Basi di dati

Schema

An **attribute** is a $(\text{name}, \text{domain})$ pair; we can define the $dom()$ function on a set of names, which associates to each **name** a specific **domain** *(different attributes can have the same domain)*

$$dom : \{\text{name}_1, ..., \text{name}_n\} \to \{\text{domain}_1, ..., \text{domain}_k\}$$
$$\text{name}_i \mapsto \text{domain}_j$$

PDF 7 slide 2

A **relation schema** $R = \{A_1, A_2, ..., A_n\}$ is a set of attributes

Tuples & instances

PDF 7 slide 3 Given a relation schema $R = A_1 A_2 \dots A_n$, a **tuple** t on R is a function such that

$$t : R \to \bigcup_{i=1}^{n} dom(A_i)$$
$$A_i \mapsto a \in dom(A_i)$$

Given a relation schema R, a subset $X \subseteq R$ and t a tuple on R, the **reduction** of t on X is defined as

$$t[X] = \{ (A, t[A]) \mid A \in X \}$$

PDF 7 slide 4 Given a relation schema R, a subset $X \subseteq R$ and t_1, t_2 tuples on R

$$t_1[X] = t_2[X] \iff t_1[A] = t_2[A] \; \forall A \in X$$

PDF 7 slide 5 Given a relation schema R and $t_1, t_2, \dots, t_k$ tuples on R, a set $r = \{ t_1, t_2, \dots, t_k \}$ is an **instance** of R

Functional dependencies

PDF 7 slide 6

Given a relation schema R and $X, Y \in \mathcal{P}(R) \setminus \{\varnothing\}$ we have that (X, Y) is a **functional dependency** on R *(noted as $X \to Y$)*

PDF 7 slide 7

Given a relation schema R and a functional dependency $X \to Y$ defined on R we say that an **instance** r of R **satisfies** the functional dependency $X \to Y$ if

$$\forall t_1, t_2 \in r \quad t_1[X] = t_2[X] \implies t_1[Y] = t_2[Y]$$

Instance legality & closure

PDF 7 slide 14

Given a relation schema R and a set F of functional dependencies defined on R, an **instance** r of R is **legal** if it satisfies every dependency in F

$$\forall X \rightarrow Y \in F \quad \forall t_1, t_2 \in r \quad t_1[X] = t_2[X] \implies t_1[Y] = t_2[Y]$$

PDF 7 slide 20

Given a relation schema R and a set F of functional dependencies defined on R, the closure of F is the **set of functional dependencies** that are satisfied by every **legal instance** of R

$$F^+ = \{\, V \rightarrow W \mid \forall \text{ legal } r \text{ of } R, r \text{ satisfies } V \rightarrow W \,\}$$

> $V \rightarrow W$ doesn't necessarily have to be in F

$F \subseteq F^+$

PDF 7 slide 21

$$F \subseteq F^+$$

Proof

$$F^+ = \{\, V \to W \mid \forall \text{ legal } r \text{ of } R, r \text{ satisfies } V \to W \,\}$$

By definition r is legal if it satisfies every dependency $X \to Y \in F \implies$ given $X \to Y \in F$, every legal instance of R satisfies $X \to Y \implies X \to Y \in F^+$

Keys

PDF 7 slide 22

Given a relation schema R and a set F of functional dependencies on R, $K \subseteq R$ is a **key** of R if

- $K \rightarrow R \in F^+$
- $\forall K' \subset K,\ K' \rightarrow R \notin F^+$

> " $\subset$ " means **proper subset**, which implies that $K \neq K'$

Trivial dependencies

PDF 7 slide 26

Given a schema R and $X, Y \in \mathcal{P}(R) \setminus \{ \varnothing \} : Y \subseteq X$, we have that **every instance** r of R **satisfies** the dependency $X \to Y$

