CSC343 Prep 10

Junming Zhang

July 2019

---

1. Consider relation $R(A, B, C, D, E, F)$ with functional dependencies S.

$$S = \{CD \to A, B \to EF, A \to BC, F \to D\}$$

   (a) Which functional dependencies indicate a violation of BCNF?
   Apply closure test to check if there is any LHS of a FD not a superkey.

$$CD^+ = ABCDEF$$
$$B^+ = BDEF$$
$$A^+ = ABCDEF$$
$$F^+ = DF$$

   Therefore, FDs $B \to EF$ and $F \to D$ violate BCNF.

   (b) Create an instance of R that satisfies its FDs and has redundant data. Identify the redundancy, and explain what it has to do with the FDs.

A	B	C	D	E	F
1	2	3	4	5	6
7	8	9	4	10	6

   Here is an instance of relation that satisfies all the FDs. Note from the FD $F \to D$, if two tuples share the same value for F, then they have the same value on D. The duplicates are allowed since the closure F^+ does not cover all the attributes in this relation (so as B^+), which means there might be a redundancy.

   (c) Suppose we use the functional dependency $B \to EF$ in the first step of the BCNF decomposition algorithm to decompose R. What two new relations will replace R?

$$R_1(B, D, E, F)$$
$$R_2(A, B, C)$$

   (d) Project the FDs onto these two relations.

   - Step 1. Find closures of attributes in R_1 that yield valid FDs.
     Result:

$$B^+ = BDEF$$
$$F^+ = FD$$

- Step 2. Project FDs in the relation R_1 by closures found in step 1.
  Result:

$$B \to DEF$$
$$F \to D$$

- Step 3. List the set of projections in R_1.
  Result: $\{B \to DEF, F \to D\}$

- Step 4. Find closures of attributes in R_2 that yield valid FDs.
  Result:

$$A^+ = ABCDEF$$
$$B^+ = BDEF$$
since $CD^+ = ABCDEF$, attribute $C \in R_2$, add C to B, then
$$BC^+ = ABCDEF$$

- Step 5. Project FDs in the relation R_2 by closures found in step 4.
  Result:

$$A \to BC$$
$$BC \to A$$

- Step 6. List the set of projections in R_2.
  Result: $\{A \to BC, BC \to A\}$

(e) Is the new schema, with these two relations, in BCNF, or would we have to recurse and continue decomposing?
The schema for the relation R_1 is not in BCNF and needs to be further decomposed, becuase the FD $F \to D$ violates BCNF (not a superkey). However, R_2 does not need to be further decomposed since every FD in R_2 is a superkey.

2. Suppose we are employing the 3NF synthesis algorithm on a relation $R(A, B, C, D, E)$, and we already have the following minimal basis:

$$S = \{A \to DE, C \to A, E \to A\}$$

(a) List all the keys for relation R
The only key is BC.

(b) How do you know that nothing else is a key?
Initially, list the closures which lead to valid FDs in the relation R, which are:

$$A^+ = ADE$$
$$C^+ = ACDE$$
$$E^+ = ADE.$$

From all closures shown above, notice that only if B is added to C there is a minimal superkey, which is a key for a relation.

$$BC^+ = ABCDE \quad \# \text{ derived from } C^+$$

Addition of B to any other attribute except C does not yield a minimal superkey (or a key), seen from the closures listed above. Also, since B is not determined by any other attribute in R, B must be a part of a key. Therefore, BC is the only key (minimal superkey) and there is nothing else a key.

(c) Show the final schema produced by the 3NF algorithm. Explain your answer in terms of the steps of the algorithm. Do not project the functional dependencies onto the relations, just show the relations.

- Step 1. Verify if S is a minimal basis (The fact that S is a minimal basis is given).

- Step 2. Union X, Y for each $X \rightarrow Y \in S$ to define a new relation.
  Result:

$$R_1(A, D, E)$$
$$R_2(A, C)$$
$$R_3(A, E)$$

Also, delete R_3 since all attributes in R_3 occur in R_1 and R_1 contains more attributes than R_3.

- Step 3. Check if there is any new relation a superkey for relation R, if not, add a relation with a schema as a key for the relation R.
  Result: There is no relation a superkey for R. Since BC is a key for R, add a new relation $R_4(B, C)$.

- Step 4. Return the final schema produced by the 3NF algorithm.
  Result: The final schema returned by this algorithm is:

$$R_1(A, D, E)$$
$$R_2(A, C)$$
$$R_4(B, C).$$