BASI DI DATI

(Modulo 1)

Transazione: Sequenza di singole operazioni. Se non viene completata (committed), viene completamente annullata (rollback)

Il Modello Relazionale

Dati n domini (non necessariamente distinti) $D_1 \dots D_n$

Una relazione r è un sottoinsieme del loro prodotto cartesiano

Di solito viene rappresentata con una tabella

Attributi: nomi delle colonne (sempre distinti)

Tuple: righe delle colonne

Istanza di una relazione: tutte le righe della tabella

Schema di base di dati relazionale: insieme di schemi di relazione $\{R_1 \dots R_n\}$

Algebra Relazionale

Insieme di operatori che possono essere applicati a una (operatori unari) o due (operatori binari) istanze di relazioni e restituiscono un'altra istanza di relazion

Proiezione: estrapolare solo i valori di determinati attributi (π) (ritorna una relazione)

Attenzione! Nell'istanza di una relazione non possono esistere elementi ripetuti (questo accade solo nell'algebra normale)

Selezione: seleziona solo le righe che soddisfano una certa condizione (σ)

29/09/16

Selezione: ordine degli operatori: $(-, +, *)(\neg, \vee, \wedge)$

operatore confronto: $A \Theta B$ $(\Theta \in \{<, =, >, \leq, \geq\})$, con A e B attributi con lo stesso dominio

Unione: Gli operandi devono avere gli stessi attributi, e gli attributi corrispondenti devono essere definiti sullo stesso dominio ($\cup$)

Differenza: si applica solo a operandi union compatibili

Costruisce una relazione contenente tutte le tuple contenute nel primo operando che non appartengono al secondo operando ($-$)

Intersezione: si applica a operandi union compatibili ($\cap$)

Per unione, differenza e intersezione per convenzione i nomi degli attributi sono quelli del primo operando

Prodotto cartesiano: ritorna una relazione che ha come attributi la somma degli attributi dei due operandi (cambiando i nomi se necessario), e come tupla ha la concatenazione di ogni tupla del primo con ogni tupla del secondo ($\times$)

Ridenominare: Modifica il nome degli attributi di una relazione (ρ)

$$\rho_{A_1, A_2, A_3 \dots}(r)$$

Join Naturale: Seleziona le tuple del prodotto cartesiano che soddisfano la condizione.

$R_1 A_1 = R_2 A_1 \wedge R_1 A_2 = R_2 A_2 \wedge \dots \wedge R_1 A_k = R_2 A_k$

Il simbolo è ($\bowtie$)

Ridondanza: quando in una relazione ripeto gli stessi dati tante volte

Anomalie di aggiornamento: se un dato deve essere modificato, non dovrebbe essere aggiornato tante volte

Anomalie di inserimento: Se un'entrata non ha tutti gli attribbuti non può essere aggiunta

Anomalie di cancellazione: Se cancello un'entrata, potrei accidentalmente eliminare dati utili

Uno schema se non presenta ridondanze e anomalie è uno schema "buono"
Di solito la soluzione migliore è dividere i concetti singoli in relazioni distinte

Schema di Relazione. un insieme di attribbuti $R = \{A_1, A_2 \dots\}$

Tupla: funzione che associa a ogni attribbuto A_m in R un valore $t[A_m]$ nel dominio di A_m

Dipendenza funzionale: un'istanza r di R soddisfa una dipendenza funzionale $X \rightarrow Y$ se:

(dove X e Y sono attribbuti)

$$\forall t_1, t_2 \in r \left(t_1[X] = t_2[X] \Rightarrow t_1[Y] = t_2[Y] \right)$$

Chiave: Dati uno schema di relazione R e un insieme F di dipendenze funzionale un sottoinsieme K di uno schema di relazione R è una chiave di R se:

1. $K \rightarrow R \in F^+$
2. Non esiste un sottoinsieme proprio K' di K tale che $K' \rightarrow R \in F$

