Lecture 18 & 19

ENTAILMENT OF CONSTRAINTS

IMPLICATION OF CONSTRAINTS

→ A set Σ of constraints implies/entails a constraint ϕ if every instance that satisfies Σ also satisfies ϕ

↳ $\boxed{\Sigma \models \phi}$

→ Given Σ and ϕ, does Σ imply ϕ? [Implication problem]

→ is important because:

1) We never get the list of all constraints tht. hold in a database
2) The given constraints may look fine, but imply some bad ones (that you don't want to hold on your dbase)
3) The given constraints may look bad, but imply good ones

AXIOMATIZATION OF CONSTRAINTS

→ is a set of rules/axioms to derive constraints

→ is called sound if every derived constraint is implied

is called complete if every implied constraint can be derived

∴ Sound & complete axiomatization gives a procedure $\vdash$ such that

$$\Sigma \models \phi \text{ if and only if } \Sigma \vdash \phi$$

NOTATION

- Attributes are denoted by A, B, C, ...
  - ↳ AB denotes the set $\{A, B\}$
- Sets of attributes are denoted by X, Y, Z, ...
  - ↳ XY denotes their union $X \cup Y$

→ XA denotes $X \cup \{A\}$

ARMSTRONG'S AXIOMS

→ are a set of sound and complete rules used to infer FDs on a databas

→ You need 3 axioms that are essential

1) Reflexivity

- $\boxed{\text{If } Y \subseteq X, \text{ then } X \to Y}$

i.e. If I have a set of attributes {A, B, C}, the values of these 3 attributes will always determine the value of A, value of B, value of C, value of AB, value of BC, ... (all possible subsets)

2) Augmentation

- $\boxed{\text{If } X \to Y, \text{ then } XZ \to YZ \text{ for any } Z}$

added in union

You can augment the LHS & RHS w/ the same set of attributes

3) Transitivity

- $\boxed{\text{If } X \to Y \text{ and } Y \to Z, \text{ then } X \to Z}$

→ There are 2 other axioms which can be derived from essential axioms

4) Union

- $\boxed{\text{If } X \to Y \text{ and } X \to Z, \text{ then } X \to YZ}$

5) Decomposition

- $\boxed{\text{If } X \to YZ \text{ then } X \to Y \text{ and } X \to Z}$

CLOSURE OF A SET OF FDs

→ Let F be a set of FDs

The $\boxed{\text{closure } F^+ \text{ of } F}$ is the set of all FDs implied by the FDs in F

→ i.e. Closure of $\{A \to B, B \to C\}$

1. $A \to B$ [given]
2. $B \to C$ [given]
3. $A \to C$ [by transitivity of 1 and 2]
4. $A \to BA$ [by augmentation of 1 w/ A]
5. $AB \to AC$ [by augmentation of 2 w/ A]
6. $ABC \to AC$ [by reflexivity]

⋮

Cumbersome to compute, so we can use attribute closure (next pg.)

ATTRIBUTE CLOSURE

→ The closure $C_F(X)$ of a set X of attributes w.r.t. a set F of FDs is the set of attributes we can derive from X using the FDs in F

PROPERTIES

1) $X \subseteq C_F(X)$
2) If $X \subseteq Y$, then $C_F(X) \subseteq C_F(Y)$
3) $C_F(C_F(X)) = C_F(X)$

→ Soln to the implication problem:

$$F \models Y \rightarrow Z \quad \text{if and only if} \quad Z \subseteq C_F(Y)$$

i.e. closure of A w.r.t. $\{AB \rightarrow C,\ A \rightarrow B,\ CD \rightarrow A\}$

Unused	Closure
$AB \rightarrow C$ ✗ cannot use bc. $AB \not\subseteq A$	A
$A \rightarrow B$ ✓	B
$CD \rightarrow A$	

↓

Unused	Closure
$AB \rightarrow C$ ✓	A
$CD \rightarrow A$	B
	C

↓

Unused	Closure
$CD \rightarrow A$ ✗ $CD \not\subseteq ABC$	A
	B
	C

$\boxed{\therefore\ C_F(A) = ABC}$

This tells us that $F \models A \rightarrow ABC$

KEYS

→ let $\underline{R}$ be a relation w/ set of attributes $\underline{U}$, and FDs $\underline{F}$

