Improved Third Normal Form Relations (LTK)

(by Ling, Tompa, and Kameda, TODS 1981)

Preparatory Algorithm.

Input. **A** - a set of attributes, **F** - a set of FDs on **A**.

Step 1. *(Remove extraneous attributes and dependencies)*

- Eliminate from both sides of each FD in **F** all attributes whose elimination leaves a set of FDs having a closure equal to $\mathbf{F}^+$.

- Eliminate from that modified set all FDs whose right side is empty. Let $\mathbf{F}_1$ be the resulting set.

Step 2. *(Partition the FDs)*

- Partition $\mathbf{F}_1$ into a set of classes C such that all FDs in each class have properly equivalent left sides.

Step 3. *(Construct relations)*

- Each class form a relation

- Let **R**= $\{R_1, \ldots, R_n\}$ be the set of relations constructed. Each left side of FDs in each class is a key (called **synthesized key**) of the constructed relation.

Step 4. *(Augment the schema, if necessary, for reconstructibility)*

If for each relation R_i in $\mathbf{R}$ with $\mathbf{A}_i$ the set of attributes in R_i such that $\mathbf{A}_i \not\rightarrow \mathbf{A} \in \mathbf{F}_1^+$

then construct a minimal subset A' of A such that

$$\mathbf{A}' \rightarrow \mathbf{A} \in \mathbf{F}_1^+$$

This is an **all key relation**. (i.e. all attributes of the relation is the key of the relation).

Output. The set of relations constructed is called the **preparatory relational schema**.

Note. The relations constructed may not in 3NF. This preparatory algorithm does not remove transitive dependencies.

Let R be preparatory relational schema consisting of relations R_i, each having a set of attributes $\mathbf{A}_i$ and a set K_i of **synthesized keys**.

$$\mathbf{R} = \{ R_1 < \mathbf{A}_1, K_1 >, R_2 < \mathbf{A}_2, K_2 >, \ldots, R_n < \mathbf{A}_n, K_n > \}$$

Define G_i be the set of **synthesized FDs** in relation R_i

i.e. $G_i = \{ K \rightarrow A_i - K \mid K \in K_i \}$

Let $G = \bigcup_{i=1}^{n} G_i$

We can show that

$$G^+ = \mathbf{F}^+$$

Definition.

Given a relational schema **R**, an attribute B is **superfluous** in a relation $R_i \in$ **R** if its removal form R_i does not affect covering or reconstructibility.

The FDs that do not involve B in R_i may be defined as:

$$D_i(\mathrm{B}) = \bigcup_{j \neq i} \{ K \rightarrow \mathbf{A}_j - K \mid K \text{ is a key of } R_j \}$$
$$\cup \{ K \rightarrow \mathbf{A}_i - K - \mathrm{B} \mid K \text{ is a key of } R_i \text{ and } \mathrm{B} \notin K \}$$

Note. D_i(B) is defined in terms of **all keys** for all relations in **R**.

The set of all **synthesized FDs** that do not involve B in R_i is:

$$G_i'(\mathrm{B}) = \bigcup_{j \neq i} G_j \cup \{ K \rightarrow \mathbf{A}_i - K - \mathrm{B} \mid \mathrm{B} \notin K \text{ and } K \in K_i \}$$

Where K_i is the set of all synthesized keys of R_i.

e.g. $\mathbf{F} = \{\mathrm{A} \rightarrow \mathrm{B}, \mathrm{B} \rightarrow \mathrm{A}, \mathrm{AC} \rightarrow \mathrm{DE}, \mathrm{BD} \rightarrow \mathrm{C}\}$

$\mathbf{R} = \{R_i<\mathrm{AB}, \{\mathrm{A}, \mathrm{B}\}>, R_2<\mathrm{ABCDE}, \{\mathrm{AC}, \mathrm{BD}\}>\}$

i.e. $\mathbf{R} = \{R_1 (\underline{\mathrm{A}}, \underline{\mathrm{B}},), R_2 (\underline{\mathrm{A, C}}, \underline{\mathrm{B, D}}, \mathrm{E})\}$

