

Le forme normali 1/2

Università di Pisa

Lezione del 14 aprile 2021

Contenuti

1 Introduzione

2 Formalizzazione

- In teoria
- In pratica

3 Dipendenze funzionali

- Definizione
- Esempio
- Teoria sulle dipendenze

4 Costruzione della chiusura

- Motivazione

- Regole di inferenza di Armstrong

- Regole derivate di Armstrong

5 Equivalenza fra insiemi di DF

- Il problema dell'equivalenza
- La chiusura di un insieme di attributi
- Big badum-ts! time

6 Ridondanza di un insieme di DF

- Il Problema part.2
- Semplificare l'insieme F
- Copertura minimale

Introduzione

Obbiettivi

- ▶ valutare la qualità della progettazione degli schemi relazionali
- ▶ conservazione dell'informazione
- ▶ minimizzazione della ridondanza

Approccio top-down: raggruppamenti di attributi analizzati e successivamente decomposti.

① **Semplice è bello**

Non raggruppare attributi da più tipi di entità/relazione in un'unica relazione.

② **No alle anomalie**

Certificare che i programmi di inserimento, cancellazione o modifica funzionano sempre.

③ **Evitare valori nulli**

Se inevitabili, assicurarsi che sono pochi rispetto al numero di tuple di una relazione.

Esempio: LG1

- Per mantenere semplicità semanticare, far corrispondere:
un schema di relazione \rightarrow un solo tipo di entità

Esempio: LG2

Anagrafe(CF, NomeP, Indirizzo, NomeC, NumAb)

Semantica attributi:

- ▶ CF determina NomeP, Indirizzo e NomeC
- ▶ NomeC determina NumAb

Considerazioni:

- ▶ NumAb ripetuto per tuple con lo stesso NomeC
⇒ da mantenere consistenza
- ▶ evitare problema trasformando lo schema in due schemi:
 - Persona(CF, NomeP, Indirizzo, NomeC)
 - Residenza(NomeC, NumAb)

Formalizzazione

Dipendenza funzionale: DF

Definizione

Esprime un legame semantico tra due gruppi di attributi di uno schema R .

- ▶ è una proprietà di R , non di un particolare stato valido r di R
- ▶ non può essere dedotta a partire da uno stato valido r
- ▶ deve essere definita esplicitamente da qualcuno che conosce la semantica degli attributi di R

Forme normali

- ▶ esistono diversi tipi, ciascuno:
 - garantisce l'assenza di determinati difetti in R
 - quindi definisce un determinato livello di qualità di R
- ▶ possibile eseguire una serie di test per certificare che R soddisfa una data forma normale

Normalizzazione

Definizione

La **procedura** che permette di portare uno schema relazionale in una determinata forma normale si dice normalizzazione.

Osservazione

La normalizzazione può essere utilizzata come **tecnica di verifica** dei risultati della progettazione, non costituisce una metodologia di progetto.

Esempio (1/2)

R (Impiegato, Stipendio, Progetto, Bilancio, Funzione)

Considerazioni:

- ▶ Ogni impiegato può partecipare a più progetti, sempre con lo stesso stipendio, e con una sola funzione per progetto.
- ▶ Ogni progetto ha un bilancio indipendentemente da quanti dipendenti ci lavorano

Esempio (2/2)

Possibili anomalie:

► di aggiornamento

- se lo stipendio di un impiegato varia → variano altre tuple (quali?)
- se bilancio di progetto varia → variano altre tuple (quali?)

► di cancellazione

- se un impiegato si licenzia, dobbiamo cancellarlo in diverse ennuple

Causa dei problemi

Abbiamo ripetizione dello stipendio di un impiegato e del bilancio di un progetto.

Errore di progetto

Usare un'unica relazione per rappresentare gruppi di informazione eterogenee.

Dipendenze funzionali

Ma quindi cosa sono le DF??

Siano:

- r relazione su $R(\mathbf{X})$
- due sottoinsiemi non vuoti di $\mathbf{X} \supseteq \mathbf{Y}, \mathbf{Z}$

Definizione

Esiste in r una **dipendenza funzionale** da \mathbf{Y} a \mathbf{Z} se

$$\forall t_1, t_2 \text{ tuple in } r, t_1[\mathbf{Y}] = t_2[\mathbf{Y}] \implies t_1[\mathbf{Z}] = t_2[\mathbf{Z}]$$

$$\mathbf{Y} \rightarrow \mathbf{Z}$$

Nota bene!