Proof

Given an instance r of R, $\forall t_1, t_2 \in r$ we have that

$t_1[X] = t_2[X] \implies$ by definition $t_1[A] = t_2[A] \; \forall A \in X \implies$ as $Y \subseteq X$ we have that $t_1[A'] = t_2[A'] \; \forall A' \in Y \implies$ by definition $t_1[Y] = t_2[Y]$

As $t_1[X] = t_2[X] \implies t_1[Y] = t_2[Y]$ we have that r satisfies $X \to Y$

Decomposition

PDF 7 slide 27

Given a schema R and a set of functional dependencies F on R, we have that

$$X \to Y \in F^+ \iff X \to A \in F^+ \; \forall A \in Y$$

Proof

$X \to Y \in F^+ \implies \forall \text{ legal } r \text{ of } R \quad \forall t_1, t_2 \in r \quad t_1[X] = t_2[X] \implies t_1[Y] = t_2[Y] \implies t_1[A] = t_2[A] \; \forall A \in Y \implies X \to A \in F^+ \; \forall A \in Y$

$X \to A \in F^+ \; \forall A \in Y \implies \forall \text{ legal } r \text{ of } R \quad \forall t_1, t_2 \in R \quad t_1[X] = t_2[X] \implies t_1[A] = t_2[A] \; \forall A \in Y \implies t_1[Y] = t_2[Y] \implies X \to Y \in F^+$

F^A

PDF 8 slide 3 F^A is a set of functional dependencies on R such that

- $X \to Y \in F \implies X \to Y \in F^A$
- $Y \subseteq X \in R \implies X \to Y \in F^A$ *(refelxivity)*
- $\forall Z \in R, X \to Y \in F^A \implies ZX \to ZY \in F^A$ *(augmentation)*
- $X \to Y, Y \to Z \in F^A \implies X \to Z \in F^A$ *(transitivity)*

PDF 8 slide 6 derivates

- $X \to Y, X \to Z \in F^A \implies X \to YZ \in F^A$ *(union)*
- $X \to Y \in F^A \land Z \subseteq Y \implies X \to Z \in F^A$ *(decomposition)*
- $X \to Y, WY \to Z \in F^A \implies WX \to Z \in F^A$ *(pseudotransitivity)*

PDF 8 slide 8 $X \to A_1 A_2 \ldots A_n \in F^A \iff \forall i = 1, \ldots, n \quad X \to A_i \in F^A$

Derivates *(Proofs)*

Union

$X \to Y, X \to Z \in F^A \implies$ by augmentation $X \to XY, XY \to YZ \in F^A \implies$ by transitivity $X \to YZ \in F^A$

Decomposition

$X \to Y \in F^A \wedge Z \subseteq Y \implies Y \to Z \in F^A \implies$ by transitivity $X \to Z \in F^A$

Pseudotransitivity

$X \to Y, WY \to Z \in F^A \implies$ by augmentation $WX \to WY \in F^A \implies$ by transitivity $WX \to Z \in F^A$

$(X)_F^+$

PDF 8 slide 9

Given a relation schema R, a set F of dependencies on R and $X \subseteq R$. The **closure** of X with respect to F, denoted $(X)_F^+$ is defined as

$$(X)_F^+ = \{\, A \in R \mid X \to A \in F^A \,\}$$

We have that $X \subseteq (X)_F^+$

Proof

$\forall A \in X$ by reflexivity $X \to A \in F^A \implies$ by definition $A \in (X)_F^+ \implies X \subseteq (X)_F^+$

> We can use Armstrong's axioms as $(X)_F^+$ is defined of F^A

> NOTE: $(X)

Lemma of closure

PDF 8 slide 10

Let R be a schema and F a set of functional dependencies on R

$$X \to Y \in F^A \iff Y \subseteq (X)^+_F$$

Proof

$X \to Y \in F^A \implies$ by decomposition $X \to A \in F^A \ \forall A \in Y \implies$ by definition
$A \in (X)^+_F \ \forall A \in Y \implies Y \subseteq (X)^+_F$

