Relational Algebra Operator

Closure Property
- The set of all relations on R is closed under union, intersection, difference, and active complement

Restriction

- Rows from a relation are used to make a new relation
- Unary operator
- Result is a relation with the same set of attributes
- Represented by the sigma (σ)

- Restrictions returns a new relation containing a subset of the tuples of the original relation.
- It is sometimes called the selection operator, but this has nothing to do with SQL select statements, so Duke uses restriction

Restriction

Definition:

Let r(R) be a relation, $A \in R$, $a \in dom(A)$.

Then, $\sigma_{A=a}(r)$ is defined as

$$r'(R) = \{ t \in r \mid t(A) = a \}$$

Restriction

r

A	B	C
a_1	b_1	c_1
a_2	b_2	c_1
a_3	b_1	c_2

$\sigma_{A=a_1}(r)$

A	B	C
a_1	b_1	c_1

$\sigma_{B=b_1}(r)$

A	B	C
a_1	b_1	c_1
a_3	b_1	c_2

Restriction

Sched

Number	From	To	Departs	Arrive
84	O'Hare	JFK	3:00p	5:55p
109	JFK	LA	9:40p	2:42p
117	Atlanta	Boston	10:05p	12:43a
213	JFK	Boston	11:43a	12:45p
214	Boston	JFK	2:20p	3:12p

$\sigma_{From=JFK}(Sched)$

Number	From	To	Departs	Arrive
109	JFK	LA	9:40p	2:42p
213	JFK	Boston	11:43a	12:45p

- Above are two examples on how restriction/selection is used

Relational Algebra

Restriction (σ)

- Returns rows of the relation that satisfy the given predicate

Let *R* (A,B,C,D)

A	B	C	D
α	α	1	7
α	β	5	7
β	β	12	3
β	β	23	10

$\sigma_{A=B \wedge D>5}(R)$

A	B	C	D
α	α	1	7
β	β	23	10

Properties:

- Idempotent

  $\sigma_A(R) = \sigma_A(\sigma_A(R))$

- Commutative

  $\sigma_A(\sigma_B(R)) = \sigma_B(\sigma_A(R))$

- Breaking up the condition

  $\sigma_{A \wedge B}(R) = \sigma_A(\sigma_B(R)) = \sigma_B(\sigma_A(R))$

  $\sigma_{A \vee B}(R) = \sigma_A(R) \cup \sigma_B(R)$

Projection

- Resulting relation contains only the specified attributes, so not the same set of attributes and the relation. (However, it can return all of them.)
- Unary operator
- Result is a set of tuples, therefore no duplicates
- Represented by the pi (π)

Projection

r

A	B	C
a_1	b_1	c_1
a_2	b_2	c_1
a_3	b_1	c_2

$\Pi_{\{A\}}(r)$

A
a_1
a_2
a_3

$\Pi_{\{B,\ C\}}(r)$

B	C
b_1	c_1
b_2	c_1
b_1	c_2

$\Pi_{\{B\}}(r)$

B
b_1
b_2

College of Engineering

- Combination is unique

Projection

Sched

Number	From	To	Departs	Arrive
84	O'Hare	JFK	3:00p	5:55p
109	JFK	LA	9:40p	2:42p
117	Atlanta	Boston	10:05p	12:43a
213	JFK	Boston	11:43a	12:45p
214	Boston	JFK	2:20p	3:12p

$\Pi_{\{From\}}(Sched)$

From
O'Hare
JFK
Atlanta
Boston

Projection

Sched

Number	From	To	Departs	Arrive
84	O'Hare	JFK	3:00p	5:55p
109	JFK	LA	9:40p	2:42p
117	Atlanta	Boston	10:05p	12:43a
213	JFK	Boston	11:43a	12:45p
214	Boston	JFK	2:20p	3:12p

$\Pi_{\{Number, To\}}(Sched)$

Number	To
84	JFK
109	LA
117	Boston
213	Boston
214	JFK

Relational Algebra

Cartesian Product(×)

- Output all pairs of rows from the two input relations
- $R \times S = \{ (r, s) \mid r \in R \text{ and } s \in S \}$