Chiave primaria: una chiave (una sola) che non può mai avere valore nullo

Dati R, F, uno schema è in terza forma normale se $\forall X \to A \in F^+$ con $A \notin X$ si ha X contiene una chiave oppure l'attributo a destra (A) contiene una chiave

Tutte le dipendenze non banali che devono valere su ogni istanza legale e hanno a destra un singolo attributo sono del tipo $X \to A$ dove X contiene una chiave

Dato uno schema R e un insieme F di dipendenze su R, se tutte le dipendenze funzionali rispettano la precedente proprietà, allora lo schema è in 3ª forma normale

Esiste un algoritmo polinomiale che dati uno schema di relazione R e un insieme di relazioni funzionali F, lo rende in terza forma normale senza perdere informazioni

Lemma Dati $R, F, X \subseteq R$ $\quad Y \subseteq X^+ \iff X \to Y \in F^A$

Una conseguenza di questo lemma è che $X \subseteq X^+$ (poiché $X \to X \in F^A$ per l'assioma della riflessività)

per dimostrare che $F^+ \subseteq F^A$, faccio vedere che $\forall X \to Y$ vale che $X \to Y \in F^+ \Rightarrow X \to F^A$,
e lo nego per assurdo, dicendo che $\exists X \to Y \in F^+ \wedge X \to Y \notin F^A$, ed è una istanza legale
Ma per il lemma precedente, $Y \subseteq X^+ \Rightarrow X \to Y \in F^A$, che da una contraddizione.
Per dimostrare che l'istanza usata è legale, lo facciamo per assurdo dicendo che
$\exists V \to W \in F$ che non è soddisfatta da r, cioè vuol dire che $\exists t_1, t_2 \in r \left(t_1[V] = t_2[V] \wedge t_1[W] \neq t_2[W]\right)$
e con un po' di logica si vede che $\neg \forall t_1, t_2 \in r \left(t_1[V] = t_2[V] \Rightarrow t_1[W] \neq t_2[W]\right)$ Dalla prima metà si deduce che $V \subseteq X^+$, e dalla seconda metà si deduce che $W \cap (R - X^+) \neq \emptyset$.
Allora $X \to V \in F^A \wedge V \to W \in F^A$, e per l'assioma della transitività, $X \to W \in F^A$, e applicando il lemma, $W \subseteq X^+$, che è in contraddizione con $W \cap (R - X^+) \neq \emptyset$, e quindi è assurdo.
Manca tutta la dimostrazione di $F^A \subseteq F^+$

F^A sono le df date dagli assiomi di Armstrong
F^+ sono tutte, anche quelle che non hai deciso.

$X_F^+ = \{A \mid X \to A \in F^A\}$, tutti gli attribbuti che dipendono da X ricavabili dagli assiomi di Armstrong

$X \to Y \in F^+$? Devo vedere se $Y \subseteq X^+_F$? (per il lemma di prima)

$\uparrow$ chiusura di X rispetto a F

perchè $F = \{A \to B_1, A \to B_2, A \to B_m\}$; $F^+ = \{A \to B_1 \dots B_m\} \cup \underbrace{\{A \to X \mid X \subseteq B_1 \dots B_m\}}_{2^m - 1}\dots$

$2^m - 1 \to$ quindi il numero di dipendenze funzionali di F^+ sono almeno 2^m

Ma m è la cardinalità di F

quindi

$$|F^+| \geq 2^{|F|} \qquad m = |F|$$

Input $R, F, X \subseteq R$
Output X^+ nella variabile Z

begin $Z := \emptyset$; $S := \{A \mid V \to W \in F \wedge V \subseteq Z \wedge A \in W\}$
 while $S \not\subseteq Z$ do
  begin $Z := Z \cup S$;
   $S := S \cup \{A \mid V \to W \in F \wedge \underline{V \subseteq Z} \wedge A \in W\}$
  end
end