$Y \subseteq (X)^+_F \implies A \in (X)^+_F \ \forall A \in Y \implies$ by definition $X \to A \in F^A \ \forall A \in Y \implies$ by union
$X \to Y \in F^A$

$F^+ = F^A$

PDF 8 slide 11

Let R be a relation schema and F a set of functional dependencies on R then $F^+ = F^A$

Proof

Let F_i be the value of F after the i-th application of an Armstrong's axiom, with $F_0 = F$

$F^A \subseteq F^+$

Base case

$$F_0 = F \subseteq F^+ \implies F_0 \subseteq F^+$$

Inductive step

$$F_i \subseteq F^+ \implies F_{i+1} \subseteq F^+$$

Let $X \to Y \in F_{i+1}$, either

- $X \to Y \in F_i \implies$ by HP $X \to Y \in F^+$
- $X \to Y \in F_{i+1} \setminus F_i$, which means that $X \to Y$ has been optained through one of the axioms

$F^A \subseteq F^+$

Reflexivity

$Y \subseteq X \implies$ given that $X \to Y$ is satisfied by every instance $X \to Y \in F^+$

Augmentation

$Z \subseteq R, X = ZV, Y = ZW \wedge V \to W \in F_i$ given $t_1, t_2 \in r$ legal instance of R we have that $t_1[X] = t_2[X] \implies (t_1[V] = t_2[V] \implies$ by HP $t_1[W] = t_2[W]) \wedge t_1[Z] = t_2[Z] \implies t_1[Y] = t_2[Y]$

Transitivity

$X \to Z, Z \to Y \in F_i \implies$ by HP $\forall$ legal r of $R, \forall t_1, t_2 \in r, t_1[X] = t_2[X] \implies t_1[Z] = t_2[Z] \implies t_1[Y] = t_2[Y]$ we have that $t_1[X] = t_2[X] \implies t_1[Y] = t_2[Y] \implies X \to Y \in F^+$

$F^+ \subseteq F^A$ *(legal instance)*

Given $X \subseteq R$ we can build an instance $r = \{t_1, t_2\}$ on R such that

r	$(X)^+_F$					$R \setminus (X)^+_F$				
t_1	1	1	1	...	1	1	1	1	...	1
t_2	1	1	1	...	1	0	0	0	...	0

Let's verify that r is a legal instance. Given $V \rightarrow W \in F$, as $V, W \neq \varnothing$ by definition, we could have

- $V \nsubseteq (X)^+_F \implies \exists A \in V : A \in R \setminus (X)^+_F \implies t_1[V] \neq t_2[V] \implies r$ satisfies $V \rightarrow W$
- $V \subseteq (X)^+_F$, we could have that
  - $W \subseteq (X)^+_F \implies t_1[V] = t_2[V] \wedge t_1[W] = t_2[W] \implies r$ satisfies $V \rightarrow W$
  - $W \nsubseteq (X)^+_F \implies \exists A \in W : A \in R \setminus (X)^+_F \implies t_1[V] = t_2[V] \wedge t_1[W] \neq t_2[W]$

$F^+ \subseteq F^A$ *(legal instance)*

In the last case r doesn't satisfy $V \to W$, so we have to show that it can't happen. Let's suppose that $\exists V \to W \in F$ such that r doesn't satisfy $V \to W$; by construction we have that

$$V \subseteq (X)^+_F \wedge \exists A \in W : A \in R \setminus (X)^+_F \implies A \notin (X)^+_F$$

We have that

- $V \subseteq (X)^+_F \implies$ by the lemma of closure $X \to V \in F^A$
- $A \in W \implies$ by decomposition $V \to A \in F^A$

By transitivity $X \to A \in F^A \implies$ by the lemma of closure $A \in (X)^+_F$ which is a contraddiction

Legality

In the first 2 cases r satisfies $V \to W \in F$, case 3 can't happen $\implies$ r is a legal instance of R

$F^+ \subseteq F^A$

Let's consider $X \to Y \in F^+$

By definition we have that $X \subseteq (X)_F^+ \implies$ by construction $t_1[X] = t_2[X] \implies$ by hypotesis and given that r is a legal instance $t_1[Y] = t_2[Y] \implies$ by the lemma $Y \subseteq (X)_F^+ \implies X \to Y \in F^A$

$F^+ = F^A$

Given that $F_i \subseteq F^+ \ \forall i \in \mathbb{N}$ and $F^+ \subseteq F^A$ we have that $F^+ = F^A$

3NF

PDF 9 slide 14

Given a relation schema R and a set of functional dependencies F on R.