→ $X \subseteq U$ is a $\boxed{\text{key}}$ of R if $F \models X \to U$

Equivalently, X is a key if $C_F(X) = U$

→ $\boxed{\text{Candidate key}}$ is a key w/ a minimal set of attributes

↳ consists of keys X such that, for each $Y \subset X$, Y is not a key

↳ if you remove any attribute from X, it stops being a key

→ $\boxed{\text{Prime attribute}}$ is an attribute of a candidate key

→ Given a set F of FDs on attributes U, how do we compute ALL candidate keys?

1. ck := $\emptyset$
2. G := DAG of the powerset 2^U of U
3. Repeat until G is empty:

    Find a node X w/o children

    ↳ start w/ empty set

    if $C_F(X) = U$:

        ck := ck ∪ {X}

        Delete X and all its ancestors from G

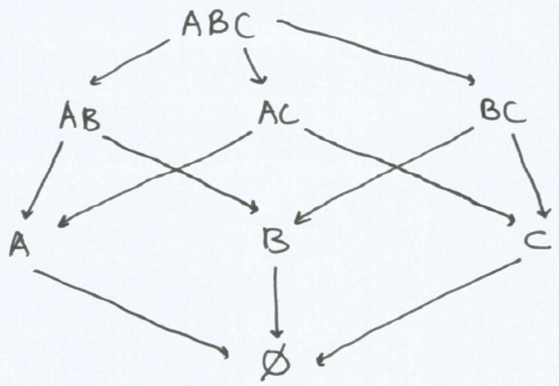
    else :

        Delete X from G

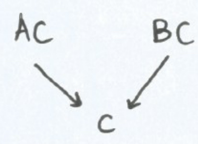
i.e. $F = \{AB \to C,\ BC \to A\}$

$U = ABC$

We first initialize: ck = {}, G :



↓ After 4



1) Start w/ Ø, $C_F(\{\}) = \{\}$ → ELSE

2) Can pick A, $C_F(A) = A$ → ELSE

3) Can pick B, $C_F(B) = B$ → ELSE

4) Can pick AB, $C_F(AB) = ABC$ → ck = {AB}

5) Can pick C, $C_F(C) = C$ → ELSE

6) Can pick AC, $C_F(AC) = AC$ → ELSE

7) Can pick BC, $C_F(BC) = BCA$ → ck = {AB, BC}

8) G is now empty

IMPLICATION OF INDs

→ Given a set of INDs, what other INDs can we entail from it?

AXIOMATIZATION

1) Reflexivity: $R[X] \subseteq R[X]$

2) Transitivity: If $R[X] \subseteq S[Y]$ and $S[Y] \subseteq T[Z]$, then $R[X] \subseteq T[Z]$

3) Projection: If $R[X, Y] \subseteq S[W, Z]$ with $|X| = |W|$, then $R[X] \subseteq S[W]$

4) Permutation: If $R[A_1, \dots, A_n] \subseteq S[B_1, \dots, B_n]$,

then $R[A_{i_1}, \dots, A_{i_n}] \subseteq S[B_{i_1}, \dots, B_{i_n}]$,

where $i_1, \dots, i_n$ is a permutation of $1, \dots, n$

↳ With these 4 rules, we get a sound & complete derivation procedure for INDs

FDs & INDs TOGETHER

→ We have seen that:
→ Given a set F of FDs and an FD f, we can decide whether $F \models f$
→ Given a set G of INDs and an IND g, we can decide whether $G \models g$

→ What about $F \cup G \models f$ or $F \cup G \models g$?

↳ This problem is undecidable; no algo. can solve it

→ Even if we restrict FDs to key constraints and INDs to foreign key constraints, the implication problem remains undecidable.

→ There is a case when the problem is decidable and this is when we consider **UNARY INCLUSION DEPENDENCIES (UINDs)**

↳ INDs where the list of attributes tht. we have in the dependency is of length 1

↳ With this, the implication problem for FDs + UINDs is decidable in polynomial time