CSCC43 Tutorial #8

Functional Dependencies

Andrew Leung

Part 1 – Functional Dependencies

1. Suppose the functional dependency BC → D holds in R(A,B,C,D). Create an instance of R that violates this FD.

In order to violate this FD, we need two tuples with the same value for B and the same value for C (both!), yet different values for D.

Eg:

A	B	C	D
1	3	6	4
2	3	6	5

Part 1 – Functional Dependencies

2. a) Are the sets A → BC and A → B, A → C equivalent? If yes, explain why. If no, construct an instance of R that satisfies one set of FDs but not the other.

These are equivalent - there is no instance of the relation that satisfies one but not the other. This can be proven, and now that you know the closure test, the proof is very concise:

• Assume that A → BC.

o Under this assumption, A+ = ABC.

o Therefore A → B, and A → C.

• Assume that A → B, and A → C.

o Under this assumption, A+ = ABC.

o Therefore A → BC.

• Therefore, each set of FDs follows from the other. They are equivalent.

In fact, we can always “split the RHS" of an FD. Review the definition of an FD to see why this makes sense.

Part 1 – Functional Dependencies

2. b) Are the sets $PQ \to R$ and $P \to R$, $Q \to R$ equivalent? If yes, explain why. If no, construct an instance of R that satisfies one set of FDs but not the other.

These are not equivalent, as demonstrated by this instance that satisfies $PQ \to R$ but not $P \to R; Q \to R$:

P	Q	R
1	2	4
3	2	5

We can never split the LHS of an FD. Again, the definition of FD makes clear why.

Part 1 – Functional Dependencies

2. c) Are the sets PQ → R and P → Q; P → R equivalent? If yes, explain why. If no, construct an instance of R that satisfies one set of FDs but not the other.

These are not equivalent, as demonstrated by this instance that satisfies PQ → R but not P → Q; P → R:

P	Q	R
1	2	4
1	8	9

Part 1 – Functional Dependencies

3. a) We claimed that if a set of attributes K functionally determines all attributes, K must be a superkey (i.e., no two tuples can agree on all attributes in K). Do you believe this? Suppose these FDs hold in R: A → BC, C → D. Does A functionally determine all attributes of R? Can two tuples agree on A?

3. b) We also said that if K is a superkey (i.e., no two tuples can agree on all attributes in K) K must functionally determine all attributes. Do you believe this? Suppose A is a superkey of R Does A functionally determine all attributes of R?

These are left as an exercise for you to explore on your own, in order to build your intuition.

Part 1 – Functional Dependencies

4. Suppose we have a relation on attributes ABCDEF with these FDs:

AC → F, CEF → B, C → D, DC → A

a) Does it follow that C → F?
b) Does it follow that ACD → B?

We use the closure test to check whether an FD follow from a set of FDs.

C+ = CDAF. Therefore, C → F does follow.

ACD+ = ACDF. Therefore, ACD → B does not follow.

Part 1 – Functional Dependencies

5. Suppose we have a relation on attributes ABCDE with these FDs:

A → C, C → E, E → BD

a) Project the FDs onto attributes ABC

- A+ = ACEBD, therefore A → BC. (It also functionally determines DE, but these are not in our set of attributes. And it functionally determines itself, but we don't need to write down dependencies that are tautologies.)
- B+ = B. This yields no FDs for our set of attributes.
- C+ = CEBD, therefore C → B.
- We don't need to consider any supersets of A. A already determines all of our attributes ABC, so supersets of A will be only yield FDs that already follow from A → BC.
- The only remaining subset of the attributes ABC to consider is BC. BC+ = BCED. This yields no FDs for our set of attributes.
- So the projection of the FDs onto ABC is: {A → BC, C → B}.

Part 1 – Functional Dependencies

5. Suppose we have a relation on attributes ABCDE with these FDs:

A → C, C → E, E → BD

b) Project the FDs onto attributes ADE

- A+ = ACEBD, therefore A → DE.
- D+ = D. This yields no non-trivial FDs..
- E+ = EBD, therfore E → D.
- Again, we don't need to consider any supersets of A, since A determines all the attributes ADE also.
- The only remaining subset of the attributes ABC to consider is DE. DE+ = DEB. This yields no FDs for our set of attributes.
- So the projection of the FDs onto ADE is: {A → DE, E → D}.