R is in **3NF** if $\forall X \rightarrow A \in F^+ : A \notin X$ either

- A is prime *(belongs to a key)*
- X is superkey

3NF pt.2

PDF 10 slide 4

Let R be a relation schema and F a set of functional dependencies on R

An attribute $A \in R$ **partially** depends on a key K if

- $\exists X \subset R : A \notin X \wedge X \rightarrow A \in F \wedge X \subset K$
- A isn't prime

An attribute $A \in R$ **transitively** depends on a key K if

- $\exists X \subset R : A \notin X \wedge X \rightarrow A \in F \wedge K \rightarrow X \in F$
- X isn't a key
- A isn't prime

> $X \subset R$ means that $X \neq R$, otherwise X would be a superkey, as $R \rightarrow R \in F^A = F^+$

3NF pt.3

PDF 10 slide 5

Given a schema R and a set of functional dependencies F on R, **TFAE**

- R is in 3NF
- there are **no attributes that partially or transitively depend on a key**
- $\forall X \rightarrow A \in F^+ : A \notin X$ either:
  - A is prime *(belongs to a key)*
  - X is superkey

Proof

> TODO: I have it, I just have to write it out in $\LaTeX$

BCNF *(Boyce-Codd)*

PDF 10 slide 20

A relation schema R is in **Boyce-Codd Normal Form** (BCNF) when every determinant in F is a superkey.

A relation that respects Boyce-Codd Normal Form is also in **3NF**, but the opposite is not true.

$(X)_F^+$

PDF 11 slie 5

```
def clousure(R, F, X):
    Z = X
    S = { A ∈ R | Y → V ∈ F ∧ Y ⊆ Z ∧ A ∈ V }

    if S ⊆ Z:
        return Z

    return closure(R, F, Z ∪ S)
```

$(X)^+_F$

PDF 11 slide 8

The algorithm `closure()` correctly computes the closure of a set of attributes X respectively to a set F of functional dependencies on R

Proof

Let's consider Z_i, S_i the values of Z and S at the i-th call of the function and $Z_f, S_f \mid S_f \subseteq Z_f$ the values of Z, S at the last call of the function. Let's prove by induction that $Z_i \subseteq (X)^+_F$

$Z_i \subseteq (X)^+_F$

Base case

$Z_0 = X \subseteq (X)^+_F$

Inductive step $Z_i \subseteq (X)^+_F \implies Z_{i+1} \subseteq (X)^+_F$

Given that $Z_{i+1} = Z_i \cup S_i$ then if $A \in Z_{i+1}$ either

- $A \in Z_i \implies$ by HP $A \in (X)^+_F$
- $A \in S_i \implies$ by construction $\exists Y \to V \in F \mid Y \subseteq Z_i \wedge A \in V \implies$ by HP $Y \subseteq (X)^+_F \implies$ by the lemma of closure $X \to Y \in F^A$ and by decomposition $Y \to A \in F^A \implies$ by transitivity $X \to A \in F^A \implies$ by definition $A \in (X)^+_F$

$(X)^+_F \subseteq Z_f$ *(legal instance)*

Given Z_f we can build an instance $r = \{t_1, t_2\}$ on R such that

r	Z_f					$R \setminus Z_f$				
t_1	1	1	1	...	1	1	1	1	...	1
t_2	1	1	1	...	1	0	0	0	...	0

Let's verify that r is a legal instance. Given $V \to W \in F$ as $V, W \neq \varnothing$ we could have either