prendo la parte destra

$Z^{(0)}$ = il valore di Z prima di entrare nel ciclo while

$Z^{(i)}$ il valore di Z dopo che è stato eseguito i volte il corpo del while

$Z^{(0)} = X$
$Z^{(i)} \subseteq Z^{(i+1)}$

Applicazione algoritmo

$R = ABCDE \qquad F = \{AB \to C, B \to D, D \to E\}$

$(AB)^+$

$Z^{(0)} = AB$
$S^{(0)} = CD$
$Z^{(1)} = Z^{(0)} \cup S^{(0)} = ABCD$
$S^{(1)} = CD \cup CDE = CDE$
$Z^{(2)} = ABCDE$
$S^{(2)} = CDE$

e con questo algoritmo trovi la chiusura di $(AB)^+_F = ABCDE$

Proviamo con la chiusura di B; $Z^{(0)} = B$, $S^{(0)} = D$; $Z^{(1)} = BD$, $S^{(1)} = DE$; $Z^{(2)} = BDE$, $S^{(2)} = DE$; $B^+ = BDE$

Proviamo la chiusura di A: $A^+ = A$ fine

Def Dato R, F, $K \subseteq R$ è chiave di R se

1) $K \to R \in F^+$
2) $\neg \exists K' \subsetneq K \mid K' \to R \in F^+$

quindi nell'esempio di prima, l'insieme AB è chiave poichè $AB \to R \in F^+$ (per il lemma di prima)

Invece $A \to R \notin F^+$ poichè $R \not\subseteq A^+$

Teor L'algoritmo calcola correttamente X^+_F

DIMOSTRAZIONE

Bisogna far vedere che $Z^{(f)} = X^+_F$, cioè che $Z^{(f)} \subseteq X^+_F \wedge X^+_F \subseteq Z^{(f)}$

1) $\forall i\ Z^{(i)} \subseteq X^+_F$, e lo dimostriamo per induzione:

Base: $(i=0)$ $Z^{(0)} = X \subseteq X^+_F$

Induzione: $(i>0)$ Ipotesi induttiva $Z^{(i-1)} \subseteq X^+_F$

Sia $A \in (Z^{(i)} - Z^{(i-1)})$, cioè un attributo qualunque che è stato aggiunto al passo i-esimo quindi $A \in S^{(i-1)}$, e ciò vuol dire che $\exists$ una dipendenza funzionale $V \to W \in F$ tale che $A \in W$ e $V \subseteq Z^{(i-1)}$

Ma siccome per ip. induttiva $Z^{(i-1)} \subseteq X^+_F$, e $V \subseteq Z^{(i-1)}$, $V \subseteq X^+_F$ e quindi $X \to V \in F^A$ (per il solito lemma)

Siccome ho che $V \to W \in F^A$ e $X \to V \in F^A$, per la transitività, $X \to W \in F^A$, e allora $W \subseteq X^+_F$ (lemma), e allora $A \in X^+_F$, poiché $A \in W$

2) $X^F_+ \subseteq Z^{(f)}$, e lo dimostriamo così: $X^+_F = \{A \mid X \to A \in F^A\}$

Sia $A \in X^F_+$; $X \to A \in F^A = F^+$,

quindi $X \to A$ è soddisfatta da ogni istanza legale. Siccome l'istanza è legale, soddisfa $X \to A$, e allora $A \subseteq Z^{(f)}$, poiché $X \subseteq Z^{(0)}$. Però dobbiamo dimostrare che l'istanza r è legale. Lo facciamo per assurdo. Se r non è legale vuol dire che $\exists\ V \to W \in F$ che non è soddisfatta da r, $\Rightarrow \exists t_1, t_2 \in r \mid t_1[V] = t_2[V] \wedge t_1[W] \neq t_2[W]$

Il fatto che $t_1[V] = t_2[V]$ vuol dire che $V \subseteq Z^{(f)}$. Il fatto che $t_2[W] \neq t_1[W]$ ci dice che $W \cap (R - Z^{(f)}) \neq \emptyset$. Cioè vuol dire che almeno un attributo in W non è contenuto in $Z^{(f)}$. Ma ciò significherebbe che $S^{(f)} \not\subseteq Z^{(f)}$, e quindi l'algoritmo non è ancora finito.

