CMPT 321
FALL 2017

# Using SQL for predictions and recommendations

Lecture 04.03

*by Marina Barsky*

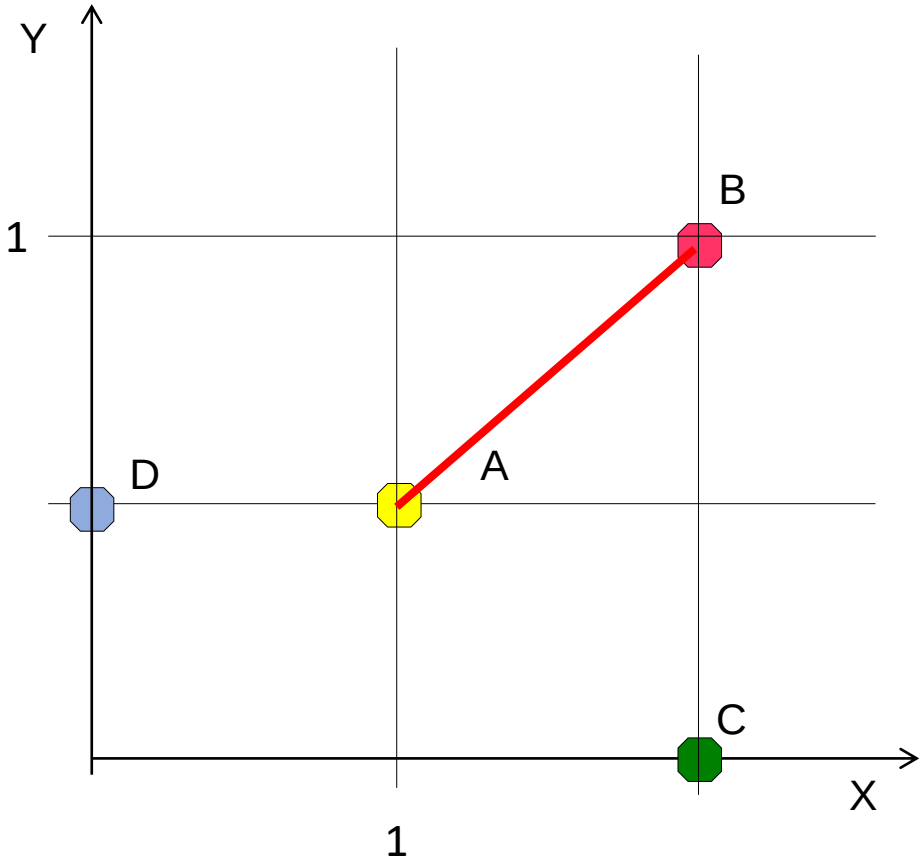
# Look-alike model for prediction

# What nearest neighbors can tell about you

To predict salary level for person X:

- Find k people most similar to X by their demographic characteristics
- Return average salary for these nearest neighbors

# Simple distance function for numeric attributes: Euclidean distance




$d(A,B)=?$

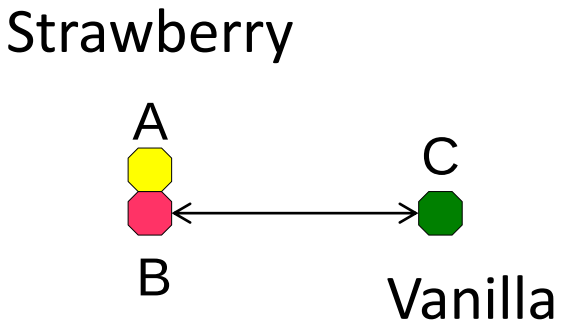
$d\ (B,C)=?$

$d(C,D)=?$

$$d(A,B) = \sqrt{|\ A_X - B_X\ |^2 + |\ A_Y - B_Y\ |^2}$$

# Distance between categorical attributes

- D(A,B)=0
- D(B,C)=1
- D(A,C)=1



Finding 35 nearest neighbors for person:

age: 55

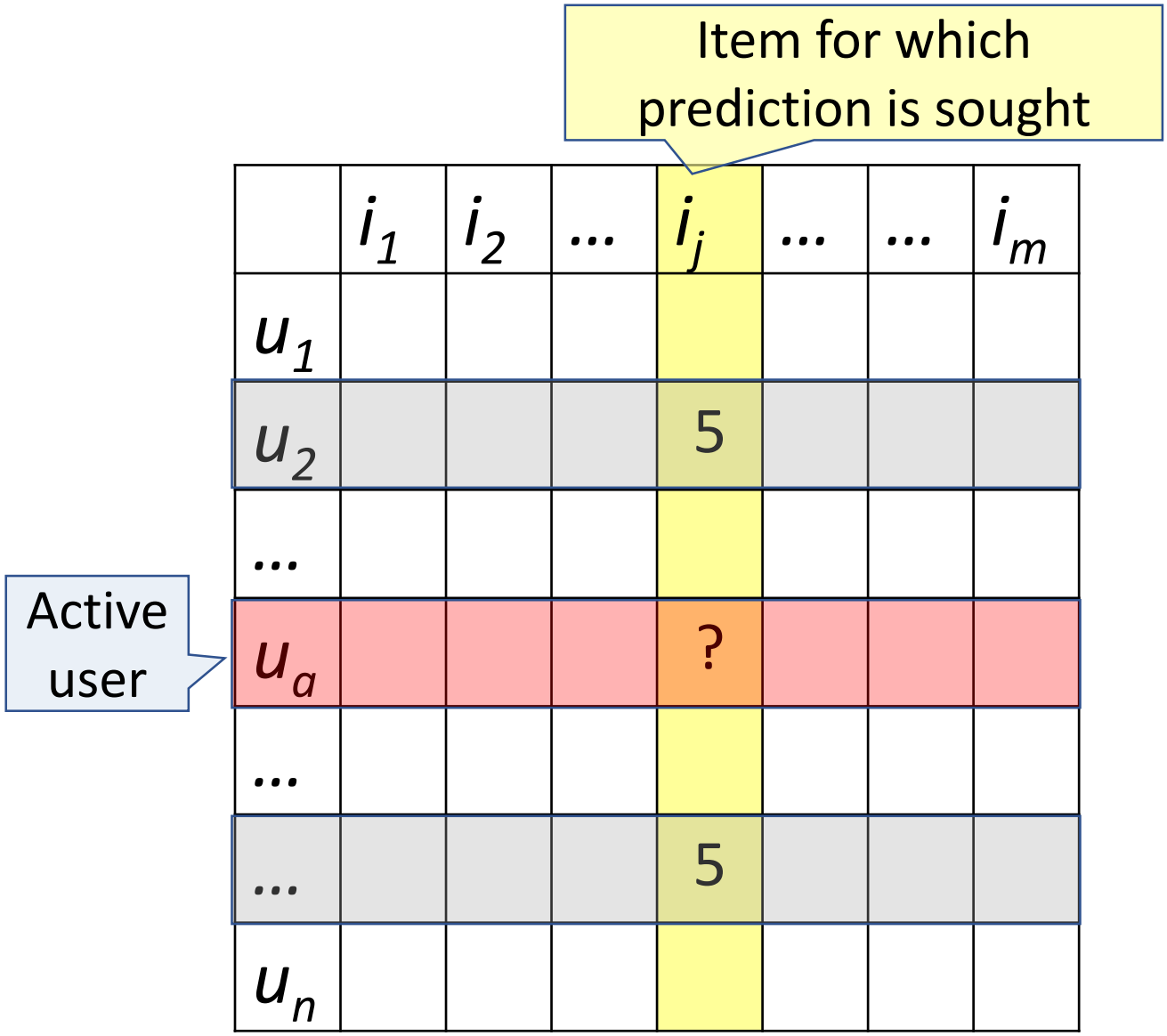
working hours per week: 48

educations numeric: 17(bachelor)