$\mathbf{Y} \rightarrow \mathbf{Z}$ non implica $\mathbf{Z} \rightarrow \mathbf{Y}$!

DF complete

Una dipendenza funzionale si dice **completa** se

$\mathbf{Y} \rightarrow \mathbf{Z}$, e $\forall \mathbf{W} \subseteq \mathbf{Y}$, non vale $\mathbf{W} \rightarrow \mathbf{Z}$

- ▶ Se \mathbf{Y} superchiave di $\mathbf{R}(\mathbf{X})$, allora \mathbf{Y} determina ogni altro attributo della relazione $\implies \mathbf{Y} \rightarrow \mathbf{X}$
- ▶ Se \mathbf{Y} chiave, allora $\mathbf{Y} \rightarrow \mathbf{X}$ è una DF completa

Esempio

R (Impiegato, Stipendio, Progetto, Bilancio, Funzione)

Caratterizziamo le dipendenze funzionali:

- ▶ Impiegato \rightarrow Stipendio
- ▶ Progetto \rightarrow Bilancio
- ▶ Impiegato Progetto \rightarrow Funzione

Definizione

Una DF $\mathbf{Y} \rightarrow \mathbf{Z}$ si dice **banale** se $\forall Z_i$ attributo di \mathbf{Z} , vale $\mathbf{Y} \rightarrow Z_i$

- ▶ Impiegato Progetto \rightarrow Progetto (DF banale, sempre soddisfatta.)

Legame fra DF e anomalie

Notare che:

- ▶ Impiegato → Stipendio, ci sono ripetizioni
- ▶ Progetto → Bilancio, ci sono ripetizioni
- ▶ Impiegato Progetto → Funzione, **non** ci sono ripetizioni

Le informazioni legate ad attributi non chiave causano problemi.

Implicazione

Siano F un insieme di dipendenze funzionali su $\mathbf{R}(\mathbf{Z})$ e $\mathbf{X} \rightarrow \mathbf{Y}$

Allora:

Definizione

Se $\forall \mathbf{r}$ istanza di \mathbf{R} che verifica tutte le dipendenze in F , risulta verificata anche $\mathbf{X} \rightarrow \mathbf{Y}$, allora si dice che F implica $\mathbf{X} \rightarrow \mathbf{Y}$.

$$F \implies \mathbf{X} \rightarrow \mathbf{Y}$$

Chiusura

Siano F un insieme di dipendenze funzionali su $\mathbf{R}(\mathbf{Z})$ e $\mathbf{X} \rightarrow \mathbf{Y}$, allora la chiusura di F è l'insieme di tutte le dipendenze funzionali implicate da F :

Definizione

$$F^+ = \{ \mathbf{X} \rightarrow \mathbf{Y} \mid F \implies \mathbf{X} \rightarrow \mathbf{Y} \}$$

Osservazione

Se un'istanza \mathbf{r} dello schema \mathbf{R} soddisfa F , allora soddisfa anche F^+ .

Superchiave

Siano F un insieme di dipendenze funzionali su $\mathbf{R}(\mathbf{Z})$, e $\mathbf{K} \subseteq \mathbf{Z}$. Allora:

Definizione

\mathbf{K} si dice superchiave di R se la dipendenza funzionale $\mathbf{K} \rightarrow \mathbf{Z}$ è logicamente implicata da F , ovvero se $\mathbf{K} \rightarrow \mathbf{Z} \in F^+$.

Ricordiamo che se nessun insieme proprio di \mathbf{K} è superchiave di \mathbf{R} , allora \mathbf{K} si dice chiave di \mathbf{R} . Gli attributi di una chiave non si possono ottenere da nessuna DF, si deve partire da loro per ottenere gli altri.

Costruzione della chiusura

Problema

Trovare tutte le chiavi di una relazione — al peggio esponenziale.
Rinuncia alla "chiusura totale"

Algoritmo:

- attributi SOLO a sx delle DF stanno in tutte le chiavi
- chiamo l'insieme di quei attributi N
- calcola N^+
- aggiungi ad N un attributo alla volta, poi una coppia, etc

Calcolo di F^+

La definizione di **implicazione** prevede un **quantificatore universale**.