- $V \nsubseteq Z_f \implies \exists A \in V : A \in R \setminus Z_f \implies t_1[V] \neq t_2[V] \implies r$ satisfies $V \to W$
- $V \subseteq Z_f$
  - $W \subseteq Z_f \implies$ by construction $t_1[V] = t_2[V] \land t_1[W] = t_1[W] \implies r$ satisfies $V \to W$
  - $W \nsubseteq Z_f \implies$ by construction $t_1[V] = t_2[V] \land t_1[W] \neq t_2[W]$

$(X)_F^+ \subseteq Z_f$ *(legal instance)*

Let's suppose that $\exists V \rightarrow W \in F : r$ doesn't satisfy $V \rightarrow W \implies$ by construction

$$V \subseteq Z_f \land \exists A \in W : A \in R \setminus Z_f \implies A \notin Z_f$$

Given that $V \subseteq Z_f \land V \rightarrow W \in F \land A \in W \implies$ by construction of $S_f,\ A \in Z_f$ which is a contraddiction

Legality

In the first 2 cases r satisfies $V \rightarrow W \in F$ case 3 can't happen $\implies r$ is a legal instance of R

$(X)_F^+ \subseteq Z_f$

Let's consider $A \in (X)_F^+$

Given that $X \to A \in F^A = F^+$ and r is a legal instance $\implies$ r satisfies $X \to Y$, and given that by construction $X \subseteq Z_f \implies t_1[X] = t_2[X] \implies$ by definition $t_1[A] = t_2[A] \implies$ by construction $A \in Z_f$

$Z_f = (X)_F^+$

Given that $Z_i \subseteq (X)_F^+ \; \forall i \in \mathbb{N}$ and $(X)_F^+ \subseteq Z_f$, we have that $Z_f = (X)_F^+$

Intersection Rule

PDF 12 slide 19

Given a relation scheme R and a set of functional dependencies F on R

$$X := \bigcap_{V \to W \in F} R - (W - V)$$

If $X \to R \in F^+$ then the intersection is the only key to R otherwise there are multiple keys, and **ALL** of them must be identified to check if R is in **3NF**

Decomposition

PDF 13 slide 8

Let R be a relation scheme, a decomposition ρ of R is such that

$$\rho = \{ R_1, R_2, ..., R_k \} \subseteq \mathcal{P}(R) : \bigcup_{i=1}^{k} R_i = R$$

Equivalence

PDF 13 slide 12

Let F and G be two sets of functional dependencies, we can define an equivalence relation

$$F \equiv G \iff F^+ = G^+$$

- reflexivity $F \implies F^+ = F^+ \implies F \equiv F$
- simmetry $F \equiv G \implies F^+ = G^+ \implies G^+ = F^+ \implies G \equiv F$
- transitivity $F \equiv G \wedge G \equiv H \implies F^+ = G^+ \wedge G^+ = H^+ \implies F^+ = H^+ \implies F \equiv H$

PDF 13 slide 14

Let F and G be two sets of functional dependencies

$$F \subseteq G^+ \implies F^+ \subseteq G^+$$

$$F \subseteq G^+ \implies F^+ \subseteq G^+$$

Base case

$F_0 = F \subseteq G^+ \implies F_0 \subseteq G^+$

Inductive Step

$F_i \subseteq G^+ \implies F_{i+1} \subseteq G^+$

$X \to Y \in F_{i+1} \implies X \to Y$ has been optained through

- reflexivity $Y \subseteq X \implies$ given that $X \to Y$ is satisfied by every instance $X \to Y \in G^+$
- augmentation $\exists Z \subseteq R, V \to W \in F_i \mid X = ZV, Y = ZW$
- transitivity

