CS 281 Lecture Notes
14.04.2015
Çiğdem Gizem KOCA

Minimal Cover Computation

T= ={ABH→CK, A→D, C→E, BGH→F, F→ AD, E→F, BH→E}

<u>Step 1:</u> Make RHS of each functional dependency a single attribute.

- ABH→CK
  ABH→K
  ABH→C

- F→AD   A→D,   A+ = AD
  F→A
  F→D

<u>Step 2:</u> Eliminate redundant attributes from LHS.
ABH→CK
How can we simplfy this?
ABH+   AB,AH,BH
AB+ → ABD   ABH+ = BH+ , therefore we can simplify ABH with BH.
AH+ → AHD
BH+ → BHEFADCK

Do the same for BGH, it is possible to have less number of attributes which has the same number of coverage of BGH+

<u>Step 3:</u> Eliminate implied FDs!

BCNF Decomposition

<u>Ex</u>: R(A,B,C,D,E,)
F={D→B, CE→A}
R1(D,B)   Now, we have covered only B and C.
R2(A,C,E,D)
R21(C,E,A) is in BCNF.
R22(C,E,D) is in BCNF.

<u>Ex:</u> R(A,B,C,D)
FD={AB→C, C→D, D→A}
Is R1(A,B,C) in BCNF? No, because C→A.
R11(A, C) is in BCNF.
R12(B,C) is in BCNF.
Is R2(C,D) in BCNF? Yes.

- R11, R12 and R2 are in BCNF.
- This is ‘not dependency preserving since right hand sides and left hand sides are not in the same paranthesis.

Lossless Decomposition

Goal is to eliminate redundancy. Decomposition must be lossless. Lossless means joining the final relations and still obtaining the initial relation/s.

R→ (R1, R2, …….., Rn)   ri instance of relation of Ri

A binary decomposition of (R, F) into R1 and R2 is lossless iff (R1 ∩ R2) → R1 or (R1 ∩ R2) → R2.

- The attributes common to R1 and R2 must contain a key for R1 and R2. Key ia a superkey in this case.

<u>Ex:</u> R(A,B,C,D)   F={A→BC}
R1(A,B,C)   R2(A,D)
A+ = ABC
A is the common attribute and key of R1, therefore we have a lossless decomposition.

<u>Ex:</u> student (sno, name, dpt) , stu1 (sno, name) , stu2 (name, dpt)
Name is a common attribute but not a key in any of the relations so we have a lossy decomposition.

Dependency Preservation

$$
(R, F) \begin{array}{l} \nearrow (R1, F1) \\ \\ \searrow (R2, F2) \end{array}
$$

The decomposition is dependency preserving iff F and F1 U F2 are equivalent.