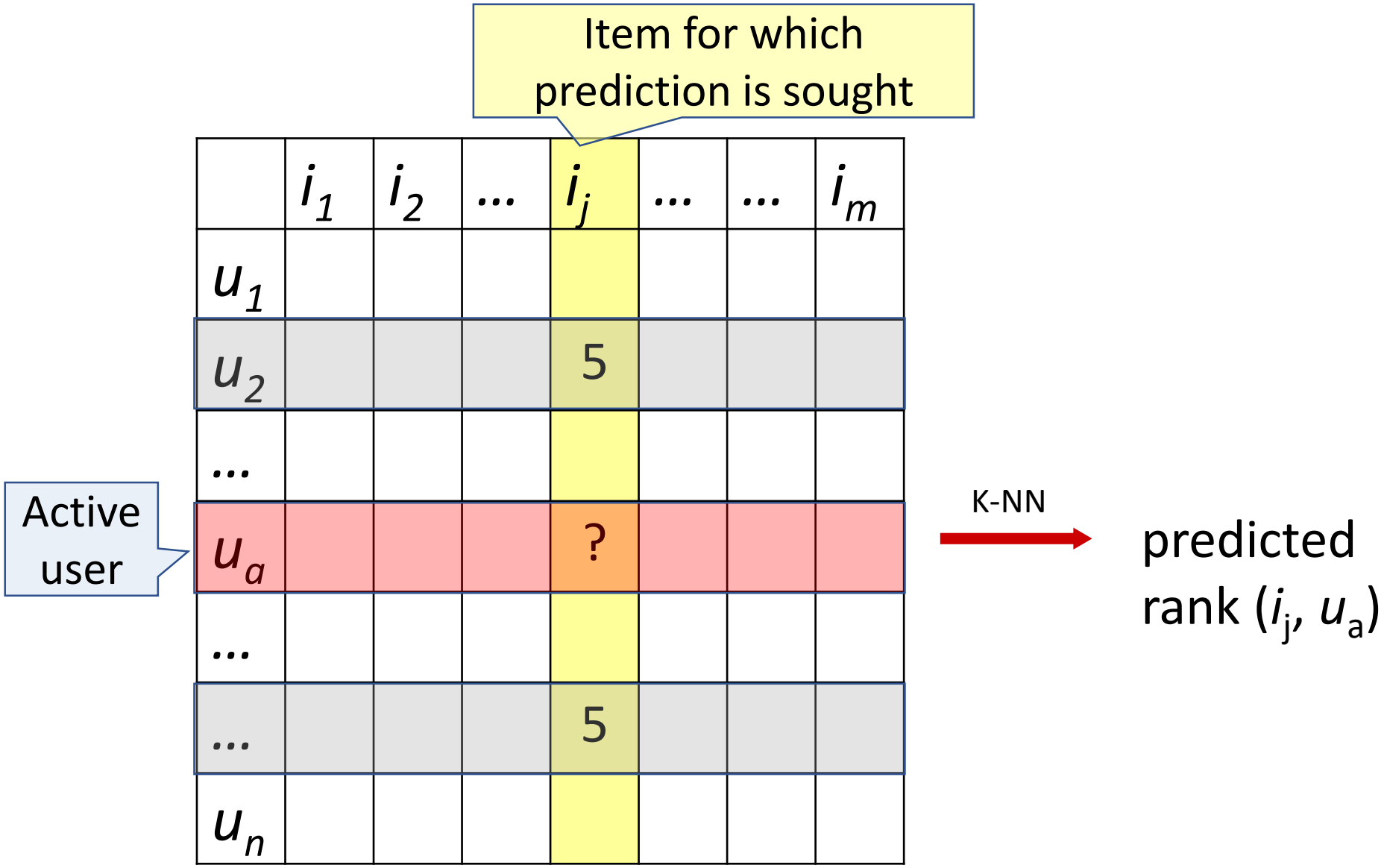
maritalstatus: married

```
SELECT average(salary) FROM
  (SELECT salary,
       ((age - 55)*(age - 55)
       + (workinhoursperweek - 48)*(workinhoursperweek - 48)
       +(educationnumeric - 17)*(educationnumeric - 17)
       +(CASE
         WHEN maritalstatus='Married-civ-spouse' THEN 0 ELSE 1
       END)) distance_column
  FROM person
  ORDER BY 2
  LIMIT 35) neighbors;
```