- ▶ non utilizzabile in pratica
- ▶ servono **regole di inferenza** per derivare costruttivamente le DF

1974, Armstrong "Dependency structures of data base relationships"
(Spiegati bene anche su Wikipedia)

Regole di inferenza di Armstrong

- 1 **Riflessività**: se $Y \subseteq X$, allora $X \rightarrow Y$
- 2 **Additività/Arricchimento**: se $X \rightarrow Y$, allora $XZ \rightarrow YZ \forall Z$
- 3 **Transitività**: se $X \rightarrow Y$ e $Y \rightarrow Z$, allora $X \rightarrow Z$

Proprietà

Teorema correttezza

Applicandole ad un insieme F di dipendenze funzionali, si ottengono solo dipendenze logicamente implicate da F .

Teorema completezza

Applicandole ad un insieme F di dipendenze funzionali, si ottengono tutte le dipendenze logicamente implicate da F .

Teorema minimalità

Ignorando anche una sola regola, l'insieme di regole che rimangono non è più completo.

Dimostrazione addittività: per assurdo

Supponiamo che $\exists \mathbf{r}$ istanza di \mathbf{R} | vale $\mathbf{X} \rightarrow \mathbf{Y}$, ma non $\mathbf{XZ} \rightarrow \mathbf{YZ}$

Allora $\exists t_1, t_2$ di \mathbf{r} tali che:

$$(1) \quad t_1[\mathbf{X}] = t_2[\mathbf{X}]$$

$$(2) \quad t_1[\mathbf{Y}] = t_2[\mathbf{Y}]$$

$$(3) \quad t_1[\mathbf{XZ}] = t_2[\mathbf{XZ}]$$

$$(4) \quad t_1[\mathbf{YZ}] \neq t_2[\mathbf{YZ}]$$

Da (1) e (3) si deduce: (5) $t_1[\mathbf{Z}] = t_2[\mathbf{Z}]$

Mentre da (2) e (5) abbiamo: (6) $t_1[\mathbf{YZ}] = t_2[\mathbf{YZ}]$,

in contraddizione con (4), assurdo.



Dimostrazione transitività: sempre per assurdo

Fai finta che $\exists \mathbf{r}$ istanza di \mathbf{R} | valgono $\mathbf{X} \rightarrow \mathbf{Y}$ e $\mathbf{Y} \rightarrow \mathbf{Z}$, ma non $\mathbf{X} \rightarrow \mathbf{Z}$

Allora $\exists t_1, t_2$ di \mathbf{r} tali che:

(1) $t_1[\mathbf{X}] = t_2[\mathbf{X}]$

(2) $t_1[\mathbf{Y}] = t_2[\mathbf{Y}]$

(3) $t_1[\mathbf{Z}] = t_2[\mathbf{Z}]$, ma anche

(4) $t_1[\mathbf{Z}] \neq t_2[\mathbf{Z}]$

in contraddizione con (3), assurdo.



Dimostrazione riflessività: fatta io, aperta a critiche

Supponiamo che $\exists \mathbf{Y} \subseteq \mathbf{X} \mid \text{non vale } \mathbf{X} \rightarrow \mathbf{Y}$.

Comunque deve valere $\mathbf{X} \rightarrow \mathbf{X}$, DF banale. Allora $\exists t_1, t_2$ di \mathbf{r} tali che:

(1) $t_1[\mathbf{X}] = t_2[\mathbf{X}]$.

Ma per ipotesi $\mathbf{Y} \subseteq \mathbf{X}$, quindi per forza di cose vale:

(2) $t_1[\mathbf{X}] = t_2[\mathbf{Y}]$,

in contraddizione con l'ipotesi che non vale $\mathbf{X} \rightarrow \mathbf{Y}$, assurdo.



Algoritmo per trovare F^+

```
F+ = F
ripeti
  per ogni DF f in F+
    aggiungi riflessivita(f) ad F+
    aggiungi arricchimento(f) ad F+
  fine
  per ogni DF f1, f2 in F+
    // f1 = X→Y, f2 = Z→W
    se Y == Z
      aggiungi transitivita(f1, f2) ad F+
  fine
fine
```

- complessità esponenziale: il per ogni nasconde il problema di trovare tutti i sottoinsiemi (backtracking), che va come 2^n .