1) $\pi_{COGNOME}\left(\left(\sigma_{ANNO=2}(CORSO) \bowtie ESAME\right) \bowtie \left(\sigma_{ANNO.I = 2013}\ STUD\right)\right)$

$\pi_{COGNOME}\left(\sigma_{ANNO.I = 2015}\left(STUD - \pi_{MATR, CO, ANN.I}\left(\left(\sigma_{ANNO=2}(CORSO) \bowtie ESAME\right) \bowtie STUD\right)\right)\right)$

Numero di matricola degli studenti che non hanno sostenuto almeno un esame del primo anno

$\pi_{Matr}\left(\sigma_{ANNO=2}(STUD)\right) - \pi_{Matr}\left(\pi_{Matr}(STUD) \times \pi_{C\#}\left(\sigma_{ANNO\_ISCR=1}(CORSO)\right) - ESAME\right)$

$IMP(I\#, Nome, Sal, Mgr)$

$IMP \bowtie \left(\rho_{A, NOME_2, SAL_2, I\#}\right)$

qual è il salario più alto?

$\pi_{SAL}(IMP) - \sigma_{SAL < S}\left(\pi_{SAL}(IMP) \times \pi_S\left(\rho_{INSM}(IMP)\right)\right)$

Decomposizione Dato R, una decomposizione di R è una famiglia

$$\rho = \{R_1, R_2, \dots R_k\} \text{ di sottoinsiemi di } R \text{ tali che } \bigcup_{i=1}^{k} R_i = R$$

Non è una partizione poichè i sottoinsiemi possono non essere a due a due disgiunti

Def Se ho due insiemi di dipendenze funzionali F, G, allora $F \equiv G$ se $F^+ = G^+$
(sono equivalenti)

Dati R, F, una decomposizione ρ di R preserva F se

$$F \equiv G, \text{ dove } G = \bigcup_{i=1}^{k} \{X \to Y \in F^+ \mid XY \subseteq R_i\}$$

Dati F, G $\quad F^+ \subseteq G^+ \Longleftrightarrow F \subseteq G^+$

DIMOSTRAZIONE

$\Rightarrow$ 1) è banale poichè $F \subseteq F^+$

$\Leftarrow$ 2) $F \subseteq G^+$ vuol dire che ogni dipendenza in F può essere ottenuta da G applicando gli assiomi di Armstrong, e ogni dipendenza in F^+ può essere ottenuta da F applicando gli assiomi; e poichè abbiamo dimostrato che $F^+ = F^A$, allora $F^+ \subseteq G^+$

Insomma per dimostrare che $F \equiv G$ basta far vedere che $F \subseteq G^+ \wedge G \subseteq F^+$

Problema Dati $R, F, \rho = \{R_1 \dots R_k\}$, ρ preserva F?

SI, se e solo se $F \subseteq G^+$ dove $G = \bigcup_{i}^{k} \{x \to y \in F^+ \mid yx \subseteq R_i\}$

Algoritmo:

INPUT $R, F, \rho = \{R_1 \dots R_k\}$

OUTPUT (boolean) successo

begin successo := true;

for every $X \to Y \in F$ do

if $Y \not\subseteq X^+_G$ // verifica se $X \to Y \in G^+$, per il solito lemma

then successo := false

end

Devo trovare un modo per calcolare la chiusura di X rispetto a G

Algoritmo per trovare la chiusura di X rispetto a G

Input $R, F, X, \rho = \{R_1 \dots R_K\}$

Output X^+_G nella variabile Z;

begin $Z := X$, $S := \emptyset$

  for $j := 1$ to K do

    $S := S \cup (Z \cap R_j)^+_F \cap R_j$;

  while $S \not\subseteq Z$ do

    begin $Z := Z \cup S$

    for $j := 1$ to K do

      $S := S \cup (Z \cap R_j)^+_F \cap R_j$

    end

  end

Esempio: $R = ABC$, $F = \{A \to B, B \to C, C \to A\}$

$\rho = \{AB, BC\}$ ρ preserva F?

