Esercizi di Normalizzazione

Stefano Calzavara

Alcune delle soluzioni omettono alcuni dettagli per mantenerne la leggibilità. Siete invitati a svolgere tutti i dettagli con cura durante la vostra soluzione, applicando gli algoritmi visti a lezione.

1. Si consideri $F = \{AB \to C, AC \to B, AD \to E, B \to D, BC \to A, E \to G\}$. Trovare la proiezione di F sull'insieme di attributi ABC e determinare la forma normale più forte soddisfatta da essa.

Solution: Applichiamo l'algoritmo per il calcolo delle proiezioni:

- $A^+ = A$, da cui nessuna nuova dipendenza
- $B^+ = BD$, da cui nessuna nuova dipendenza
- $C^+ = C$, da cui nessuna nuova dipendenza
- $AB^+ = ABCDEG$, da cui la dipendenza $AB \to C$
- $AC^+ = ACBDEG$, da cui la dipendenza $AC \to B$
- $BC^+ = BCDAEG$, da cui la dipendenza $BC \to A$

Otteniamo quindi $\pi_{ABC}(F) = \{AB \to C, AC \to B, BC \to A\}$. Osserviamo che le parti sinistre di ciascuna dipendenza formano una superchiave, di conseguenza tale insieme è in BCNF. Tale forma normale è più forte di 3NF, quindi l'insieme è anche in 3NF.

2. Si consideri uno schema di relazione $R(A, B, C, D, E)$ con il seguente insieme di dipendenze funzionali: $F = \{B \to C, B \to D, B \to E, C \to B\}$. Dimostrare che R non è in 3NF e convertirlo in 3NF.

Solution: Prima di tutto dobbiamo identificare gli attributi primi, calcolando tutte le chiavi. Osserviamo che A non occorre a destra in nessuna dipendenza funzionale, quindi deve fare parte di ogni chiave. Osserviamo poi che:

- $AB_F^+ = ABCDE$, quindi AB è una chiave. Nessuna altra chiave può contenere AB;
- $AC_F^+ = ACBDE$, quindi AC è una chiave. Nessuna altra chiave può contenere AC;
- $AD_F^+ = AD$, quindi AD non è una chiave;
- $AE_F^+ = AE$, quindi AE non è una chiave;
- $ADE_F^+ = ADE$, quindi ADE non è una chiave e la ricerca delle chiavi è finita.

Concludiamo quindi che ci sono solo due chiavi: AB e AC. A questo punto è facile osservare che R non è in 3NF a causa della dipendenza $B \to D$: in particolare B non è una superchiave e D non è primo. Applichiamo quindi l'algoritmo di conversione in 3NF, osservando che l'insieme di dipendenze funzionali F è già in forma canonica:

1. Trasformiamo F in $G = \{B \rightarrow CDE, C \rightarrow B\}$
2. Introduciamo gli schemi $R_1(B, C, D, E)$ e $R_2(B, C)$
3. Eliminiamo lo schema $R_2(B, C)$, poichè esso è contenuto in $R_1(B, C, D, E)$
4. Aggiungiamo lo schema $R_3(A, C)$, poichè nella decomposizione non è presente alcuno schema i cui attributi costituiscono una superchiave.

La conversione in 3NF produce quindi la decomposizione $\{R_1(B, C, D, E), R_3(A, C)\}$.

3. Si consideri uno schema di relazione $I(L, N, S)$ con le dipendenze funzionali $F = \{N \rightarrow L, L \rightarrow S\}$. Dimostrare che I non è in 3NF, nè in BCNF. Convertirlo poi sia in 3NF che in BCNF e dire se la conversione in BCNF ha preservato le dipendenze.

Solution: Per dimostrare che I non è in 3NF dobbiamo prima trovare le sue chiavi. Dato che N non occorre a destra di nessuna dipendenza funzionale, essa deve fare parte di ogni chiave. Osserviamo che $N^+ = NLS$, quindi N è una chiave ed in particolare l'unica chiave. A questo punto possiamo notare che $L \rightarrow S$ viola le condizioni di 3NF, perchè L non è una superchiave e S non è primo. Dato che L non è una superchiave, possiamo anche notare che I non è neppure in BCNF.

Per convertire I in 3NF, lo decomponiamo negli schemi $I_1(L, N)$ e $I_2(L, S)$. In questo caso non è necessario eliminare schemi, perchè nessuno schema è contenuto in un altro, nè aggiungere schemi, perchè LN è una superchiave per lo schema di partenza.