Esempio

Sia lo schema **R** (A, B, C, G, H, I), con le dipendenze funzionali
 $F = \{ A \rightarrow B, A \rightarrow C, CG \rightarrow H, CG \rightarrow I, B \rightarrow H \}$ Alcuni membri di F^+
sono:

- $A \rightarrow H$, ottenuto da Transitività($A \rightarrow B, B \rightarrow H$);
- $AG \rightarrow I$, ottenuto da Arricchimento($A \rightarrow C, G$) = $AG \rightarrow CG$
e Transitività($AG \rightarrow CG, CG \rightarrow I$);
- $CG \rightarrow HI$, ottenuto da Arricchimento($CG \rightarrow I, CG$) = $CG \rightarrow CGI$
e Arricchimento($CG \rightarrow H, I$) = $CGI \rightarrow HI$
e Transitività($CG \rightarrow CGI, CGI \rightarrow HI$);

Regole derivate di Armstrong

- 1 **Unione:** $\{X \rightarrow Y, X \rightarrow Z\} \implies X \rightarrow YZ$
- 2 **Pseudotransitività:** $\{X \rightarrow Y, WY \rightarrow Z\} \implies WX \rightarrow Z$
- 3 **Decomposizione:** se $Z \subseteq Y$, $\{X \rightarrow Y\} \implies X \rightarrow Z$

Dimostrazione della Regola dell'Unione

Le regole derivate si dimostrano a partire dalle regole di braccio forte. Infatti, per ipotesi valgono:

(1) $\mathbf{X} \rightarrow \mathbf{Y}$, che per addittività diventa (3) $\mathbf{XZ} \rightarrow \mathbf{YZ}$

(2) $\mathbf{Y} \rightarrow \mathbf{Z}$, che per addittività diventa (4) $\mathbf{XX} \rightarrow \mathbf{XZ} = \mathbf{X} \rightarrow \mathbf{XZ}$

Applicando la transitività a (4) e (3), segue la tesi.



Esempio (LG2)

R (Impiegato, Stipendio, Progetto, Bilancio, Funzione)

- ▶ Ogni impiegato può partecipare a più progetti, sempre con lo stesso stipendio, e con una sola funzione per progetto.
- ▶ Ogni progetto ha un bilancio indipendentemente da quanti dipendenti ci lavorano

-
- (a) Impiegato \rightarrow Stipendio + arricchimento
= Impiegato Progetto \rightarrow Stipendio Progetto (d)
- (b) Progetto \rightarrow Bilancio + arricchimento
= Progetto Impiegato \rightarrow Stipendio Impiegato (e)
- (c) Impiegato Progetto \rightarrow Funzione

Applicandocenter la Regola dell'Unione alle $\{(c), (d), (e)\}$ si trova:

Impiegato Progetto \rightarrow Stipendio Progetto Impiegato Funzione,

e quindi Impiegato, Progetto è chiave.

Equivalenza fra insiemi di DF

Il problema dell'equivalenza

Dati F , G insiemi di DF, torna utile sapere se sono **equivalenti**. Ovvero:

Definizione

F , G si dicono equivalenti se $F^+ = G^+$, quindi

- $\forall \mathbf{X} \rightarrow \mathbf{Y} \in F$ risulta $\mathbf{X} \rightarrow \mathbf{Y} \in G^+$, e viceversa
- $\forall \mathbf{X} \rightarrow \mathbf{Y} \in G$ risulta $\mathbf{X} \rightarrow \mathbf{Y} \in F$.

- In pratica, per verificare l'equivalenza, prendi tutte le DF in F , e prova a dimostrarle usando le DF in G , e viceversa. Se tutto va bene, congratulazioni, F equivalente a G !

Chiusura transitiva di un insieme di attributi

Il calcolo di F^+ è molto costoso, ma tutte le DF banali manco servono!
Spesso ci interessa se F^+ contiene qualche specifica dipendenza. Che fare?

Definizione

La chiusura di un insieme di attributi A è l'insieme di tutti gli attributi che dipendono da A , relativamente a un dato F .

Boom!

