# COMP 543: Tools & Models for Data Science

## Relational Calculus

Chris Jermaine & Risa Myers

Rice University

# Relational Calculus

- Nothing more than a First Order Logic predicate...
- Embedded within a set constructor

# Example: Cold Brew Drinkers

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

? Query: Who goes to a cafe serving Cold Brew?

# Example: Cold Brew Drinkers

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

- Query: Who goes to a cafe serving Cold Brew?

$$\{f.\mathrm{DRINKER} \mid \mathrm{FREQUENTS}(f) \wedge \exists(s)(\mathrm{SERVES}(s) \wedge s.\mathrm{COFFEE} = \text{'Cold Brew'} \wedge s.\mathrm{CAFE} = f.\mathrm{CAFE})\}$$

# Example: Cold Brew Haters

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

? Query: Who has not gone to a cafe serving Cold Brew?

# Example:Cold Brew Haters, Common Approach

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

- Query: Who has not gone to a cafe serving Cold Brew?

$$\{f.\text{DRINKER} \mid \text{FREQUENTS}(f) \wedge \neg\exists(s)(\text{SERVES}(s) \\ \wedge\ s.\text{COFFEE} = \text{'Cold Brew'} \wedge\ s.\text{CAFE} = f.\text{CAFE})\}$$

# Example: Cold Brew Haters, Common Approach

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

- Query: Who has not gone to a cafe serving Cold Brew?

$$\{f.\text{DRINKER} \mid \text{FREQUENTS}(f) \wedge \neg\exists(s)(\text{SERVES}(s) \wedge s.\text{COFFEE} = \text{'Cold Brew'} \wedge s.\text{CAFE} = f.\text{CAFE})\}$$

Wrong! This gives us “Who has gone to a cafe that does not serve Cold Brew”
In parenthesis we have cafes that serve ‘Cold Brew’ that someone has visited.
Negating it, and checking for existence, gives us cafes that someone has visited that do NOT serve Cold Brew.

# Walk–through Data

FREQUENTS

| DRINKER | CAFE |
|---|---|
| Chris | A |
| Chris | B |
| Chris | C |
| Risa | A |
| Risa | B |

SERVES

| CAFE | COFFEE |
|---|---|
| A | Cold Brew |
| A | Drip |
| A | Espresso |
| B | Espresso |
| C | Drip |

# Original Approach

Option A: Who has gone to a cafe that does not serve ‘Cold Brew’?

$$\{f.\text{DRINKER} \mid \text{FREQUENTS}(f) \wedge \neg\exists(s)(\text{SERVES}(s) \\ \wedge\ s.\text{COFFEE} = \text{'Cold Brew'} \wedge\ s.\text{CAFE} = f.\text{CAFE})\}$$

1. Do the cross product between FREQUENTS and SERVES
2. Restrict ourselves to the case where s.CAFE = f.CAFE
3. Does the statement evaluate to TRUE?
   - If Yes: include f.DRINKER in the result set

| DRINKER | CAFE | CAFE | COFFEE | Result |
|---|---|---|---|---|
| Risa | B | B | Drip | T |
| Risa | A | A | Cold Brew | F |
| Risa | A | A | Drip | |
| Risa | A | A | Espresso | |

- When f.CAFE = ‘B’, Risa gets included in the result set.
- However, based on the information when f.CAFE = ‘A’, Risa should NOT be in the result set
- Issue: We want to look at all the relevant rows together

# New Approach

Option B: Who has not gone to a cafe serving 'Cold Brew'?

$$\{f.\text{DRINKER} \mid \text{FREQUENTS}(f) \wedge \neg\exists(f_2, s)(FREQ(f_2) \wedge SERVES(s) \wedge f_2.CAFE = s.CAFE \wedge s.COFFEE = \text{'Cold Brew'} \wedge f.DRINKER = f_2.DRINKER)\}$$

| DRINKER |
|---|
| Risa |

| DRINKER | CAFE | COFFEE | Result |
|---|---|---|---|
| Risa | A | Cold Brew | F |
| Risa | A | Drip | |
| Risa | A | Espresso | |
| Risa | B | Drip | |

- Here, we have another variable, $f_2$
- We consider each drinker in turn. Let's look at Risa
- Now, look at all the combinations of FREQUENTS and SERVES where the CAFE matches
- If there is any row where the Coffee is 'Cold Brew', we exclude the drinker
- Now, in this case, one of the cafes that Risa frequents does serve Cold Brew, so Risa is not added to the result set

# Example: People Who Like to Drink Coffee

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

- ? Query: Who goes to a cafe that serves a coffee they like?