Vediamo ora la conversione di I in BCNF:

1. Partiamo dallo schema I con dipendenze funzionali F. Poichè $L \rightarrow S$ viola le condizioni di BCNF, calcoliamo gli insiemi $T_1 = L^+_F = \{L, S\}$ e $T_2 = \{L\} \cup (\{L, N, S\} \setminus T_1) = \{L, N\}$.
2. Calcoliamo ora le proiezioni delle dipendenze. Dato che $L^+ = LS$, $N^+ = NLS$ e $S^+ = S$, otteniamo $F_1 = \pi_{T_1}(F) = \{L \rightarrow S\}$ e $F_2 = \pi_{T_2}(F) = \{N \rightarrow L\}$.
3. Osserviamo ora che $I_1(T_1, F_1)$ e $I_2(T_2, F_2)$ sono entrambi in BCNF, quindi la decomposizione si ferma. Essa ha preservato le dipendenze, perchè $F_1 \cup F_2 = F$ e quindi $F_1 \cup F_2 \equiv F$.

4. Si supponga che per archiviare dati sull'inventario delle apparecchiature di un'azienda sia stata usata una tabella con la seguente struttura: Inventario(NInventario, Modello, Descrizione, NSerie, Costo, Responsabile, Telefono).

Il numero di inventario identifica un'apparecchiatura. Un'apparecchiatura ha un costo, un modello e un numero di serie. Apparecchiature dello stesso modello possono avere costi differenti, perchè acquistate in momenti diversi, ma hanno la stessa descrizione. Il numero di serie identifica univocamente un'apparecchiatura fra tutte quelle dello stesso modello. Ogni apparecchiatura ha un responsabile, che può avere più apparecchiature, ma un unico numero di telefono.

Definire le dipendenze funzionali e dire se lo schema proposto presenta anomalie, giustificando la risposta. Trasformare poi la rappresentazione in uno schema relazionale in 3NF.

Solution: Traducendo il testo in dipendenze funzionali otteniamo:

1. Il numero di inventario identifica un'apparecchiatura. Un'apparecchiatura ha un costo, un modello e un numero di serie: $NInventario \rightarrow Costo\ Modello\ NSerie$

2. Apparecchiature dello stesso modello hanno la stessa descrizione: $Modello \to Descrizione$
3. Il numero di serie identifica univocamente un'apparecchiatura fra tutte quelle dello stesso modello: $Modello\ NSerie \to NInventario$
4. Ogni apparecchiatura ha un responsabile: $NInventario \to Responsabile$
5. Un responsabile può avere un unico numero di telefono: $Responsabile \to Telefono$

Lo schema presenta varie anomalie, per esempio c'è un'anomalia di aggiornamento sul numero di telefono dei responsabili, come suggerito dall'ultima dipendenza anomala.

Per convertire lo schema in 3NF, costruiamo prima di tutto una copertura canonica delle dipendenze funzionali. Dividiamo $NInventario \to Costo\ Modello\ NSerie$ in tre dipendenze con un singolo attributo a destra, poi verifichiamo che non vi sono attributi estranei e dipendenze ridondanti. Dopo aver unito le dipendenze funzionali con la stessa parte sinistra, otteniamo i seguenti schemi:

- $R_1(NInventario, Costo, Modello, NSerie, Responsabile)$
- $R_2(Modello, Descrizione)$
- $R_3(Modello, NSerie, NInventario)$
- $R_4(Responsabile, Telefono)$

Lo schema R_3 viene eliminato, perché contenuto in R_1. Visto che gli attributi di R_1 sono una superchiave per lo schema di partenza, non è necessario aggiungere altri schemi.

5. Sia $R(T, F)$ uno schema relazionale, con F una copertura canonica. Si dimostri che se lo schema ha una sola chiave ed è in 3NF, allora è anche in BCNF.

Solution: Sia $R(T, F)$ in 3NF e sia Y la sua unica chiave. Assumiamo per assurdo che lo schema non sia in BCNF, deve quindi esistere una dipendenza non banale $X \to A \in F$ tale che X non è una superchiave. Notiamo però che A deve essere primo, dato che lo schema è in 3NF, quindi $A \in Y$. Poiché $X \to A \in F$, si ha che $XY \setminus \{A\} \to Y \in F^+$, pertanto $XY \setminus \{A\}$ deve essere una superchiave per transitività. Ma allora esiste $Z \subseteq XY \setminus \{A\}$ che è una chiave diversa da Y, in quanto $A \notin Z$.

6. Si consideri lo schema $R(P, C, L, A)$ con dipendenze funzionali: $F = \{P \to CLA, CL \to AP, A \to C\}$:
   (a) Trovare una copertura canonica delle dipendenze funzionali;
   (b) Trovare tutte le chiavi della relazione;
   (c) Dire se lo schema è in BCNF e se non lo è convertirlo;
   (d) Dire se lo schema è in 3NF e se non lo è convertirlo.