Part 2 – Projection and Minimal Basis

1. Suppose we have these FDs: S = {ABE → CF, DF → BD, C → DF, E → A, AF → B}. Project the FDs onto L = CDEF.

C	D	E	F	closure	FDs
✓				$C^+ = CDFBD$	$C \to DF$
	✓			$D^+ = D$	nothing
		✓		$E^+ = EA$	nothing
			✓	$F^+ = F$	nothing
✓	✓			$CD^+ = CDFB$	nothing, since $CD \to DF$ is weaker than $C \to DF$ which we have already
✓		✓		$CE^+ = CEDFAB$	nothing, since $CE \to DF$ is weaker than $C \to DF$ which we have already
✓			✓	$CF^+ = CFDB$	nothing, since $CF \to D$ is weaker than $C \to DF$ which we have already
	✓	✓		$DE^+ = DEA$	nothing
	✓		✓	$DF^+ = DFB$	nothing
		✓	✓	$EF^+ = EFABCD$	$EF \to CD$
✓	✓	✓		$CDE^+ = CDEF$	nothing, since $CDE \to F$ is weaker than $C \to DF$ which we have already
✓	✓		✓	$CDF^+ = CDFB$	nothing
✓		✓	✓	since EF is a key, supersets of EF can only yield FDs that are weaker than ones we have.	
	✓	✓	✓	since EF is a key, supersets of EF can only yield FDs that are weaker than ones we have.	

Final answer: The projection of S onto L is C → DF, EF → CD.

Part 2 – Projection and Minimal Basis

2. Find a minimal basis for this set of FDs: S = {ABF → G, BC → H, BCH → EG, BE → GH}.

Step 1: Split the RHSs to get our initial set of FDs, S1:

(a) ABF → G

(b) BC → H

(c) BCH → E

(d) BCH → G

(e) BE → G

(f) BE → H

Part 2 – Projection and Minimal Basis

2. Find a minimal basis for this set of FDs: S = {ABF → G, BC → H, BCH → EG, BE → GH}.

Step 2: For each FD, try to reduce the LHS:

(a) A+ = A, B+ = B, F+ = F. In fact, no singleton LHS yields anything. AB+ = AB, AF+ = AF, and BF+ = BF, so none of them yields G either. We cannot reduce the LHS of this FD.

(b) Since this FD has only two attributes on the LHS, and no singleton LHS yields anything, we cannot reduce the LHS of this FD.

(c) Since no singleton LHS yields anything, we need only consider LHSs with two or more attributes. We only have three to begin with, so that leaves LHSs with two attributes. BC+ = BCHEG. So we can reduce the LHS of this FD, yielding the new FD: BC → E.

(d) By the same argument, we can reduce this FD to: BC → G.

(e) Since no singleton LHS yields anything, we cannot reduce the LHS of this FD.

(f) Since no singleton LHS yields anything, we cannot reduce the LHS of this FD.

Part 2 – Projection and Minimal Basis

2. Find a minimal basis for this set of FDs: S = {ABF → G, BC → H, BCH → EG, BE → GH}.

Our new set of FDs, let's call it S2, is

(a) ABF → G

(b) BC → H

(c) BC → E

(d) BC → G

(e) BE → G

(f) BE → H

Part 2 – Projection and Minimal Basis

2. Find a minimal basis for this set of FDs: S = {ABF → G, BC → H, BCH → EG, BE → GH}.

Step 3: Try to eliminate each FD.

(a) $ABF^{+}_{S2-(a)} = ABF$. We need this FD.

(b) $BC^{+}_{S2-(b)} = BCEGH$. We can remove this FD.

(c) $BC^{+}_{S2-\{(b),(c)\}} = BCG$. We need this FD.

(d) $BC^{+}_{S2-\{(b),(d)\}} = BCEGH$. We can remove this FD.