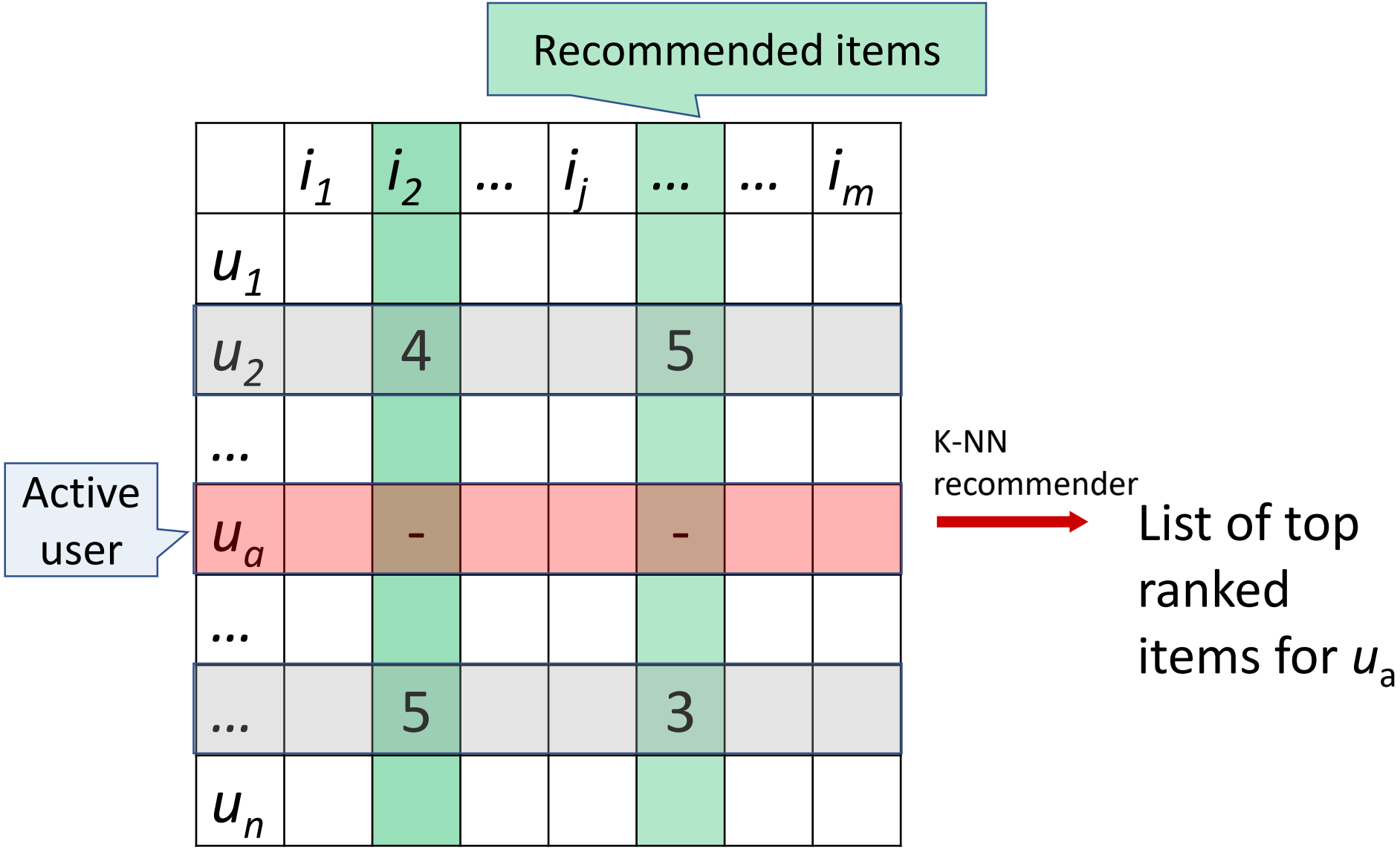
# K-NN prediction



# K-NN prediction



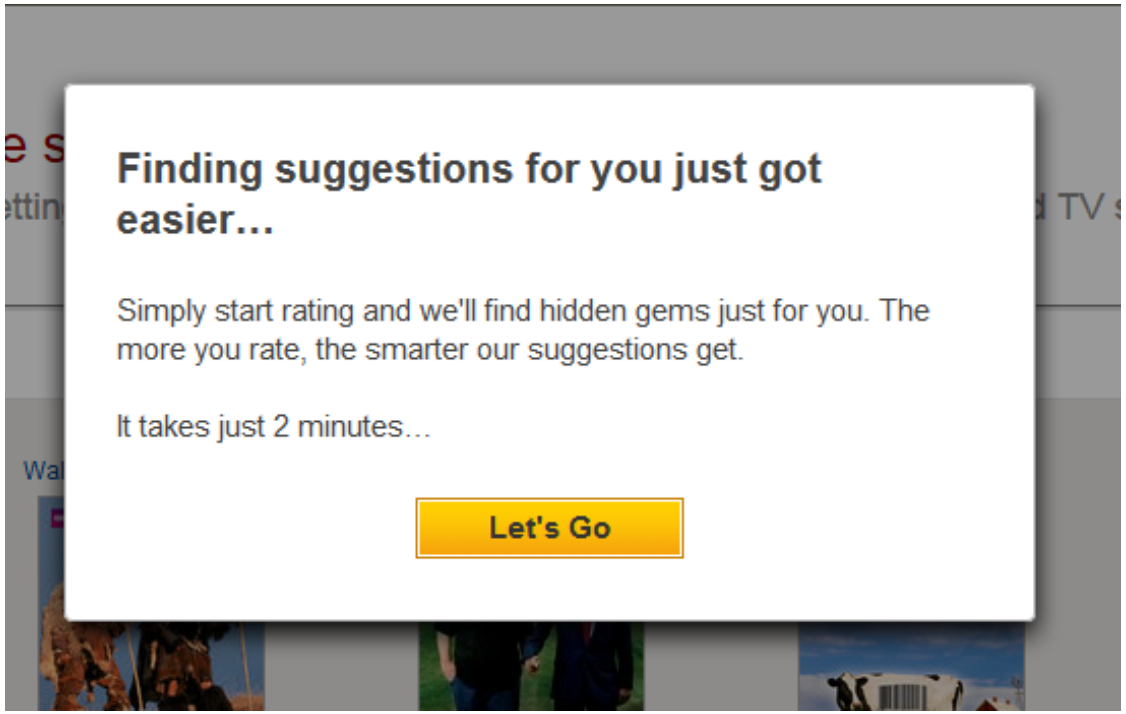
# K-NN recommender (collaborative filtering)



# Automated recommender system (collaborative filtering)

I. Build profile for active user
II. Compare active profile with the profiles of other customers, locate similar “neighbors”
III. Find and combine ratings of a peer group selected by similar tastes for the items that active user did not rank yet
IV. Rank predictions and output top-scored ones

# Creating customer profile




Netflix.ca

# Example: music recommender

You:

{Lady Gaga, Katy Perry, Justin Bieber, Maroon 5}

# II. Finding users with similar tastes

1. April:
   {The Beatles, Lady Antebellum}

2. Ben:
   {Lady Gaga, Adele, Kelly Clarkson, The Dixie Chicks, Lady Antebellum}

3. Cory:
   {Kelly Clarkson, Lady Gaga, Katy Perry, Justin Bieber, Lady Antebellum}

4. Dave:
   {The Beatles, Maroon 5, Lady Antebellum}

5. Edgar:
   {Adele, Maroon 5, Katy Perry, Bruno Mars}

# II. Finding users with similar tastes

| | Adele | The Beatles | Justin Bieber | The Dixie Chicks | Kelly Clarkson | Lady Gaga | Lady Antebellum | Maroon 5 | Bruno Mars | Katy Perry |
|---|---|---|---|---|---|---|---|---|---|---|
| April | | Yes | | | | | Yes | | | |
| Ben | Yes | | | Yes | Yes | Yes | Yes | | | |
| Cory | | | Yes | | Yes | Yes | Yes | | | Yes |
| Dave | | Yes | | | | | Yes | Yes | | |
| Edgar | Yes | | | | | | | Yes | Yes | Yes |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| You | | | **Yes** | | | **Yes** | | **Yes** | | **Yes** |