# Example: People Who Like to Drink Coffee

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

- Query: Who goes to a cafe that serves a coffee they like?

$$
\begin{aligned}
\{f.\mathrm{DRINKER} \mid \mathrm{FREQUENTS}(f) \wedge \exists(s,l)(\mathrm{SERVES}(s) \wedge \mathrm{LIKES}(l) \\
\quad \wedge\ s.\mathrm{COFFEE} = l.\mathrm{COFFEE} \\
\quad \wedge\ s.\mathrm{CAFE} = f.\mathrm{CAFE} \\
\quad \wedge\ l.\mathrm{DRINKER} = f.\mathrm{DRINKER})\}
\end{aligned}
$$

? We didn't refer to any table more than once. Why not?

# Example: People Who Like to Drink Coffee

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

- Query: Who goes to a cafe that serves a coffee they like?

$$\{f.\text{DRINKER} \mid \text{FREQUENTS}(f) \wedge \exists(s,l)(\text{SERVES}(s) \wedge \text{LIKES}(l)$$
$$\wedge\ s.\text{COFFEE} = l.\text{COFFEE}$$
$$\wedge\ s.\text{CAFE} = f.\text{CAFE}$$
$$\wedge\ l.\text{DRINKER} = f.\text{DRINKER})\}$$

- We didn't refer to any table more than once. Why not?
- It wasn't needed since we didn't have any 'Always' or 'Never' predicates
- We were looking for 'Any'

# Example: People Who Avoid Bad Cafes

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

? Query: Which people only go to cafes that serve a coffee they like?

# Example: People Who Avoid Bad Cafes

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

? Query: Which people only go to cafes that serve a coffee they like?

- Two possible approaches: using $\forall$ or using $\exists$

? Which is easier to reason about?

# Example: People Who Avoid Bad Cafes

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

- Query: Which people only go to cafes that serve a coffee they like?

$$\{f.\text{DRINKER} \mid \text{FREQUENTS}(f) \wedge \forall (f_2)(\text{if } f_2 \text{ tells us a cafe that } f.\text{DRINKER goes to then that cafe needs to serve a coffee that } f.\text{DRINKER likes})\}$$

# Example: People Who Avoid Bad Cafes

LIKES (DRINKER, COFFEE)
FREQUENTS (DRINKER, CAFE)
SERVES (CAFE, COFFEE)

- Query: Which people only go to cafes that serve a coffee they like?

$$\begin{aligned}
&\{f.\text{DRINKER} \mid \text{FREQUENTS}(f) \wedge \forall(f_2)(\text{FREQUENTS}(f_2) \\
&\quad \wedge f.\text{DRINKER} = f_2.\text{DRINKER} \rightarrow \exists(s,l)(\text{SERVES}(s) \\
&\quad \wedge \text{LIKES}(l) \wedge\ s.\text{CAFE} = f_2.\text{CAFE} \wedge\ l.\text{COFFEE} = s.\text{COFFEE} \\
&\quad \wedge\ l.\text{DRINKER} = f_2.\text{DRINKER}))\}
\end{aligned}$$

# Implication

$$
\begin{aligned}
\{f.\text{DRINKER} \mid\ & \text{FREQUENTS}(f) \wedge \forall(f_2)(\text{FREQUENTS}(f_2) \\
& \wedge f.\text{DRINKER} = f_2.\text{DRINKER} \rightarrow \exists(s,l)(\text{SERVES}(s) \\
& \wedge \text{LIKES}(l) \wedge\ s.\text{CAFE} = f_2.\text{CAFE} \wedge\ l.\text{COFFEE} = s.\text{COFFEE} \\
& \wedge\ l.\text{DRINKER} = f_2.\text{DRINKER}))\}
\end{aligned}
$$

? Note: we invariably have a "$\rightarrow$" within a $\forall$ quantifier. Why?

# Implication

$$\{f.\text{DRINKER} \mid \text{FREQUENTS}(f) \wedge \forall(f_2)(\text{FREQUENTS}(f_2)$$
$$\wedge f.\text{DRINKER} = f_2.\text{DRINKER} \rightarrow \exists(s,l)(\text{SERVES}(s)$$
$$\wedge \text{LIKES}(l) \wedge s.\text{CAFE} = f_2.\text{CAFE} \wedge l.\text{COFFEE} = s.\text{COFFEE}$$
$$\wedge l.\text{DRINKER} = f_2.\text{DRINKER}))\}$$

- Note: we invariably have a "$\rightarrow$" within a $\forall$ quantifier. Why?
  - $\rightarrow$ is a logical IF–THEN statement

# Questions?