(e) $BE^{+}_{S2-\{(b),(d),(e)\}} = BEH$. We need this FD.

(f) $BE^{+}_{S2-\{(b),(d),(f)\}} = BEG$. We need this FD.

Part 2 – Projection and Minimal Basis

2. Find a minimal basis for this set of FDs: S = {ABF → G, BC → H, BCH → EG, BE → GH}.

Our final set of FDs is:

(a) ABF → G

(c) BC → E

(e) BE → G

(f) BE → H

Part 3 – The Chase Test

1. Suppose we have a relation on attributes NFLCG with these FDs:

$N \rightarrow FL$, $NC \rightarrow G$

a) Suppose we decompose into relations NF, FLC and LCG. Use the Chase Test to determine whether this is a lossless-join decomposition.

Initial tableau:

N	F	L	C	G
n	f	l1	c1	g1
n2	f	l	c	g2
n3	f3	l	c	g

If you apply both FDs, the tableau does not change so this is a lossy-join decomposition.

Part 3 – The Chase Test

1. Suppose we have a relation on attributes NFLCG with these FDs:

N → FL, NC → G

b) Suppose we decompose into relations NF, NL and NCG. Use the Chase Test to determine whether this is a lossless-join decomposition.

Initial tableau:

N	F	L	C	G
n	f	l1	c1	g1
n	f2	l	c2	g2
n	f3	l3	c	g

Part 3 – The Chase Test

1. Suppose we have a relation on attributes NFLCG with these FDs:

N → FL, NC → G

b) Suppose we decompose into relations NF, NL and NCG. Use the Chase Test to determine whether this is a lossless-join decomposition.

Apply N → FL:

N	F	L	C	G
n	f	l	c1	g1
n	f	l	c2	g2
n	f	l	c	g

<n,f,l,c,g> appears as a row so this is a lossless-join decomposition

Part 3 – The Chase Test

1. Suppose we have a relation on attributes NFLCG with these FDs:

N → FL, NC → G

c) Suppose we decompose into relations NFC, and NLG. Use the Chase Test to determine whether this is a lossless-join decomposition.

Initial tableau:

N	F	L	C	G
n	f	l1	c	g1
n	f2	l	c2	g

Part 3 – The Chase Test

1. Suppose we have a relation on attributes NFLCG with these FDs:

N → FL, NC → G

c) Suppose we decompose into relations NFC, and NLG. Use the Chase Test to determine whether this is a lossless-join decomposition.

Apply N → FL:

N	F	L	C	G
n	f	l	c	g1
n	f	l	c2	g

Applying NC → G does not change this tableau so this is a lossy-join decomposition

Part 3 – The Chase Test

2. Suppose we have a relation on attributes ABCDEF and it is to be decomposed into relations ABCD and DEF.

a) Invent a set of FDs that would make this a lossless-join decomposition.

There are many solutions to this question. A simple solution is the single functional dependency $D \rightarrow ABC$. The set of functional dependencies $D \rightarrow BF$, $F \rightarrow C$, $BC \rightarrow A$ is a more complicated solution.

Part 3 – The Chase Test

2. Suppose we have a relation on attributes ABCDEF and it is to be decomposed into relations ABCD and DEF.

b) Invent a set of three FDs that would make this is a lossy-join decomposition.

Again, there are many solutions to this question. The simplest is just the empty set. It is perhaps more fun to come up with a solution that has a bunch of FDs. The set $D \rightarrow AB$, $E \rightarrow C$, $F \rightarrow EA$ is one such solution.

Part 3 – The Chase Test

2. Suppose we have a relation on attributes ABCDEF and it is to be decomposed into relations ABCD and DEF.

c) If there were no FDs at all, is it possible that the decomposition is lossless?

This particular decomposition is lossy if there are no functional dependencies. And in fact this holds for any non-trivial decomposition, that is, any decomposition into 2 or more relations, none of which includes all the attributes of the original relation.

Important: In practice, one never invents FDs! They are facts about the domain that either hold or don't hold. So this question is completely unrealistic, but if you can solve it, you really understand the Chase Test.