, B \rightarrow C$ be replaced by $A \rightarrow C$?
  - Can $A \rightarrow C, D$ be replaced by $A \rightarrow C$?
- $A \rightarrow C$ is stronger than $A, B \rightarrow C$
  - Check that $A \rightarrow C$ is already in F^+
- $A \rightarrow C$ is weaker than $\underline{A \rightarrow C, D}$
  - Let $G = (F \setminus \{A \rightarrow C, D\}) \cup \{A \rightarrow C\}$
  - Check that $A \rightarrow C, D$ is in G^+

A^+ wrt G contains C, D

stud → fac

⇑

stud, dept → fac



s1, d1 → f1

s1, d2 → f2

s1 → f1, f2

Dependency preservation, formally

- Given a set of dependencies F and a decomposition of R as $R_1, R_2, \ldots, R_k$

Dependency preservation, formally

- Given a set of dependencies F and a decomposition of R as $R_1, R_2, \ldots, R_k$
- Can locally check a depen*d*ency $\alpha \rightarrow \beta$ in R_i if $\alpha \cup \beta \subseteq R_i$

Dependency preservation, formally

- Given a set of dependencies F and a decomposition of R as $R_1, R_2, \ldots, R_k$
- Can locally check a depenency $\alpha \rightarrow \beta$ in R_i if $\alpha \cup \beta \subseteq R_i$
- Let F_i be set of dependencies in F^+ locally checkable in R_i

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

$A \rightarrow C$ is in F^+

$G = \{A \rightarrow C, A \rightarrow B\}$

A	C

A	B

Is $G^+ = F^+$? X

Dependency preservation, formally

- Given a set of dependencies F and a decomposition of R as $R_1, R_2, \ldots, R_k$
- Can locally check a depenency $\alpha \rightarrow \beta$ in R_i if $\alpha \cup \beta \subseteq R_i$
- Let F_i be set of dependencies in F^+ locally checkable in R_i
- Let $G = F_1 \cup F_2 \cup \cdots \cup F_k$. Is $G^+ = F^+$

Dependency preservation, formally

- Given a set of dependencies F and a decomposition of R as $R_1, R_2, \ldots, R_k$
- Can locally check a depenency $\alpha \to \beta$ in R_i if $\alpha \cup \beta \subseteq R_i$
- Let F_i be set of dependencies in F^+ locally checkable in R_i
- Let $G = F_1 \cup F_2 \cup \cdots \cup F_k$. Is $G^+ = F^+$
- How do we compute F_i for each R_i?
  - Let R_i have attributes $A_1, A_2, \ldots, A_m$
  - For each subset α of $A_1, A_2, \ldots, A_m$, compute α^+ with respect to F^+
  - For each $B \in \alpha^+ \cap \{A_1, A_2, \ldots, A_m\}$, add $\alpha \to B$ to R_i

$$\frac{\begin{array}{c} A \to B \\ B \to C \end{array}}{A \to C}$$

R_i

A	C

A^+?

Add C via $A \to C$

Beyond functional dependencies

- Suppose we collect emergency contact details for each students — phone and email
  - At least two emergency contacts of each type

Beyond functional dependencies

- Suppose we collect emergency contact details for each students — phone and email
  - At least two emergency contacts of each type
- Consider a table `Emergency(student_id,phone,email)`
  - Two phone numbers and two emails will generate four rows

sid, p1, e1
sid, p2, e2

sid, p1, e2
sid, p2, e1

Beyond functional dependencies

- Suppose we collect emergency contact details for each students — phone and email
  - At least two emergency contacts of each type
- Consider a table `Emergency(student_id,phone,email)`
  - Two phone numbers and two emails will generate four rows
- This redundancy cannot be explained in terms of functional dependencies

Multivalued dependencies

- Closure under swaps

α	β	γ
a	b_1	c_1
a	b_2	c_2

R

$\implies$

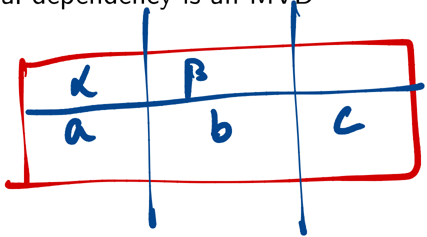
$a \quad b_1 \quad c_2$

$a \quad b_2 \quad c_1$

$\gamma = R \setminus (\alpha \cup \beta)$

Multivalued dependencies

- Closure under swaps
- Every functional dependency is an MVD



α is a "key"

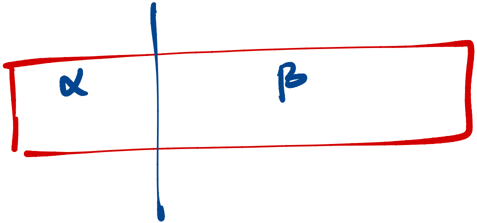
$\alpha \rightarrow \beta$ is an fd

$\alpha \twoheadrightarrow \beta$ is an MV

Multivalued dependencies

- Closure under swaps
- Every functional dependency is an MVD
- Trivial MVD

No γ



$\gamma = \phi$

$\alpha \twoheadrightarrow \beta$

$R, \quad \alpha, \beta \subseteq R$

4NF

- Relational schema R, set D of functional and multivalued dependencies
- R is in 4NF if, for every $\alpha \twoheadrightarrow \beta \in D^+$, one of the following holds
  - $\alpha \twoheadrightarrow \beta$ is trivial (i.e., $\beta \subseteq \alpha$) or $\alpha \cup \beta = R$
  - α is a superkey for R

4NF

- Relational schema R, set D of functional and multivalued dependencies
- R is in 4NF if, for every $\alpha \twoheadrightarrow \beta \in D^+$, one of the following holds
  - $\alpha \twoheadrightarrow \beta$ is trivial (i.e., $\beta \subseteq \alpha$)
  - α is a superkey for R
- 4NF decomposition as usual

stud | phone | email

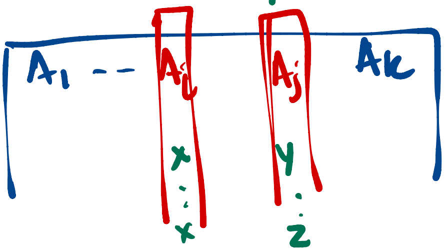
stud | phone

stud | email

Cannot directly describe FDs in SQL

Can we (manually) check an FD using SQL

Table



$A_i \to A_j$ is an FD

fac d1

:

fac d2

Count rows in (unique) projection