$A \to B \subseteq G^+$ sì poiché AB è contenuto in uno schema della decomposizione e poiché $A \to B \in F \subseteq F^+$

$B \to C \subseteq G^+$ sì.

$C \to A \in G^+$? se e solo se $A \in C^+_G$

Ora calcoliamo C^+_G con il secondo algoritmo

$(K=2)$

$Z = C$

$S = \emptyset$

$S = \emptyset \cup (C \cap AB)^+_F \cap AB = \emptyset$

$S = \emptyset \cup (C \cap BC)^+_F \cap BC = \emptyset \cup CAB \cap BC = BC$

$Z = C \cup BC = BC$

$S = BC \cup (BC \cap AB)^+_F \cap AB = BC \cup BCA \cap AB = BC \cup AB = ABC$

$S = ABC \cup (BC \cap BC)^+_F \cap BC = ABC \cup ABC \cap BC = ABC$

$Z = ABC$, finito.

1) $\text{CLIENTE}_{\sigma_{C=ROMA}} \bowtie (\text{ORDINE} \bowtie (\sigma_{\text{ARTICOLO}=\text{FRIGO}}\ \text{ARTICOLO}))$

2) $\text{CLIENTE}_{\sigma_{C=ROMA}} \bowtie (\text{ORDINE} \bowtie (\sigma_{\text{ARTICOLO}\neq\text{FRIGO}}\ \text{ARTICOLO}))$

3) $\underset{\sigma_{C=ROMA}}{}$ $\text{CLIENTE} - \text{CLIENTE}_{\sigma_{C=ROMA}} \bowtie (\text{ORDINE} \bowtie (\sigma_{\text{ARTICOLO}=\text{FRIGO}}\ \text{ARTICOLO}))$

4) $\text{CLIENTE}_{\sigma_{C=ROMA}} \bowtie (\text{ORDINE} \bowtie (\sigma_{\text{ARTICOLO}=\text{FRIGO}}\ \text{ARTICOLO})) -$

$\text{CLIENTE}_{\sigma_{C=ROMA}} \bowtie (\text{ORDINE} \bowtie (\sigma_{\text{ARTICOLO}\neq\text{FRIGO}}\ \text{ARTICOLO}))$

Prop $Z^{(f)} \subseteq X^+_G$ (la Z è quella dell'algoritmo)

DIMOSTRAZIONE

$\forall i\ Z^{(i)} \subseteq X^+_G \Rightarrow$ Per induzione su i:

Base: $i=0$ $Z^{(0)} = X \subseteq X^+_G$

Induzione: $i>0$ Per hp induttiva, $Z^{(i-1)} \subseteq G$

Sia $A \in Z^{(i)} - Z^{(i-1)}$. Deve esistere $j \mid A \in (Z^{(i-1)} \cap R_j)^+_F \cap R_j$, da cui $A \in (Z^{(i-1)} \cap R_j)^+_F$ e $A \in R_j$

Siccome $Z^{(i)} = Z^{(i-1)} \cup S^{(i-1)}$, allora $A \in S^{(i-1)}$. Da (1) segue che $Z^{(i-1)} \cap R_j \to A \in F^+$. Poiché $Z^{(i-1)} \cap R_j \subseteq R_j$ e $A \in R_j$

si ha che $Z^{(i-1)} \cap R_j \to A \in G$. $X \to Z^{(i-1)} \in G^A$

decomp. $\to \vdash X \to Z^{(i-1)} \cap R_j \in G^A$ per il lemma $Y \subseteq X^+_F \Leftrightarrow X \to Y \in F^A$ $(=F^+)$