We have $D_2(\mathrm{B}) = \{\mathrm{A} \rightarrow \mathrm{B}, \mathrm{B} \rightarrow \mathrm{A}, \mathrm{AC} \rightarrow \mathrm{DE}, \mathbf{AD \rightarrow CE}\}$

Since **AD** is an **implicit key** of R_2

However

$G_1=\{\mathrm{A} \rightarrow \mathrm{B}, \mathrm{B} \rightarrow \mathrm{A}\}$, $G_2=\{\mathrm{AC} \rightarrow \mathrm{BDE}, \mathrm{BD} \rightarrow \mathrm{ACE}\}$

$G_2'(\mathrm{B}) = \{\mathrm{A} \rightarrow \mathrm{B}, \mathrm{B} \rightarrow \mathrm{A}, \mathrm{AC} \rightarrow \mathrm{DE}\}$

Note.

(1) $D_2(\mathrm{B})^+ \neq G_2'(\mathrm{B})^+$

(2) Either A or B is **superfluous** in R_2. How to test it?

Definition

An attribute B is **restorable** in R_i

iff $\exists\ K \in K_i$ such that $B \notin K$

and $K \rightarrow B \in G_i'(B)^+$

(i.e. the B-value in R_i is derivable from the rest of the schema)

Definition

B is **non-essential** in R_i

iff $\forall\ K \in K_i$ such that $B \in K$,
$\exists\ K' \subseteq \mathbf{A}_i - B\ \ni\ K' \rightarrow \mathbf{A}_i \in G^+$
and $K \rightarrow K' \in G_i'(B)^+$

i.e. the closure of K relative to $G_i'(B)$ contains a key K' for R_i (possibly non-synthesized) such that $B \notin K'$.

(i.e. B is not required to derive the value of any other attribute of R_i)

Theorem

Let $\mathbf{R}$ be a preparatory relational schema including R_i, and let B be an attribute in R_i. The attribute B is superfluous in R_i.

iff it is restorable and nonessential in R_i.

e.g. $\mathbf{R} = \{R_1(\underline{A}, B, C), R_2(\underline{B}, C)\}$

Attribute C in R_1 is restorable since

$$A \rightarrow C \in G_1'(C)^+$$

Where $G_1 = \{A \rightarrow BC\}$, $G_2 = \{B \rightarrow C\}$

$$G_1'(C) = \{A \rightarrow \mathbf{B}\} \cup G_2$$

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

Attribute C in R_1 is non-essential

since it is not contained in any explicit key of R_1.

So, attribute C is superfluous in R_1.

Note that B is also non-essential in R_1

C is also non-essential in R_2

They are both non-primes in R_1 and R_2 respectively.

Definition

A relation R_i in a preparatory relational schema **R** is in **improved third normal form** (or **LTK normal form**) if each non-essential attribute is not restorable in R_i.

Theorem:

If R_i is in improved 3NF then it is also in 3NF.

Theorem:

If all relations in a preparatory relational schema **R** *are in improved 3NF, then there is no superfluous attribute in any relation of* **R**.

Note.

(1). Any non-prime attribute is non-essential.

(2). If $A \rightarrow B$ is a T.D. in a relation, then B is restorable.

Superfluous attribute detection algorithm

Input **R**, a preparatory relational schema
B, an attribute in R_i
(Test whether B is superfluous in R_i)

Step 1. Mark B superfluous
Construct
$\boldsymbol{K_i'} = \{ K \in K_i \mid B \notin K\}$
(ie. all synthesized keys of R_i which do not contain B)
Construct $G_i'(B)$

Step 2. **(Check restorability)**

If K_i' is empty *(B is not restorable)*
then mark B non-superfluous and RETURN
Else if $\exists K \in K_i'$ such that
$K \rightarrow B \in G_i'(B)^+$
then goto step 3
(ie. B is restorable in R_i)
else mark B non-superfluous and RETURN

Step 3. **(Check non-essentiality)**

(At this point, B is restorable)

For each key K in K_i - K_i' **(i.e. B $\in$ *K*)**

WHILE B is marked superfluous *DO*

IF $K \rightarrow \mathbf{A}_i \notin G_i'(B)^+$

then let M = the closure of K relative to $G_i'(B)$

IF $(M \cap \mathbf{A}_i) - B \rightarrow \mathbf{A}_i \notin G^+$

then mark B non-superfluous

else insert into K_i' any key of R_i contained in $(M \cap \mathbf{A}_i)$- B

(find a new synthesized key)

Output. If B is marked superfluous

(i.e. B is really superfluous)

then output K_i'

(the new set of synthesized key of R_i, none of them contains B).

else output $\varnothing$.

