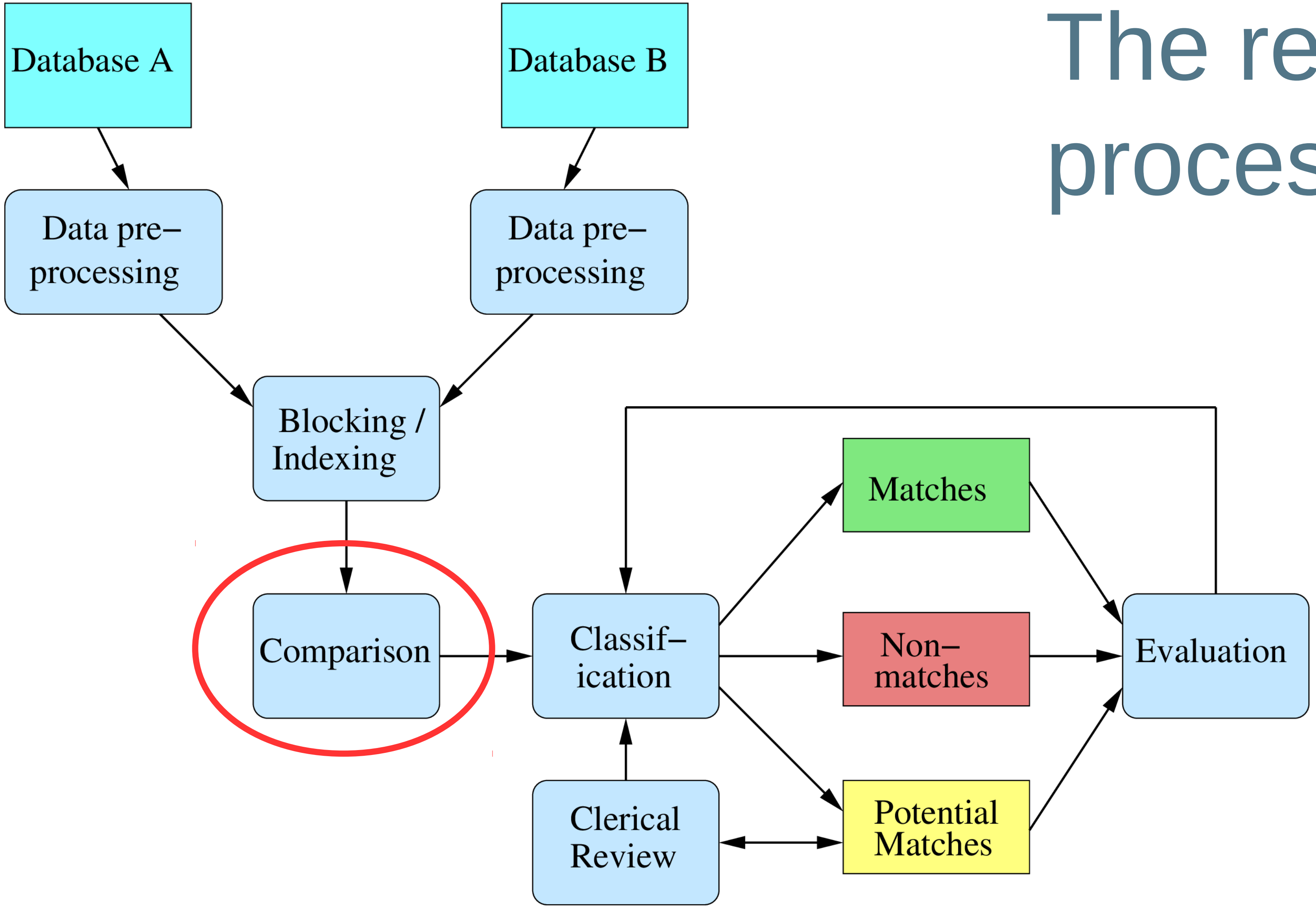
COMP3430 / COMP8430
Data wrangling

Lecture 15: Record pair comparison (1)
(Lecturer: Peter Christen)

Lecture outline

- Comparing records for record linkage
- Similarity and distance functions
- Basic comparison functions
- Numerical and other comparison functions

The record linkage process



Comparing record pairs (1)

- The blocking process generates groups / clusters of records
- From each block, record pairs are generated
  - For a linkage of two databases, all records from database A are paired with all records from database B from blocks that have the same BKV
  - For the deduplication of a single database, all unique record pairs are formed from each block (a record is not compared with itself)
- Record pairs are compared based on their common available attributes (fields)
  - These are commonly names, addresses, dates, phone numbers, etc.
  - They contain variations and errors (even after cleaning), or can be missing or out-of-date

Comparing record pairs (2)

- Exact comparison of attribute values will not provide good linkage results
  - Even true matching record pairs often contain different attribute values
  - For example:
    ['peter', 'paul', 'meier', '2/21 main st', 'acton', 'act', ' 2601']
    ['peter', 'p', 'meyer', '21 main street', 'acton', 'act', ' 2602']

- Approximate comparison functions are required
  - To calculate similarities between attribute values, not only 'is the same or is different'
  - They need to be appropriate for the content of a certain attribute (text: names, addresses, dates, phone numbers; numerical: ages, salaries)

Similarities and distances (1)

