Australian National University

COMP3430 / COMP8430
Data wrangling

Lab 3: Blocking for Record Linkage

Objectives of this lab

- Today's lab is the first in a series of five labs during which we will gradually build a complete record linkage system.
- We will provide you with basic Python skeleton modules and over the next few labs you will be asked to complete the different components of the modules.
- Completion of the blocking module of the overall system.

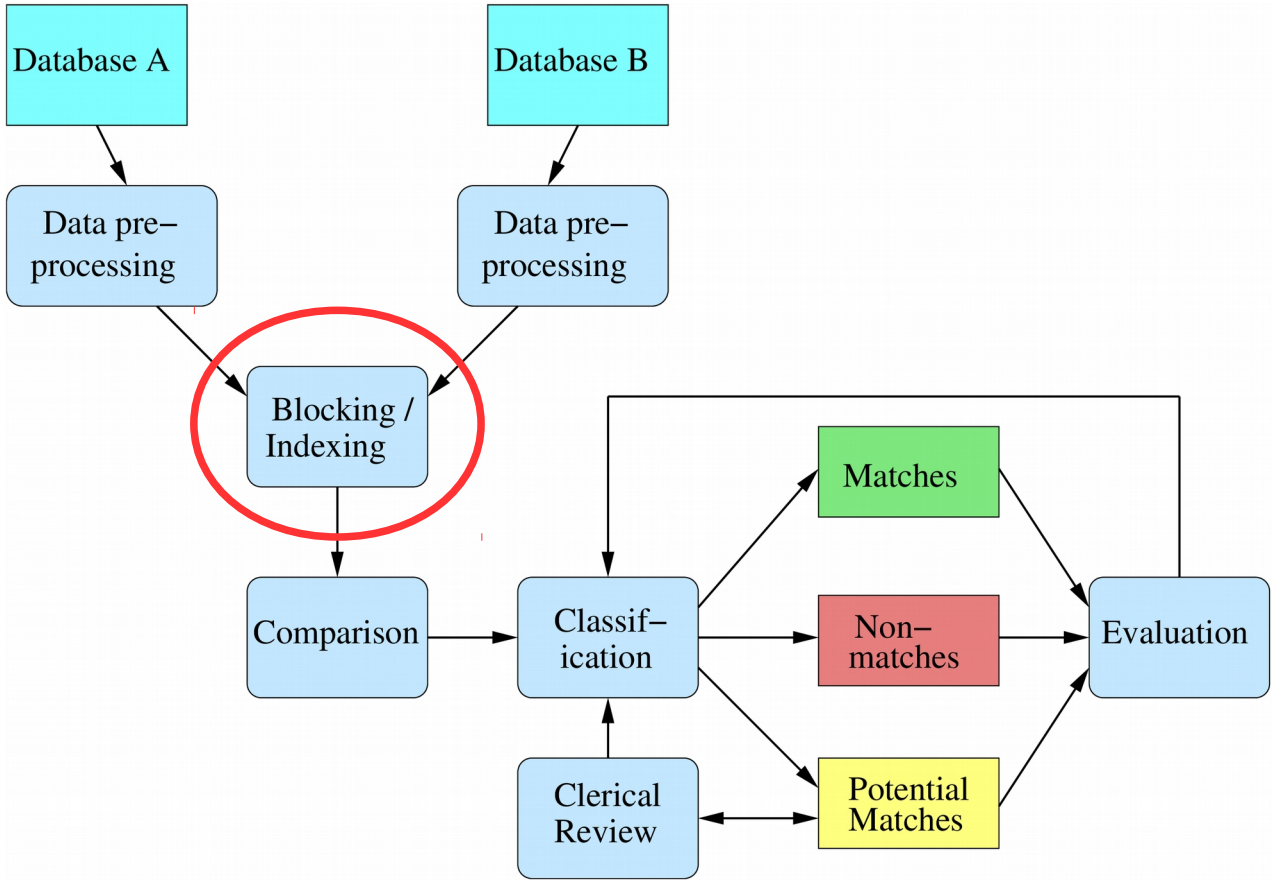
Outline of this lab

- Understand how the record linkage (RL) program works
- Explore and implement different blocking techniques
- Evaluate blocking performance
- Summary

Preliminaries

- **Before you begin, aim to review lectures 13 and 14 if you have not already viewed them.**
- For the record linkage (RL) program we provide you with a set of basic Python skeleton modules and set of data sets.
- Download the **comp3430_comp8430_reclink-lab-3-6.zip** archive from Wattle.
- Create a folder for the program in your machine and extract the code and data sets there.