Solution:

(a) Prima di tutto riscriviamo l'insieme per avere un solo simbolo a destra delle frecce:

$$\{P \to C, P \to L, P \to A, CL \to A, CL \to P, A \to C\}$$

E' facile vedere che non ci sono attributi estranei, perché $C^+ = C$ e $L^+ = L$. Ci sono invece due dipendenze ridondanti, vale a dire $P \to C$ e $CL \to A$. La copertura canonica è perciò:

$$G = \{P \to L, P \to A, CL \to P, A \to C\}$$

(b) Per quanto riguarda le chiavi, osserviamo che tutti gli attributi occorrono a destra di una dipendenza funzionale. Abbiamo quindi:

- $P^+ = PLAC$, quindi P è chiave;
- $A^+ = AC$, quindi A non è chiave;
- $C^+ = C$, quindi C non è chiave;
- $L^+ = L$, quindi L non è chiave;
- $AC^+ = AC$, quindi AC non è chiave;
- $AL^+ = ALCP$, quindi AL è chiave;
- $CL^+ = CLPA$, quindi CL è chiave;
- non ci sono altri candidati da testare, perché essi contengono tutti uno fra P, AL o CL.

L'insieme delle chiavi è quindi $K = \{P, AL, CL\}$.

(c) Lo schema non è in BCNF a causa della dipendenza $A \to C$. Abbiamo quindi:

- $T_1 = A^+ = \{A, C\}$
- $G_1 = \pi_{T_1}(G) = \{A \to C\}$
- $T_2 = \{A\} \cup (\{P, C, L, A\} \setminus \{A, C\}) = \{P, L, A\}$
- $G_2 = \pi_{T_2}(G) = \{P \to L, P \to A, AL \to P\}$ (dopo forma canonica)

A questo punto la decomposizione si ferma, perché i due schemi sono in BCNF.

(d) Lo schema è già in 3NF, perché la dipendenza $A \to C$ che violava BCNF non viola invece 3NF, in quanto l'attributo C è primo.

7. Si consideri $R(A, B, C, D, E, F, G)$ con dipendenze $\{ABC \to DE, BD \to AF, EG \to B, B \to A\}$:
   (a) Trovare una copertura canonica delle dipendenze funzionali;
   (b) Trovare tutte le chiavi della relazione;
   (c) Effettuare la conversione in 3NF usando l'algoritmo di sintesi.

Solution:

(a) Prima di tutto riscriviamo le dipendenze per garantire un solo simbolo a destra delle frecce:

$$\{ABC \to D, ABC \to E, BD \to A, BD \to F, EG \to B, B \to A\}$$

Le dipendenze $ABC \to D$ e $ABC \to E$ contengono l'attributo estraneo A, che può essere eliminato, ottenendo $BC \to D$ e $BC \to E$. Inoltre $BD \to A$ contiene l'attributo estraneo D, in quanto abbiamo già la dipendenza $B \to A$, e quindi essa può essere eliminata. Di conseguenza otteniamo:

$$\{BC \to D, BC \to E, BD \to F, EG \to B, B \to A\}$$

Non ci sono dipendenze ridondanti, quindi questa è una copertura canonica.

(b) Tutte le chiavi devono contenere C e G, dato che tali attributi non occorrono nella parte destra di alcuna dipendenza funzionale. Osserviamo poi che A e F non possono fare parte di nessuna chiave, perchè tali attributi non occorrono nella parte sinistra di alcuna dipendenza funzionale. Calcoliamo quindi le chiavi tramite i seguenti candidati:

- $CG :: (BDE)$: calcolo $CG^+ = CG$, quindi CG non è chiave e genero i nuovi candidati $CGB :: (DE)$, $CGD :: (E)$, $CGE :: ()$;
- $CGB :: (DE)$: calcolo $CGB^+ = CGBDEFA$, quindi CGB è chiave e non genero nuovi candidati;
- $CGD :: (E)$: calcolo $CGD^+ = CGD$, quindi CGD non è chiave e genero il nuovo candidato $CGDE :: ()$;
- $CGE :: ()$: calcolo $CGE^+ = CGEBADF$, quindi CGE è chiave e non genero nuovi candidati;
- $CGDE :: ()$: osservo che $CGDE$ contiene la chiave CGE e quindi lo scarto.

Non ci sono altri candidati da considerare, quindi l'insieme delle chiavi è $K = \{CGB, CGE\}$.

(c) Prima di tutto accorpiamo le dipendenze con la stessa parte sinistra:

$$\{BC \to DE, BD \to F, EG \to B, B \to A\}$$

A questo punto generiamo quattro schemi: $R_1(B, C, D, E)$, $R_2(B, D, F)$, $R_3(B, E, G)$ e $R_4(A, B)$. Nessuno schema è contenuto in un altro, quindi non ne va eliminato nessuno. Visto che nessun insieme di attributi costituisce una superchiave per lo schema di partenza, è necessario aggiungere un quinto schema i cui attributi formino una chiave, per esempio $R_5(B, C, G)$.