Teorema

$$\mathbf{X} \rightarrow \mathbf{Y} \in F^+ \iff \mathbf{Y} \subseteq \mathbf{X}^+$$

Quindi è sufficiente fare la chiusura di \mathbf{X} per capire se la dipendenza vale

Algoritmo per trovare X^+

```
InsiemeAttributi  $X^+ = X$ 
InsiemeAttributi oldX+ = NULL
InsiemeDF F

finche ( $X^+ \neq \text{oldX}^+$ )
  ripeti
    oldX+ =  $X^+$ 
    per ogni DF  $V_i \rightarrow W_i$  in F
      se ( $V_i$  incluso in  $X^+$ ) e ( $W_i$  non incluso in  $X^+$ )
         $X^+ = X^+$  reunito con  $W_i$ 
      fine
    fine
  fine
fine
```

Esempio: calcola se A è superchiave

Sia $F = \{A \rightarrow B, BC \rightarrow D, B \rightarrow E, E \rightarrow C\}$. Calcoliamo A^+

- $A^+ = A$
- $A^+ = AB$ poiché $A \rightarrow B$ e $A \subseteq A^+$
- $A^+ = ABE$ poiché $B \rightarrow E$ e $B \subseteq A^+$
- $A^+ = ABEC$ poiché $E \rightarrow C$ e $E \subseteq A^+$
- $A^+ = ABCED$ poiché $BC \rightarrow D$ e $BC \subseteq A^+$

Quindi da A dipendono tutti gli altri attributi, ovvero A è superchiave (e anche chiave)!

Esempio: calcola se F, G equivalenti

Siano: $F = \{A \rightarrow C, AC \rightarrow D, E \rightarrow AD, E \rightarrow H\}$,
 $G = \{A \rightarrow CD, E \rightarrow AH\}$

Invece di verificare se $\forall \mathbf{X} \rightarrow \mathbf{Y} \in F$ anche in G^+ , vedo se $\mathbf{Y} \subseteq (\mathbf{X})^{+G}$
(notazione fancy per chiusura di \mathbf{X} rispetto a G) e viceversa.

- per $A \rightarrow C$ si ha $(A)^{+G} = ACD$, $C \subseteq (A)^{+G}$ ✓
- per $AC \rightarrow D$ si ha $(AC)^{+G} = ACD$, $D \subseteq (AC)^{+G}$ ✓
- per $E \rightarrow AD$ si ha $(E)^{+G} = EACDH$, $AD \subseteq (E)^{+G}$ ✓
- per $E \rightarrow H$ si ha $H \subseteq (E)^{+G}$ ✓

The viceversa is left as an exercise for the reader.

QUINDI...

F, G equivalenti sse...

▶ $\forall \mathbf{X} \rightarrow \mathbf{Y} \in F, \mathbf{Y} \in (\mathbf{X})^{+G}$

▶ $\forall \mathbf{X} \rightarrow \mathbf{Y} \in F, \mathbf{Y} \in (\mathbf{X})^{+G}$

Conclusioni sulla chiusura di attributi

Sia $R(Z)$ con le sue dipendenze in F .

Allora, la chiusura di $X \subseteq Z$ utile per verificare se:

- ▶ una dipendenza funzionale è logicamente implicata da F (vedi Teorema)
- ▶ un insieme di attributi è superchiave o chiave
 - X è superchiave di R sse $X \rightarrow Z \in F^+$, ovvero sse $Z \subseteq (X)^+$
 - X è chiave di R sse $X \rightarrow Z \in F^+$ e $\nexists Y \subset X$ tale che $Z \subseteq (Y)^+$

Ridondanza di un insieme di DF

Il Problema part.2

Si vuole usare il concetto di equivalenza tra insiemi di DF per partire da una F più semplice possibile.

- ▶ a **destra**: $\{A \rightarrow BC\}$ equivalente a $\{A \rightarrow B, A \rightarrow C\}$ "DF semplici"
- ▶ a **sinistra**: $\{A \rightarrow B, AB \rightarrow C\}$ equivalente a $\{A \rightarrow B, A \rightarrow C\}$
"senza **attributi estranei**"
- possono esserci DF ridondanti, aka ottenibili da altre DF:
 $A \rightarrow C$ ridondante $\{A \rightarrow B, B \rightarrow C, A \rightarrow C\}$

La riduzione della complessità può riguardare il numero di attributi che si usano in una dipendenza, o il numero di dipendenze nell'insieme.

DF semplici

Possiamo portare un insieme di DF F in forma “standard”, quella in cui sulla destra c'è un singolo attributo.

Per esempio:

Se $F = \{AB \rightarrow CD, AC \rightarrow DE\}$ lo si scompone in:
 $\{AB \rightarrow C, AB \rightarrow D, AC \rightarrow D, AC \rightarrow E\}$

Attributi estranei

Definizione

Gli attributi a sinistra di una DF che sono "inutili" (cioè la DF vale anche senza di loro) rispetto ad un dato F si dicono estranei.