- Approximate matching functions generally calculate a numerical similarity value
  - *sim = 0*: Two values are completely different ('peter' and 'david')
  - *sim = 1*: Two values are exactly the same ('peter and peter')
  - *0 < sim < 1*: Two values are *somewhat* similar ('peter' and 'pedro')
- For the same pair of values, different functions calculate different similarities
- Some functions calculate distances
- Distances can be converted into similarities
  - *sim = 1/dist*, with *sim = 1* if *dist = 0*
  - *dist = 1-sim*, if *0 ≤ dist ≤1*

Similarities and distances (2)

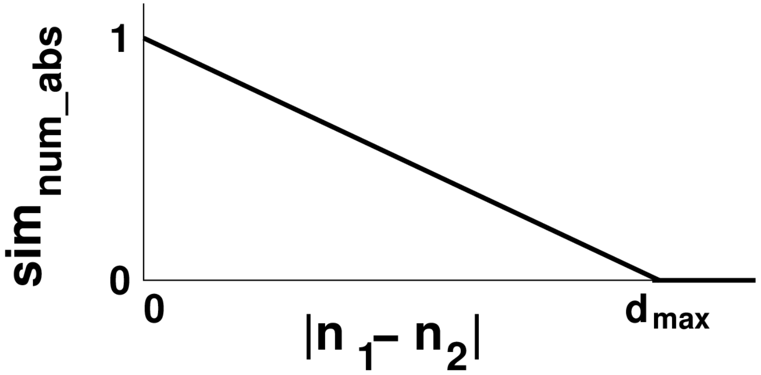
- Distance functions (or distance metrics) have several properties:
  - *dist(val1, val1) = 0* Distance from an object to itself is always 0
  - *dist(val1, val2) ≥ 0* Distances are always positive
  - *dist(val1, val2) = dist(val2, val1)* Distances are symmetric
  - *dist(val1, val2) + dist(val1, val3) ≥ dist(val2, val3)* Triangular inequality must hold
- Not all approximate matching functions are proper metric distances
  - Triangular inequality does not hold for certain functions
  - Some functions are not symmetric (for example, those that calculate if one attribute value is included in another, like 'pete' and 'peter')

Basic comparison functions

- **Exact** comparison:
  - $s_{exact}(val1, val2) = 1$ if $val1 = val2$, 0 otherwise
  - *val1* and *val2* can be strings, numbers, etc.
- **Truncate** string comparison (*x* characters the same at beginning)
  - For string values only
  - $s_{trunc}(val1, val2) = 1$ if $val1[:x] = val2[:x]$, 0 otherwise
  - Similar for testing if the end is the same
- **Phonetic** encoded comparisons
  - For strings only
  - $s_{encode}(val1, val2) = 1$ if encode($val1) = encode(val2)$, 0 otherwise
  - *encode()* is a phonetic encoding function as discussed previously

Numerical comparison functions (1)

- For numerical values, we also want to have a comparison that calculates a similarity between 0 and 1
- We set a *maximum absolute difference* allowed, or a *maximum percentage difference* allowed
  – If two values differ more their similarity will be 0
- For absolute maximum difference of d_{max} and two values n_1 and n_2:
  – If $abs(n_1 – n_2) \geq d_{max}$: $sim_{num\_abs} = 0$
  – If $abs(n_1 – n_2) < d_{max}$: $sim_{num\_abs} = 1 - (abs(n_1 – n_2) / d_{max})$

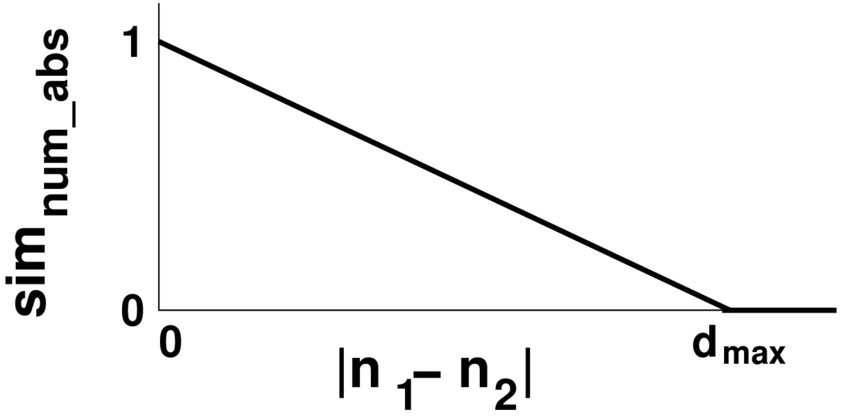


Numerical comparison functions (2)

- Similar for maximum percentage difference
  - Similarity for income (salary) differences of maximum 5% is more suitable compared to a maximum difference of $10,000
  - Similarity of age difference by 10% is better than maximum age difference of 5 years (young compared to old people)

- **Question**: *Calculate similarities for absolute maximum difference of* $d_{max} = 5$, $n_1 = 42$ *and* $n_2 = \{37, 38, 40, 41, 49\}$

Then calculate percentage differences assuming these are ages



Date and time comparison

- Dates are often used when records are compared
  - Comparing dates as strings is not a good idea: 31/12/1999 versus 01/01/2000, 24/11/2017 versus 24/01/2017
  - Dates can be converted into number by counting the number of days since a certain fix date, then calculate numerical similarity between day numbers
- Specific issue with how dates are recorded
  - Sometimes day and month numbers are swapped: US versus (almost) the rest of the world, for example 12/07/2000 versus 07/12/2000
- Time values also are modulo

  23:59 is more similar to 00:01 than to 13:59