Per transitività, $X \to A \in G^A$, e quindi $A \in X^+_G$

MUSEO (IdM, Nome, Indirizzo, Città)

OPERA (IdO, Titolo, IdM, IdA)

ARTISTA (IdA, Nome, Cognome, Soprannome, Periodo)

1) Dati dei musei in cui non sono conservate opere di Caravaggio

2) Dati dei musei in cui sono conservate solo opere del rinascimento

1) $MUSEO - \left(MUSEO \bowtie \pi_{IdM}\left(\sigma_{SOPRANNOME = CARAVAGGIO}(ARTISTA) \bowtie OPERA \bowtie MUSEO\right)\right)$

2) $MUSEO - \left(MUSEO \bowtie \pi_{IdM}\left(\sigma_{PERIODO \neq RINASCIMENTO}(ARTISTA) \bowtie OPERA \bowtie MUSEO\right)\right)$

Prop Dati: R, F, $\rho = \{R_1 \dots R_k\}$ ρ ha un join senza perdita se

l'istanza legale r di R si ha che $r = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \dots \bowtie \pi_{R_k}(r)$

Esempio per join con perdita $R = ABC$ $F = \{A \to B, C \to B\}$ $\rho = \{AB, BC\}$

r legale

$$\begin{array}{ccc} A & B & C \\ a_1 & b_1 & c_1 \\ a_2 & b_1 & c_2 \end{array} \Rightarrow \pi_{AB} \begin{array}{cc} A & B \\ a_1 & b_1 \\ a_2 & b_1 \end{array} \quad \pi_{BC} \begin{array}{cc} b_1 & c_1 \\ b_1 & c_2 \end{array} \Rightarrow \pi_{AB} \bowtie \pi_{BC} = \begin{array}{ccc} a_1 & b_1 & c_1 \\ a_1 & b_1 & c_2 \\ a_2 & b_1 & c_1 \\ a_2 & b_1 & c_2 \end{array}$$

E' tempo di un algoritmo!

Input $R, F, \rho = \{R_1 \dots R_N\}$

Output decide se ρ ha un join senza perdita

Costruisce una tabella con $|R|$ colonne e $|\rho|$ righe

All'incrocio tra la riga i e la colonna j mette a_j se $A_j \in R_i$, b_{ij} altrimenti

```
repeat
    for every X->Y ∈ F do
        begin if in r ci sono due tuple tali che t1[X] = t2[X] ∧ t1[Y] ≠ t2[Y]
            then
                for every A ∈ Y do
                    if t1[Aj] = aj then
                        t2[Aj] := t1[Aj]
                    else t1[Aj] := t2[Aj]
until la tabella r non cambia
if r ha una riga con tutti 'a' then
    ρ ha un join senza perdita
else ρ non ha un join senza perdita
```

DIMOSTRAZIONE – il fatto che r^f è un'istanza legale è una dimostrazione troppo lunga

Dobbiamo dimostrare che $\{R_1 \dots R_k\} = \rho$ ha un j.s.p. $\Rightarrow r^f$ ha una riga con tutti 'a'

Supponiamo per assurdo che ρ ha un j.s.p. e r^f non ha una riga con tutti 'a'

$r^f \neq \pi_{R_1}(r^f) \bowtie \dots \bowtie \pi_{R_k}(r^f)$, ma è una contraddizione.

G è una copertura minimale di F se $F \equiv G$ e

1) La parte destra di ogni dipendenza in G è un singleton $(X \to A)$
2) Per ogni dipendenza in G, e per ogni attributo B di X, $(G - \{X \to A\}) \cup \{(X-B) \to A\} \not\equiv G$
3) Per ogni dipendenza in G, $(G - \{X \to A\}) \not\equiv G$