Sia $F = \{AB \rightarrow C, A \rightarrow B\}$, e calcoliamo A^+ , B^+ :

- $A^+ = A$
- $A^+ = AB$ poiché $A \rightarrow B$ e $A \subseteq A^+$
- $A^+ = ABC$ poiché $AB \rightarrow C$ e $AB \subseteq A^+$

Quindi C dipende solo da A . Per quello possiamo riscrivere F come:

$$F := \{A \rightarrow C, A \rightarrow B\}$$

In generale, in una DF del tipo $A\mathbf{X} \rightarrow B$, A è estraneo se $B \subseteq \mathbf{X}^+$.

DF ridondanti

Dopo aver portato le DF in F in "forma standard" (o "semplici", con un solo attributo a destra) ed eliminato gli attributi estranei, vogliamo assicurarci che tutte le ridondanze in F sono necessarie.

- Come si fa a dire che $\mathbf{X} \rightarrow A$ è ridondante?

Procedimento:

- elimina $\mathbf{X} \rightarrow A$ da F
- calcola \mathbf{X}^+ rispetto al nuovo F
- verifica se $A \in \mathbf{X}^+$

Se così, si riesce a trovare A anche senza la DF $\mathbf{X} \rightarrow A$, e quindi quest'ultima è ridondante.

Copertura minimale

Let's pretend for a second this doesn't sound like politics.

Definizione

Un insieme F di DF si dice **copertura minimale** se:

- I. è in forma standard (o semplice) — a destra delle DF
- II. non presenta attributi estranei — a sinistra delle DF
- III. non vi sono ridondanze

In pratica, è un insieme di DF equivalente a F ma di complessità minore.

In generale una copertura minimale **non è unica**.

Algoritmo per trovare la Copertura Minimale

InsiemeDF $M = F$

per ogni ogni DF $X \rightarrow \{A_1, \dots, A_n\}$ in M
 sostituisci con $X \rightarrow A_1, \dots, A \rightarrow A_n$

per ogni DF $X \rightarrow A$ in M
 se A incluso in $(X - \{B\})^+$
 sostituisci con $(X - \{B\}) \rightarrow A$

per ogni DF $X \rightarrow A$ in M
 se A incluso in X^+ anche rispetto a $M - \{X \rightarrow A\}$
 rimuovi $X \rightarrow A$ da M

Esempio

Sia $F = \{AB \rightarrow C, B \rightarrow A, C \rightarrow A,\}$

- ▶ $A^+ = A, B^+ = ABC$ quindi A estraneo in $AB \rightarrow C$ quindi:
 $F' := \{B \rightarrow C, B \rightarrow A, C \rightarrow A,\}$
- ▶ eliminando $B \rightarrow C$ non possiamo arrivare a C , ma eliminando $B \rightarrow A$ rimane $B^+ = ABC$ per cui:
 $F'' := \{B \rightarrow C, C \rightarrow A,\}$

Nota Bene

Tentare ad eliminare le DF ridondanti prima di risolvere gli attributi estranei risulta in inevitabile fallimento e dolore. Il dottore avvisa contro.

Esempio

Sia $F = \{A \rightarrow BC, B \rightarrow C, A \rightarrow B, AB \rightarrow C\}$

- ▶ Portando le DF in forma standard:

$$F := \{A \rightarrow B, A \rightarrow C, B \rightarrow C, A \rightarrow B, AB \rightarrow C\}$$

- ▶ Vogliamo poi eliminare possibili attributi estranei. L'unica DF che potrebbe presentarli è l'ultima, quindi si calcolano $A^+ = ABC$, $B^+ = BC$. Quindi sia da A che da B si può raggiungere C . Scegliamo di togliere A ,

$$F = \{A \rightarrow B, A \rightarrow C, B \rightarrow C, A \rightarrow B, B \rightarrow C\}$$

- ▶ $B \rightarrow C$ e $A \rightarrow B$ appaiono due volte e quindi le possono togliere. Inoltre, calcolando A^+ rispetto ad $F - A \rightarrow C$, abbiamo sempre che $A^+ = ABC$, e dunque la $A \rightarrow C$ ridondante.

$$F = \{A \rightarrow B, B \rightarrow C\}$$