Lab 2

Functional dependencies and Normal forms



**EXERCISES**

1. Consider the relation scheme with attributes S (store), D (department), I (item), and M (manager), with functional dependencies SI**** D and SD**** M.
   1. Find all keys for SDIM.

key is SI

* 1. Show that SDIM is in second normal form but not third normal form.

DIM not depend on 3NF, but depend on 2NF, because of SD→M violation 3NF, 011

1. Consider the relation scheme with attributes CITY, ST, and ZIP, which we here abbreviate

C, S, and Z. We observed the dependencies CS**** Z and Z**** C. The decomposition of the relation scheme CSZ into SZ and CZ has a lossless join. Does this decomposition preserve dependencies?

F = {CS → Z, Z →C}

R1 = {SZ}, F1 = Ø

R2 = {CZ}, F2 = {Z→C}

The reference of F on SZ and CZ has only Z → C and no CS → Z. Therefore, the

separation function does not preserve dependency.

1. Let F = {AB**** C, A**** D, BD**** C}.
   1. Find a minimal basic for F. { A**** D, BD**** C }

F = {CS → Z, Z →C}

R1 = {SZ}, F1 = Ø

R2 = {CZ}, F2 = {Z→C}

The reference of F on SZ and CZ has only Z → C and no CS → Z. Therefore, the

separation function does not preserve dependency.

* 1. Give a 3NF, dependency-preserving decomposition of ABCD into only two schemes (with respect to the set of functional dependencies F).

we split into functional dependencies:

- R1 (AD), F1 = {A→ D} and

- R2 (BCD), F2 = {BD→ C}

* 1. What are the projected dependencies for each of your schemes?

-Reference of F to F1 is: A→ D

- The reference for F on F2 is: BD→ C

* 1. Does your answer to (b) have a lossless join? If not, how could you modify the database scheme to have a lossless join and still preserve dependencies?

keys of F are: AB and no properties of F1 and F2 are super keys.

→T add dependencies R3 (AB) -> R (AD, BCD, AB)

1. Let F = {AB**** C, A**** B}.
   1. Find a minimal basic for F.

Minimum function dependency of F = {A→BC}

* 1. When (a) was given on an exam at a large western university, more than half the class answered G = {A**** B, B**** C}. Show that answer is wrong by giving a relation that satisfies F but violates G.

dependency of G function is wrong because in dependency of F function, we can

not determine B→C dependency, but in G there is B→C dependence, this is a

violation

1. Suppose we are given relation scheme ABCD with functional dependencies

{(A**** B, B**** C, A**** D, D**** C}. Let p be the decomposition (AB, AC, BD).

* 1. Find the projected dependencies for each of the relation schemes of p.
* the reference of F on dependency (AB) is: A→B
* the reference for F on dependencies (AC) is: A→C
* the reference for F on dependencies (BD) is: Ø
  1. Does p preserve the given dependencies?

Because we cannot deduce B→C and D→C from the reference of F on decomposition

dependencies, there is no functional dependency guarantee

1. Consider the relation scheme ABCD with dependencies

F={A ****B, B ****C, D ****B]

We wish to find a lossless-join decomposition into BCNF.

1. Suppose we choose, as our first step, to decompose ABCD into ACD and BD. What are the projected dependencies in these two schemes?

The projected dependency in R1(A,C,D) OF R(ABCD): F1{ A→C;D→C} VÌ

{A→B,B→C} VÀ {B→C,D→B}

The projected dependency in R2(B,D) OF R(ABCD): F1{ B→D}

1. Are these schemes in BNCF? If not, what further decomposition is necessary?