Adam Gincel

CS442B

11 December 2016

I pledge my honor that I have abided by the Stevens Honor System.

Assignment 4

1. Consider a relation R(A, B, C, D, E) with

   FD = {AB -> C, CD -> E, C-> A, C -> D, D -> B}

   Scratch work:

   AB -> {ABCDE}

   AC -> {ABCDE} (redundant because of C)

   CD -> {ABCDE} (redundant because of C)

   AD -> {ABCDE}

   C -> {ABCDE}

   D -> {BD}

   E -> {E}

   Keys:

   AB -> {ABCDE}

   AD -> {ABCDE}

   C -> {ABDCE}

2. Consider a relation schema R(A, B, C, D, E, F) with

   FD = {AB -> C, AD -> B, C -> B, F -> AD, F-> E}

   a. Use Armstrong's Axioms to prove F is a superkey of R.

      F+ -> {F} (Reflexivity)

      F+ -> {ADF} (F-> AD) (given)

      F+ -> {ADEF} (F -> E) (given)

      F+ -> {ABDEF} (AD -> B) (Transitive)

      F+ -> {ABCDEF} (F -> A & F -> B because of Decomposition) (AB -> C) (Transitive)

      F is a superkey of R.

   b. Construct a BCNF decomposition of R.

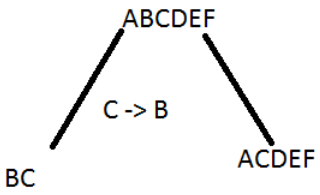
      i. Find candidate keys.

Left	Middle	Right
F	A, B, C, D	E

      F is a superkey alone.

      ii. Decompose the ones that are not a superkey.

      AB -> C, AD -> B, and C -> B are all violations. We try to pull them out from ABCDEF.



      This leaves us with BC and ACDEF.

iii. Is this lossless?

BC intersect ACDEF yields us C, which by the given FD (C -> B) can give us BC back, proves that this BCNF decomposition is lossless, as all BCNF decompositions are by definition.

iv. Is this dependency preserving?

If there exists a given FD that cannot be performed in the decomposed tables without joining them, the decomposition is not dependency preserving. One given FD is (AB -> C). BC cannot do this, as A is not one of its members. ACDEF cannot do this, as B is not one of its members. Without joining these tables, we are unable to preserve this dependency. As such, this BCNF decomposition is lossless, but not dependency preserving.

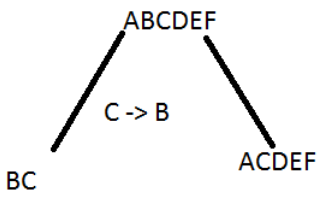
c. Produce a lossless, dependency preserving 3NF decomposition for R.

i. Find the minimal cover for R.

Minimize the right side: F' = {F -> A, F -> D, F -> E, AB -> C, C -> B, AD -> B}

The left side cannot be further minimized. None of these FDs are redundant, as such F' is our minimal cover for R.

ii. BCNF Decomposition



We decompose into BC and ACDEF. D = {BC, ACDEF}

iii. Identify dependencies N in F' not preserved

N = {AB -> C, AD -> B}

iv. For each X -> A in N, create a relation schema XA and add it to D

D = {BC, ACDEF, ABC, ADB}

With step iv complete, D is now a lossless, dependency preserving 3NF decomposition of scheme R.