Homework assignment 2

Hyukjoon Yang

1. Given a relation *R(E,F,G,H)* andthe set of functional dependencies (2 points)

*S = {E →F, EG →H, EF →G}*:

1. Find *E+*  
   {EFGH}
2. Is *E* a candidate key? Please explain your answer.  
   E is a candidate key because it can infer all the attributes in the set and there isn’t any subset of E.
3. Given a relation *R(A, B, C, G, H, I)* and the set of functional dependencies

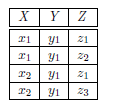
S = {A → B, A → C, CG → H, CG → I, B → H} (2 points)

1. Find (AG)+{A, B, C, G, H, I}
2. Is AG a candidate key? Please explain your answer.  
   AG is a candidate key because the key can infer all attributes in the set. Furthermore, A and G cannot infer all the attributes by alone so there is no subset of AG
3. Consider the relation schema *R(A,B,C)*, which has the FD:  *B → C.* If A is a candidate

key for R, is it possible for R to be in BCNF? If so, under what conditions?

If not, explain why not. (2 points)  
No it is not possible for R to be in BCNF. Only way to make this relation be in BCNF only if there exists FD A → C in the relation schema. Then the relation can be decomposed to be in BCNF.

1. (2 points)

a. List all the functional dependencies that the following relation instance satisfies:  
S = {X → Y, Z →Y, XZ → Y}

b. Assume that the value of attribute *Z* of the last record in the relation is changed

from z3 to z2. Does your list of all the functional dependencies in (a) change?   
 Does not change the functional dependencies because X implies Y, Z implies Y, and XY implies Y after the change of the value of attribute Z

1. (2 points) Suppose that we have the following three tuples in a valid instance of a relation schema *R* with three attributes (A,B,C): (1,6,7), (4,6,7), and (5,7,7).

Specify whether each of the following FDs holds over *R*?

(a) A → B  
Hold

(b) BC → A

Does not hold

(c) B → C  
Hold

1. (3 points) Given a relation R(V,W,X,Y,Z) and its functional dependencies:

*V → W*

*WX → Z*

*ZY → V*

provide answers to the following:

1. List all keys for *R*.  
   Candidate key: VXY, WXY, ZXY  
   Super key: Any combination of attributes that include these candidate keys
2. Is *R* in 3NF?  
   Yes
3. Is *R* in BCNF?  
   No because to be BCNF, all left-hand side of FDs must be super key but V *→W* is not a super key
4. (4 points) Suppose you are given a relation *R* with four attributes, *R*(*A,B,C,D)* and its FDs:

*AB → C*

*AB → D*

*C → A*

*D → B*

Assuming these are the only FDs that hold in *R*, answers the following:

1. Identify the candidate key(s) for *R*.   
   AB, CD, AD, BC
2. Identify all the normal forms that *R* satisfied (1NF, 2NF, 3NF, or BCNF).   
   1NF, 2NF, 3NF
3. If *R* is not in BCNF, decompose it into a set of BCNF relations that preserve the dependencies.  
   There is no decomposition. C→A and D→B do not satisfy conditions to be BCNF but if we decompose, the functional dependencies (AB→C and AB*→*D) do not fit into the decomposition. Therefore, the dependencies are lost. (CA)(DB)(CD)
4. (4 points) Suppose you are given a relation *R(A,B,C,D)*. For each of the following sets of FDs (labelled as (i), (ii), (iii) below), assuming they are the only dependencies that hold in *R*, do the following:
5. Identify the candidate key(s) in *R*.
6. State whether the proposed decomposition of *R* is good or not, and briefly explain why or why not.

i. *B → C, D → A*; decompose into *BC* and *AD*.  
a. BD  
b. Good decomposition because it preserves FDs

ii. *AB → C, C → A, C → D*; decompose into *ACD* and *BC*.  
 a. AB, BC  
 b. Bad decomposition because FD AB → C is not preserved in the decomposition

iii. *A → BC*, *C → AD*; decompose into *ABC* and *AD*.  
 a. A, C  
 c. Bad decomposition due to FD C → AD is not preserved in the decomposition

1. (4 points)  
   Consider the relation *R(A, B, C, D, E)* with the set of FDs:   
   F = {*AB → E*, *CD* *→E*, *A* *→ C*, *C* *→* *A*}.   
   Decompose *R* into BCNF. Write and explain all steps in your decomposition.  
     
   Candidate keys ABD, and BCD  
     
   AB→E violates BCNF because AB is not a key.   
   AB+ = AB, E  
   Then relation (ABE)(ABCD) is created.  
     
   CD→E violates BCNF because CD is not a key.   
   But since E is not in ABCD and CD is not in ABE, the FD does not apply.  
     
   A→C violates BCNF because A is not a key.   
   A+= A, C  
   For ABCD, we create (AC)(ABD).  
     
   C→A does not violate BCNF because C can be a key in AC.  
     
   Therefore, the final decomposition is (ABE)(AC)(ABD)  
   This decomposition is not dependency-preserving relation.