> TODO

Preserving F

PDF 13 slide 15

Let R be a relation scheme, F a set of functional dependencies on R and $\rho = \{ R_1, R_2, ..., R_k \}$ a decomposition of R, we say that ρ preseves F if

$$F \equiv G = \bigcup_{i=1}^{k} \pi_{R_i}(F)$$

Where

$$\pi_{R_i}(F) = \{ X \rightarrow Y \in F^+ \mid XY \subseteq R_i \}$$

PDF 13 slide 16

Given the definition of G, it will always be that $G \subseteq F^+ \implies G^+ \subseteq F^+$ so it is enough to verify that $F \subseteq G^+$

Dependency preservation

PDF 13 slide 17

```
def preserves_dependencies(R, F, ρ):
    for X → Y ∈ F:
        if Y ⊈ closure_G(R, F, ρ, X):
            return false

    return true
```

This algorithm is enough as we just have to check wether $F \subseteq G^+$

Given $X \to Y \in F$ we have that $X \to Y \in G^+ = G^A \iff$ by the lemma of closure $Y \subseteq (X)^+_G$

$(X)^+_G$

```
def clousure_G(R, F, X, ρ):
    Z = X
    S = ∅

    for P ∈ ρ:
        S = S ∪ (closure(R, F, Z ∩ P) ∩ P)

    if S ⊆ Z
        return Z

    return closure_G(R, F, Z ∪ S)
```

PDF 13 slide 23 Let R be a relation schema, F a set of functional dependencies on R and $\rho = \{ R_1, R_2, \ldots, R_k \}$ a decomposition of R and $X \subseteq R$ the algorithm `closure_G()` correctly computes $(X)^+_G$

$Z_f \subseteq (X)^+_G$

Let Z_i, S_i the values of Z and S at the i-th call of the function, with $Z_0 = X$, and $S_f \subseteq Z_f$

Base case

$Z_0 = X \subseteq (X)^+_G \implies Z_0 \subseteq (X)^+_G$ by HP

Inductive step

$Z_i \subseteq (X)^+_G \implies Z_{i+1} \subseteq (X)^+_{G'}$, given that $S_i = \bigcup_{j=1}^{k} (Z_i \cap R_j)^+_F \cap R_j$

Let $A \in Z_{i+1} = Z_i \cup S_i \implies \exists j : A \in (Z_i \cap R_j) \cap R_j \implies Z_i \cap R_j \to A \in G^A$

By HP we have that $Z_i \subseteq (X)^+_G \implies X \to Z_i \in G^A$, let $Z_i = (Z_i \cap R_j) \cup V$ by decomposition we have that $X \to Z_i \cap R_j \in G^A \implies$ by transitivity $X \to A \in G^A$

$X \subseteq Y \implies (X)^+_F \subseteq (Y)^+_F$

$X \subseteq Y \implies Y \to X \in F^A$ by reflexivity

Given $A \in (X)^+_F \implies$ by the lemma of closure $X \to A \in F^A \implies$ by transitivity $Y \to A \in F^A$ $\implies$ by the lemma of closure $A \in (Y)^+_F$

$(X)^+_G \subseteq Z_f$

$X \subseteq Z_f \implies (X)^+_G \subseteq (Z_f)^+_G$, we have to prove that $Z_f = (Z_f)^+_G$

Let's consider $A \in S' = \{ A \in R \mid V \to W \in G \land V \subseteq Z_f \land A \in W \} \implies \exists V \to W \in G :$ $V \subseteq Z_f \land A \in W \implies \exists R_j \in \rho : VW \subseteq R_j \implies V \subseteq Z_f \cap R_j \land A \in R_j \implies A \in (Z_f \cap R_j)^+_F \cap R_j \implies A \in S_f \implies A \in Z_f$

Loseless join

PDF 15 slide 11 Let R be a relation schema. A decomposition $\rho = \{ R_1, R_2, ..., R_k \}$ of R has a lossless join if $\forall r$ legal instance of R we have that $r = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie ... \bowtie \pi_{R_k}(r)$

PDF 15 slide 13 Let R be a relation schema and let $\rho = \{ R_1, R_2, ..., R_k \}$ be a decomposition of R; for each legal instance r of R, we denote $m_\rho(r) = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie ... \bowtie \pi_{R_k}(r)$