What is blocking?



- This week we focus on implementing different bocking techniques.
- What is the main aim of blocking?
- Can you run the RL process without blocking?

Implement different blocking techniques

- Before we start let us have a look how simple blocking, Soundex, and the SLK-581 methods works.
- In **simple blocking** records are placed into different blocks based on the value of a chosen blocking attribute (or attributes).

- In **Soundex blocking** records are placed into different blocks based on the Soundex value. For a full description of Soundex, see lecture 14.

- In **SLK-581 blocking** we group record based on their SLK-581 identifier. SLK-581 is made up of four elements, including three letters from family name (surname or last name), two letters from given name (first name), date of birth, and gender.

Implement different blocking techniques

- Compute Soundex codes for Brian Schmidt and Queen Elizabeth II.
- Compute Soundex codes for your first name and last name.
- Compute SLK-581 for Brian Schmidt and Queen Elizabeth II. Use their publicly available personal details for computing these.
- Compute SLK-581 for you.

Soundex example

Soundex algorithm

1) Keep first letter of a string

2) Remove all following occurrences of: a, e, i, o, u, y, h, w

3) Replace all consonants from position 2 onwards with digits using these rules:

b, f, p, v → 1	c, g, j, k, q, s, x, z → 2
d, t → 3	l → 4
m, n → 5	r → 6

4) Only keep unique adjacent digits

5) If length of a code is less than 4 add zeros, if longer truncate at length 4

Attribute value : Brian

1) **B**rian

2) **B**rn

3) **B**65

4) **B**65

5) **B650**

SLK-581 example

SLK-581 steps

1) Take the 2nd, 3rd, and 5th letters of a record's family name (surname)

2) Take the 2nd and 3rd letters of the record's given name (first name)

3) Take the day, month and year of the person, concatenated in that order (ddmmyyyy) to form the date of birth

4) Take the gender of the person (1=male, 2=female, 9=unknown)

5) If names too short use 2, if full name component missing use 999

Record : Brian Schmidt, 24 February 1967, male

1) **chi**

2) **ri**

3) **24021967**

4) **1**

5) **chiri240219671**

Understanding the RL program

- Have a look through the Python skeleton modules to get a feel for how it is structured and what the different parts are.
- First look at **recordLinkage.py** since this is the module that runs the complete process.

- **Run recordLinkage.py as it is**. It will use some of the provided data sets, and the functions already implemented. This will show you what the output for the different steps will look like.

- Once your program is working, apply it on the other, larger, data sets.

Implement different blocking techniques

- Now start looking at **blocking.py** and explore how the blocking functions work (inputs, return values, etc.).
- We have already provided one blocking function, **simpleBlocking.**

- Run the blocking step on the two small data sets using both noBlocking and simpleBlocking**.**

- Now try to implement **soundexBlocking** and **slkBlocking** in the blocking module.

Questions to consider

- Can you see any difference in the number of blocks generated, the minimum, average, and maximum block sizes when you use different blocking techniques on the same data set?

- Which do you think are the best blocking functions and attributes for blocking?
- Can you come up with a list of criteria for good blocking keys based on the experiments you conducted?

- Extra tasks – see if you can implement canopy clustering or sorted neighbourhood blocking techniques.

Summary

- In this lab we implemented different blocking techniques and learnt how they can be used in the RL program.
- Make sure to complete any unfinished work in this module before you come to the next lab.
- In the next lab we will be looking at how different comparison functions work and how they can be used in the RL program.