A	B
α	1
β	2

r

C	D	E
α	10	a
β	10	a
β	20	b
γ	10	b

s

A	B	C	D	E
α	1	α	10	a
α	1	β	10	a
α	1	β	20	b
α	1	γ	10	b
β	2	α	10	a
β	2	β	10	a
β	2	β	20	b
β	2	γ	10	b

r × s

Properties:

- NOT commutative
- NOT associative

$$(R \cap S) \times (W \cap Z) = (R \cap W) \times (S \cap Z)$$
$$(R \cup S) \times (W \cup Z) \neq (R \cup W) \times (S \cup Z)$$
$$R \times (S \cap W) = (R \times S) \cap (R \times S)$$
$$R \times (S \cup W) = (R \times S) \cup (R \times S)$$

Relational Algebra

Cartesian Product(×)

- Output all pairs of rows from the two input relations
- $R \times S = \{ (r, s) \mid r \in R \text{ and } s \in S \}$

A	B
α	1
β	2

r

C	D	E
α	10	a
β	10	a
β	20	b
γ	10	b

s

A	B	C	D	E
α	1	α	10	a
α	1	β	10	a
α	1	β	20	b
α	1	γ	10	b
β	2	α	10	a
β	2	β	10	a
β	2	β	20	b
β	2	γ	10	b

r × s

Properties:

- NOT commutative
- NOT associative

$$(R \cap S) \times (W \cap Z) = (R \cap W) \times (S \cap Z)$$
$$(R \cup S) \times (W \cup Z) \neq (R \cup W) \times (S \cup Z)$$
$$R \times (S \cap W) = (R \times S) \cap (R \times S)$$
$$R \times (S \cup W) = (R \times S) \cup (R \times S)$$

Relational Algebra

Cartesian Product(×)

- Output all pairs of rows from the two input relations
- $R \times S = \{ (r, s) \mid r \in R \text{ and } s \in S \}$

A	B
α	1
β	2

r

C	D	E
α	10	a
β	10	a
β	20	b
γ	10	b

s

A	B	C	D	E
α	1	α	10	a
α	1	β	10	a
α	1	β	20	b
α	1	γ	10	b
β	2	α	10	a
β	2	β	10	a
β	2	β	20	b
β	2	γ	10	b

r × s

Properties:

- NOT commutative
- NOT associative

$$(R \cap S) \times (W \cap Z) = (R \cap W) \times (S \cap Z)$$
$$(R \cup S) \times (W \cup Z) \neq (R \cup W) \times (S \cup Z)$$
$$R \times (S \cap W) = (R \times S) \cap (R \times S)$$
$$R \times (S \cup W) = (R \times S) \cup (R \times S)$$

Relational Algebra

Natural join (⋈)

- Let r and s be relations on schemas R and S, respectively
- Natural join of relations R and S is a relation on schema $R\ S$ created by the following:
  - Consider each pair of tuples t_r from r and t_s from s.
  - If t_r and t_s have the same value on each of the attributes in $R \cap S$, add a tuple t to the result, where

    t has the same value as t_r on r

    t has the same value as t_s on s
- Properties: associative and commutative

A	B	C	D
α	1	α	a
β	2	γ	a
γ	4	β	b
α	1	γ	a
δ	2	β	b

r

B	D	E
1	a	α
3	a	β
1	a	γ
2	b	δ
3	b	ε

s

A	B	C	D	E
α	1	α	a	α
α	1	α	a	γ
α	1	γ	a	α
α	1	γ	a	γ
δ	2	β	b	δ

$r \bowtie s$

Relational Algebra

Theta-join (θ)

- $R \bowtie_\theta S = \sigma_\theta (R \times S)$
- Selection σ meeting condition θ after cross product

R

A	B
3	2
2	5

S

C	D
2	1
4	3
4	7

$R \bowtie_{A<D} S$

A	B	C	D
3	2	4	7
2	5	4	3
2	5	4	7

Relational Algebra

Outer join

- An extension of the join operation that avoids loss of information
- Computes the join and then adds tuples from one relation that does not match tuples in the other relation
- Uses null values
  - null signifies unknown or does not exist
  - All comparisons involving null are false by definition
  - The result of any arithmetic expression involving null is null