**Similarity measures for asymmetric binary data**

**Simple matching coefficient:**

number of matches / total attributes

**Jaccard index:**

number of matches/total not-both-null attributes

# II. Finding users with similar tastes (Jaccard index)

| | Adele | The Beatles | Justin Bieber | The Dixie Chicks | Kelly Clarkson | Lady Gaga | Lady Antebellum | Maroon 5 | Bruno Mars | Katy Perry |
|---|---|---|---|---|---|---|---|---|---|---|
| April | | Yes | | | | | Yes | | | |
| Ben | Yes | | | Yes | Yes | Yes | Yes | | | |
| Cory | | | Yes | | Yes | Yes | Yes | | | Yes |
| Dave | | Yes | | | | | Yes | Yes | | |
| Edgar | Yes | | | | | | | Yes | Yes | Yes |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| You | | | **Yes** | | | **Yes** | | **Yes** | | **Yes** |

You vs April: 0/6
You vs Ben: 1/8
You vs. Cory: 3/6
You vs. Dave: 1/6
You vs. Edgar: 2/6

**Your peer group:**

Cory: similarity 0.50
Dave: similarity 0.17
Edgar: similarity 0.33

# III. Combine ratings for new items (weighted voting)

| | Adele | The Beatles | Justin Bieber | The Dixie Chicks | Kelly Clarkson | Lady Gaga | Lady Antebellum | Maroon 5 | Bruno Mars | Katy Perry |
|---|---|---|---|---|---|---|---|---|---|---|
| April | | Yes | | | | | Yes | | | |
| Ben | Yes | | | Yes | Yes | Yes | Yes | | | |
| Cory | | | Yes | | Yes | Yes | Yes | | | Yes |
| Dave | | Yes | | | | | Yes | Yes | | |
| Edgar | Yes | | | | | | | Yes | Yes | Yes |
| You | | | **Yes** | | | **Yes** | | **Yes** | | **Yes** |

## Your peer group:

Cory: similarity 0.50

Dave: similarity 0.17

Edgar: similarity 0.33

## Predicted likes for new items:

Adele: 1 like *0.33 = 0.33

The Beatles: 1 like * 0.17 = 0.17

Kelly Clarkson: 1 like * 0.50 = 0.50

Lady Antebellum: 1 like *0.50 + 1 like * 0.17 = 0.67

Bruno Mars: 1 like * 0.33 = 0.33

# IV. Output top-ranked

| | Adele | The Beatles | Justin Bieber | The Dixie Chicks | Kelly Clarkson | Lady Gaga | Lady Antebellum | Maroon 5 | Bruno Mars | Katy Perry |
|---|---|---|---|---|---|---|---|---|---|---|
| April | | Yes | | | | | Yes | | | |
| Ben | Yes | | | Yes | Yes | Yes | Yes | | | |
| Cory | | | Yes | | Yes | Yes | Yes | | | Yes |
| Dave | | Yes | | | | | | Yes | | |
| Edgar | Yes | | | | | | | Yes | Yes | Yes |
| You | | | **Yes** | | | **Yes** | | **Yes** | | **Yes** |

**Your peer group:**

Cory: similarity 0.50

Dave: similarity 0.17

Edgar: similarity 0.33

**Top ranked:**

Lady Antebellum: 0.67

Kelly Clarkson: 0.50

These are your recommendations !

# Possible project: Course recommender

- Task:
  - Develop course recommender system using databases and SQL queries
- Challenges:
  - Distance metrics (who are most similar users)
  - How to build a student profile?
  - What are the best courses to recommend:
    - Courses where subject is most interesting?
    - Courses where grades are the highest?
    - Courses by instructor?