Alan Moss

CSCD 327

Lab 7

1. (6 points) A given relation R = {A, B, C, D, E} is decomposed into three relations:

R1 = {A, B, C}, R2 = {B, C, D}, and R3 = {A, C, E}

a. Based on the given set of FDs F={B->E, CE->A}, is the above decomposition a lossless-join decomposition? Why?

	A	B	C	D	E
R1	α	α	α	β	β
R2	β	α	α	α	β
R3	α	β	α	β	α

Here is a table that was filled out based on the initial decomposed relations. We use the functional dependencies to update the rows, with alpha values being positive and beta being negative. In the case a whole row has all alpha values, we can say the decomposition is lossless-join. Because we had no updates to make based on the functional dependencies, we have no rows with all alpha values, so ***this decomposition is lossy (not lossless).***

b. Based on the given set of FDs F={AC->E, BC->D}, is the above decomposition a lossless-join decomposition? Why?

	A	B	C	D	E
R1	α	α	α	α	α
R2	β	α	α	α	β
R3	α	β	α	β	α

Here we can see the highlighted row has been switched to all alphas. This is because the functional dependencies have changed. The functional dependency AC->E allows up to put an alpha in $R1_E$ at the top right , since we have R3 with alphas at A,C and E. The functional dependency BC->D allows us to put an alpha at $R1_D$ since we have R2 with alphas at B,C, and D.

Because we have a row with all alpha values, we can say ***this decomposition is lossless-join***.

2. (10 points) A given relation R = {A, B, C, D, E}, and a given set of FDs F = {AB->C, DE->C, B->D}.

Candidate key = ABE

a. Is R in BCNF? If not, do the decomposition accordingly.

Candidate key: ABE

To determine if R is in BCNF, check each non-trivial dependency α -> β. Compute attribute closure of alpha and verify that it includes all attributes of R, that is it is a superkey of R. If α^+ is not a superkey, then R is not in BCNF.

We can check if R is in BCNF quickly by checking the left-hand side of the functional dependencies. The left-hand side should be a superkey of the relation, but since none of the functional dependencies' left side is a superkey, ***the relation is not in BCNF.***

R

R1: A,B,E	R2: B,D	R3: A,B,C
FD: none	FD: B->D	FD: AB->C

b. Is your decomposition a lossless-join decomposition? Why?

https://www.geeksforgeeks.org/lossless-join-and-dependency-preserving-decomposition/

In order to be lossless:

1) The union of attributes of our decomposition (R1, R2, and R3) must be equal to the original relation (R)'s attributes
2) Intersection of attributes of R1, R2, and R3 must not be null
3) The common attribute between R1, R2, and R3 must be a key for at least one relation.

All of these cases are satisfied, with B being the key for R2. Therefore ***my decomposition is lossless.***

c. Is your decomposition a dependency-preserving decomposition? Why?

This decomposition is not dependency-preserving, because DE->C was not preserved. I.e. there is no relation with D, E, and C.

d. List all the candidate keys of relation R.

A B E

e. Is R in the 3rdNF? Why?

https://www.geeksforgeeks.org/third-normal-form-3nf/

R is not in the 3rdNF. For a relation to be in 3NF, one of the following conditions must hold true for every non-trivial functional dependency X->Y:

X is a super key

Y is a prime attribute

It is easiest to see if a relation is in 3NF if there are partial dependencies in the sets of functional dependencies. According to https://www.tutorialspoint.com/Partial-Dependency-in-DBMS "Partial Dependency occurs when a non-prime attribute is functionally dependent on part of a candidate key." We can see there are two partial dependencies in FD set, AB->C and B->D. 2NF gets rid of partial dependencies, and a relation must be in 2NF in order to be in 3NF. Therefore we can say that ***R is not in the 3rdNF.***