- $r \subseteq m_\rho(r)$
- $\pi_{R_i}(m_\rho(r)) = \pi_{R_i}(r)$
- $m_\rho(m_\rho(r)) = m_\rho(r)$

Given $S_1, ..., S_k$ relation schemas with their instances $s_1, ..., s_k$, let's define the $\bowtie$ operator as

$$\mathop{\bowtie}_{i=1}^{k} S_i = \{ \bigcup_{i=1}^{k} t_j \mid \forall s_i \ \forall t_j \in s_i \ \wedge \bigcup_{i=1}^{k} t_j \text{ is a function} \}$$

$r \subseteq m_\rho(r)$

$t \in r \implies t[R_i] \in \pi_{R_i}(r)\ \forall R_i \in \rho$ by definition

$$\bowtie_{i=1}^{k} \pi_{R_i}(r) = \{\ \bigcup_{i=1}^{k} p_i[R_i] \mid p_i[R_i] \in \pi_{R_i}(r) \land \bigcup_{i=1}^{k} p_i[R_i] \text{ is a function}\ \}$$

$\forall t \in r,\ t = \bigcup_{i=1}^{k} t[R_i]$ as by definition of ρ we have that $R = \bigcup_{i=1}^{k} R_i$

$t \in r \implies t$ is a function by definition

$$t = \bigcup_{i=1}^{k} t[R_i] \in \bowtie_{i=1}^{k} \pi_{R_i}(r) = m_\rho(r) \implies t \in m_\rho(r)$$

$\pi_{R_i}(m_\rho(r)) = \pi_{R_i}(r)$

$t \in r \implies$ by definition $t[R_i] \in \pi_{R_i}(r)\ \forall R_i \in \rho$

$\pi_{R_i}(m_\rho(r)) = \{\, q[R_i] \mid q \in \Join_{i=1}^{k} \pi_{R_i}(r) \,\}$

$\pi_{R_i}(r) \subseteq \pi_{R_i}(m_\rho(r))$

$t \in r \implies t \in m_\rho(r) \implies t[R_i] \in \pi_{R_i}(m_\rho(r))$

$\pi_{R_i}(m_\rho(r)) \subseteq \pi_{R_i}(r)$

$q \in \Join_{i=1}^{k} \pi_{R_i}(r) \implies$ by definition of join $q = \Join_{i=1}^{k} \{\, p_i[R_i] \,\} \mid p_i \in r \implies$ given that q is a function $q[R_i] = p_i[R_i]$ and $p_i \in r \implies p_i[R_i] \in \pi_{R_i}(r)$ we have that $q[R_i] \in \pi_{R_i}(r)$

$$m_\rho(m_\rho(r)) = m_\rho(r)$$

$$m_\rho(m_\rho(r)) = \bowtie_{i=1}^{k} \pi_{R_i}(m_\rho(r)) = \bowtie_{i=1}^{k} \pi_{R_i}(r) = m_\rho(r)$$

Loseless join pt.2

PDF 15 slide 15 Given $\rho = \{ R_1, R_2, \ldots, R_k \}$, build a table r with $|R|$ columns and k rows. At the i-th row and j-th column put a_j if $A \in R_i$ else b_{ij}

```
def has_looseless_join(R, F, ρ):
    while !(∃ t ∈ r | ∀ A ∈ R, t[A] = a) and r changed:
        for X → Y ∈ F:
            for t1 ∈ r:
                for t2 ∈ r:
                    if t1[X] = t2[X] and t1[Y] ≠ t2[Y]:
                        for A ∈ Y:
                            if t1[A] = a:
                                t2[A] = t1[A]
                            else:
                                t1[A] = t2[A]

    return ∃ t ∈ r | ∀ A ∈ R, t[A] = a
```