Deletion Normalization Algorithm

Input. Given a set of attributes **A** and a set of FD_s of.

Step 1. (prepare a relational schema)

Use the preparatory algorithm for **A** and **F** to yield **R.**
(**R** = *{R_1, …, R_n} together with their synthesized keys)*

Step 2.(Test each relation for superfluous attributes)

For i =1 to |**R**| Do
For each B in $\mathbf{A}_i$ *(Test for each attribute in* $\mathbf{A}_i$*. i.e. relation* R_i*)*
if the superfluous attribute detection algorithm return a non-empty set K_i' for R_i and B
then construct R_i' such that $\mathbf{A}_i = \mathbf{A}_i$ - B
and K_i' is the returned set of keys
Replace R_i by R_i' in **R**.
(i.e. remove the superfluous attribute B from R_i *with a new set of synthesized keys)*

Output. **R**, a relational schema in improved 3NF.
(LTK normal form).

Definition.

A relation R_i in a relational schema **R** is in **improved BCNF** if no attribute is restorable in R_i.

Result.

Improved BCNF relation is also in BCNF.

Result.

Improved BCNF relation is also in improved 3NF

Example 1. Let **A** = {A, B, C, D, E, F} and

F = {AB → CD, A → E, B → F, EF → C}

The preparatory algorithm will produce the relational schema

R = {R_1 (A, B, D), R_2 (A, E), R_3 (B, F), R_4 (E, F, C)}

Note that C is an extraneous attribute in AB → CD.

There is no attribute that is both nonessential and restorable in **R**, therefore all relations are in improved 3NF.

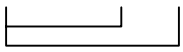
Example 2.

Let **A** = {A, B, C, D, E, F}

F = {AD → B, B → C, C → D, AB → E, AC → F}

The preparatory algorithm will produce

R = {R_1 (A, B, C, D, E, F), R_2 (B, C), R_3 (C, D)}



The attribute C is nonessential and restorable in R_1 and therefore can be dropped form R_1.

The resulting schema is in improved 3NF.

Test C is superfluous in R_1.

Step 1. $K_1' = \{AB, AD\}$

Step 2. $G_1'(C) = \{AB \rightarrow DEF, AD \rightarrow BEF\} \cup G_2 \cup G_3$
$= \{AB \rightarrow DEF, AD \rightarrow BEF, B \rightarrow C, C \rightarrow D\}$

Since $AB \rightarrow C \in G_1'(C)^+$
C is restorable in R_1.

Step 3. The only explicit key in R_1 which contains C is AC.

Since $C \rightarrow D \in G_1'(C)$
$AC \rightarrow AD \in G_1'(C)^+$
and $AD \rightarrow ABCDEF \in G_1'(C)^+$
C is non-essential in R_1

Hence C is superfluous
(not need to find another explicit key)

Step 4. $K_1' = \{AB, AD\}$
and C is marked superfluous.

Example Let **A** = { S#, IC#, Name, C#, CName,
Description, mark, Year}

F = {S# → IC#, Name
IC# → S#
C# → CName, Description
CName → C#
S#, C# → mark
IC#, Cname → Year}

The preparatory algorithm *(also Bernstein's algorithm)* will produce the following relations

R_1 (S#, IC#, Name)
R_2 (C#, CName, Description)
R_3 (S#, C#, IC#, CName, mark, Year)

Either S# or IC# is superfluous in R_3,

also either C# or CName is superfluous in R_3.

One of the 4 possible improved 3NF schema is

R = {R_1 (S#, IC#, Name)
R_2 (C#, CName, Description)
R_3 (S#, CName, mark, Year)}