Andrew Plumm
Prof. Jamil
CS 360
11/11/2023

Assignment #7

1) Given $R = \{ABCDELGHIJK\}$;
$F = \{I \to K, AI \to BLG, IC \to ADE, BIG \to CJ, K \to HA\}$

a) Find a canonical cover of F

X→A form: $\{I \to K, AI \to B, AI \to L, AI \to G, IC \to A, IC \to D, IC \to E,$
$BIG \to C, BIG \to J, K \to H, K \to A\}$

Remove redundant FDs:
$\{I \to K, AI \to B, AI \to L, AI \to G, IC \to D, IC \to E, BIG \to C, BIG \to J,$
$K \to H, K \to A\}$

Remove left redundant FDs:
$\{I \to K, I \to B, I \to L, I \to G, I \to D, I \to E, I \to C, I \to J, K \to H, K \to A\}$

Answer 1a: Canonical cover of F:
$\{I \to K, I \to B, I \to L, I \to G, I \to D, I \to E, I \to C, I \to J, K \to H, K \to A\}$

b) Find a 3NF decomposition of R
$S = \{IKBLGDECJ, KHA\}$
IKBLGDECJ is a superkey so

Answer 1b: $S = \{IKBLGDECJ, KHA\}$ is a 3NF decomposition of R

c) Is there a BCNF decomposition of R that is both dependency-preserving and also lossless join? If so compute such a decomposition.

① ABCDELGHIJK $\{I \to K, AI \to BLG, IC \to ADE, BIG \to CJ, K \to HA\}$
Key: I ✓ ✓ ✓ ✓ ×

①.1 KHA $\{K \to HA\}$
Key: K ✓

①.2 BCDELGIJK $\{I \to K, AI \to BLG, IC \to ADE, BIG \to CJ\}$

We no longer have HA which is needed so

Answer 1c: BCNF form isn't dependency preserving

Back Side

2) Let $R = \{ABCDEH\}$; $F = \{ABC \to DE, AB \to D, DE \to ABCH, E \to C\}$

a) Is this scheme in 3NF, BCNF, or none?

Candidate keys of R: $\{DE\}$

3NF: $\{ABC \to DE, AB \to D, DE \to ABCH, E \to C\}$
✓ ✓ ✓ ✗

BCNF: $\{ABC \to DE, AB \to D, DE \to ABCH, E \to C\}$
✓ ✗ ✓ ✗

Answer 2a: The scheme is not in 3NF ∴ not in BCNF

b) If it isn't, decompose this scheme into a normal form that is both dependency preserving ∴ loss-less join.

Work: I am going to decompose this scheme only into 3NF because it is always dependency preserving ∴ loss-less join ∴ it never said the repetition isn't fine.

3NF Decomposition of R:

- Find Canonical Cover:
  - ▫ X→A Form: $\{ABC \to D, ABC \to E, AB \to D, DE \to A, DE \to B, DE \to C, DE \to H, E \to C\}$
  - ▫ Remove redundant FDs: $\{ABC \to E, AB \to D, DE \to A, DE \to B, DE \to H, E \to C\}$
  - ▫ Remove left redundant FDs: $\{ABC \to E, AB \to D, DE \to A, DE \to B, DE \to H, E \to C\}$
- Use Canonical Cover to find S: $S = \{ABCE, ABD, ADE, BDE, DEH\}$

ABCE is a superkey so

Answer 2b: $S = \{ABCE, ABD, ADE, BDE, DEH\}$ is a 3NF decomposition of R

3) Consider the scheme $R=\{ABCDEGHI\}$ and the associated functional dependencies
$F=\{A\to BCDEIGH,\ BCD\to AEIGH,\ BCE\to ADEIGH,\ CE\to H,\ CD\to H\}$
a) Find a BCNF decomposition of R. Is the decomposition you computed dependency preserving? Why or why not?

(1) $ABCDEGHI\ \{A\to BCDEIGH,\ BCD\to AEIGH,\ BCE\to ADEIGH,\ CE\to H,\ CD\to H\}$
✓ ✓ ✓ ✗ ✗

(1.1) $CEH\ \{CE\to H\}$
Key: CE ✓

(1.2) $ABCDEGI\ \{A\to BCDEIG,\ BCD\to AEIG,\ BCE\to ADEIG,\ CD\to CD\}$
✓ ✓ ✓ ✗

(1.2.1) $CD\ \{CD\to CD\}$
Key: CD ✓

(1.2.2) $ABCDEGI-CD=ABEGI\ \{A\to BEIG,\ B\to AEIG,\ BE\to AEIG\}$
✓ ✓ ✓

Answer 3: Final BCNF Decomposition: {ABEGI, CD, CEH}

R_1(ABEGI): {A→BEIG, B→AEIG, BE→AEIG}

R_2(CD): {CD→CD}

R_3(CEH): {CE→H}

The BCNF decomposition I computed is not dependency preserving; this can be seen by the resulting set of dependencies differing from the initial set of dependencies. The reason the functional dependencies were different between the initial and resulting set was because H had to be dropped from ABCDEGHI because CE→H didn't satisfy BCNF changing CD→H to CD→CD. Then CD→CD didn't satisfy BCNF so CD was dropped from ABCDEGI which changed all of the previous functional dependencies that had C, D, or CD. After that, all of the functional dependencies satisfied BCNF, but by then most all of them had changed from what they initially were.