Correctness

PDF 15 slide 19

Let R be a relation scheme, F a set of functional dependencies on R and let $\rho = \{ R_1, R_2, ..., R_k \}$ be a decomposition of R; the algorithm correctly decides whether ρ has a lossless join

$r = m_\rho(r) \iff r$ has a tuple with all a when the algorithm termintes

> TODO: I can prove $r = m_\rho(r) \implies r$ has a tuple with all a when the algorithm terminates, I just have to write it in LaTeX

Minimal cover

PDF 17 slide 7

Let R be a schema and F be a set of functional dependencies on R. A **minimal cover** of F is a set of functional dependencies $G \equiv F$ such that:

- $\forall X \rightarrow Y \in G, |Y| = 1$
- $\forall X \rightarrow A \in G, \nexists X' \subset X \mid G \equiv (G - \{ X \rightarrow A \}) \cup \{ X' \rightarrow A \}$
- $\nexists X \rightarrow A \in G \mid G \equiv G - \{ X \rightarrow A \}$

Minimal cover *(step 1)*

$F_1 = \{ X \to A \mid X \to Y \in F \wedge A \in Y \}$

$F \stackrel{A}{\to} F_1$ by decomposition $F_1 \stackrel{A}{\to} F_1^A \implies F \subseteq F_1^A$

$F_1 \stackrel{A}{\to} F$ by union $F \stackrel{A}{\to} F^A \implies F_1 \subseteq F^A$

$F \equiv F_1$

Minimal cover *(step 2)*

Given $X \rightarrow A \in F_1, X' \subset X \wedge X' \rightarrow A \in F_1^+ \implies F_2 = (F_1 \setminus \{ X \rightarrow A \}) \cup \{ X' \rightarrow A \}$

$X' \subseteq X \implies X \rightarrow X' \in F_1^+ \wedge X \rightarrow X' \in F_2^+$ by reflexivity

$X \rightarrow A \in F_1$

- $X \rightarrow A \in F_2 \implies X \rightarrow A \in F_2^+$
- $X \rightarrow A \notin F_2 \implies X \rightarrow X' \in F_2^+ \wedge X' \rightarrow A \in F_2^+ \implies X \rightarrow A \in F_2^+$ by transitivity

$X \rightarrow A \in F_2$

- $X \rightarrow A \in F_1 \implies X \rightarrow A \in F_1^+$
- $X \rightarrow A \notin F_1 \implies X \rightarrow A \in F_1^+$ by HP

$F_2 \equiv F_1 \implies F \equiv F_2$ by transitivity of the $\equiv$ relationship

Minimal cover *(step 3)*

$X \to A \in F_2,\ A \in (X)^+_{F_2 \setminus \{ X \to A \}} \implies F_3 = F_2 \setminus \{ X \to A \}$

$X \to A \in F_2$

- $X \to A \in F_3 \implies X \to A \in F_3^+$
- $X \to A \notin F_3 \implies X \to A \in F_3^+$ by HP as $A \in (X)^+_{F_3}$

$X \to A \in F_3$

- $X \to A \in F_2 \implies X \to A \in F_2^+$
- $X \to A \notin F_2$ is a contraddiction as $F_3 = F_2 \setminus \{ X \to A \}$ by definition

$F_2 \equiv F_3 \implies F \equiv F_3$

Decomposition

```
def decomposition(R, F: minimal cover):
    S = ∅
    ρ = ∅

    for A ∈ R | ∄ X → Y ∈ F : A ∈ XY:
        S = S ∪ {A}

    if S ≠ ∅:
        R = R - S
        ρ = ρ ∪ {S}

    if ∃ X → Y ∈ F | XY = R:
        ρ = ρ ∪ {R}
    else:
        for X → A ∈ F:
            ρ = ρ ∪ {XA}
```

Decomposition pt.2

PDF 19 slide 5

Let R be a relational schema and F a set of functional dependencies on R, which is a minimal cover; the algorithm `decomposition()` computes *(in polynomial time)* a decomposition ρ of R such that:

- each relational schema in ρ is in 3NF
- ρ preserves F