Relational Algebra

Instructor

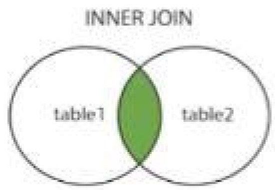
ID	name	dept_name
10101	Srinivasan	Comp. Sci.
12121	Wu	Finance
15151	Mozart	Music

Teaches

ID	course_id
10101	CS-101
12121	FIN-201
76766	BIO-101

nner join

- Instructor ⋈ Teaches



ID	name	dept_name	course_id
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201

Relational Algebra

Instructor

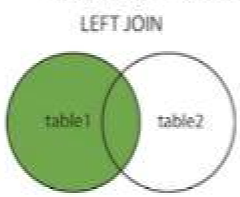
ID	name	dept_name
10101	Srinivasan	Comp. Sci.
12121	Wu	Finance
15151	Mozart	Music

Teaches

ID	course_id
10101	CS-101
12121	FIN-201
76766	BIO-101

Left outer join

- Instructor ⟕ Teaches



ID	name	dept_name	course_id
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201
15151	Mozart	Music	null

Relational Algebra

Instructor

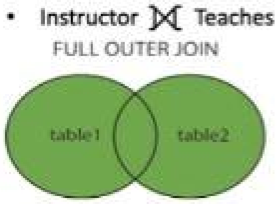
ID	name	dept_name
10101	Srinivasan	Comp. Sci.
12121	Wu	Finance
15151	Mozart	Music

Teaches

ID	course_id
10101	CS-101
12121	FIN-201
76766	BIO-101

Full outer join

- Instructor ⟗ Teaches



ID	name	dept_name	course_id
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201
15151	Mozart	Music	null
76766	null	null	BIO-101

Relational Algebra

Division

- Let *r* and *s* be relations on schemas *R* and *S* respectively and S ⊆ R
- R / S is a relation on schema R – S
- A tuple *t* is in R/S if satisfies both:
  - *t* is in $\prod_{R-S}(r)$
  - For every tuple t_s in *s*, there is a tuple t_r in *r* satisfying both:
    - $t_r[S] = t_s[S]$
    - $t_r[R-S] = t$

>> The resulting relation

- Here's a division operator
- Note that the set of attributes of relation S must be a subset of the attributes of a relation R
- The resulting relation consists of the attributes in R that are not in S division
- R divided by S is the set of all tuples that are, that have the same values for the attributes.

Relational Algebra

Division

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

pno
p2
p4

B2

pno
p1
p2
p4

B3

sno
s1
s2
s3
s4

A/B1

Relational Algebra

Division

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

pno
p2
p4

B2

pno
p1
p2
p4

B3

sno
s1
s4

A/B2

- In this case, we're only going to include the tuples that have values for both in B2.
- So we have s1 has a p2 and p4, so we'll include S1 and S4. There's a p2 and p4.

Relational Algebra

Division

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

pno
p2
p4

B2

pno
p1
p2
p4

B3

sno
s1

A/B3

Summary of Relational Algebra Operators

Symbol (Name)	Example of Use
(Selection) σ	salary > = 85000 (instructor)
	Return rows of the input relation that satisfy the predicate.
(Projection) Π	$\Pi_{ID,\ salary}(instructor)$
	Output specified attributes from all rows of the input relation. Remove duplicate tuples from the output.
(Cartesian Product) x	*instructor* x *department*
	Output pairs of rows from the two input relations that have the same value on all attributes that have the same name.
(Union) ∪	$\Pi_{name}(instructor) \cup \Pi_{name}(student)$
	Output the union of tuples from the *two* input relations.
(Set Difference) −	$\Pi_{name}(instructor)$ -- $\Pi_{name}(student)$
	Output the set difference of tuples from the two input relations.
(Natural Join) ⋈	*instructor* ⋈ *department*
	Output pairs of rows from the two input relations that have the same value on all attributes that have the same name.

- 2

Review Terms

– Restrict (aka select) (σ)
– Project (**π**)
– Union (∪)
– Set intersection (∩)
– Set difference (–)
– Cartesian product (×)
– Rename (ρ)

– Natural join
– Outer join
– Left outer join
– Right outer join
– Full outer join