Module X: Blocking

Rebecca C. Steorts

Agenda

- Data Cleaning Pipeline
- Blocking
- Traditional Blocking
- Probabilistic Blocking

Load R packages

```
## Loading required package: DBI

## Loading required package: RSQLite

## Loading required package: ff

## Loading required package: bit

##
## Attaching package: 'bit'

## The following object is masked from 'package:base':
##
##     xor

## Attaching package ff

## - getOption("fftempdir")=="/var/folders/bv/xhclmwh90zg08

## - getOption("ffextension")=="ff"

## - getOption("ffdrop")==TRUE
```

Data Cleaning Pipeline

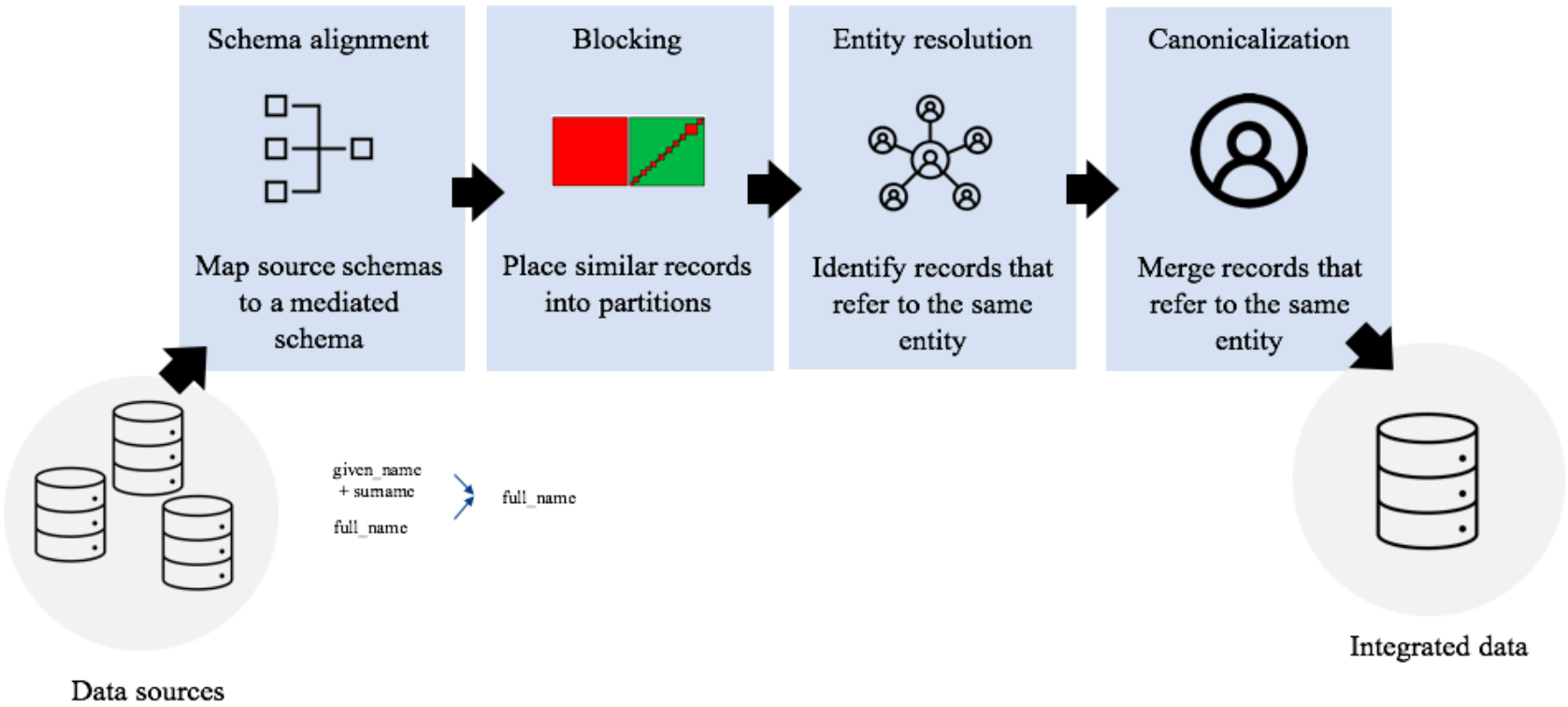

Figure 1: Data cleaning pipeline.

Blocking

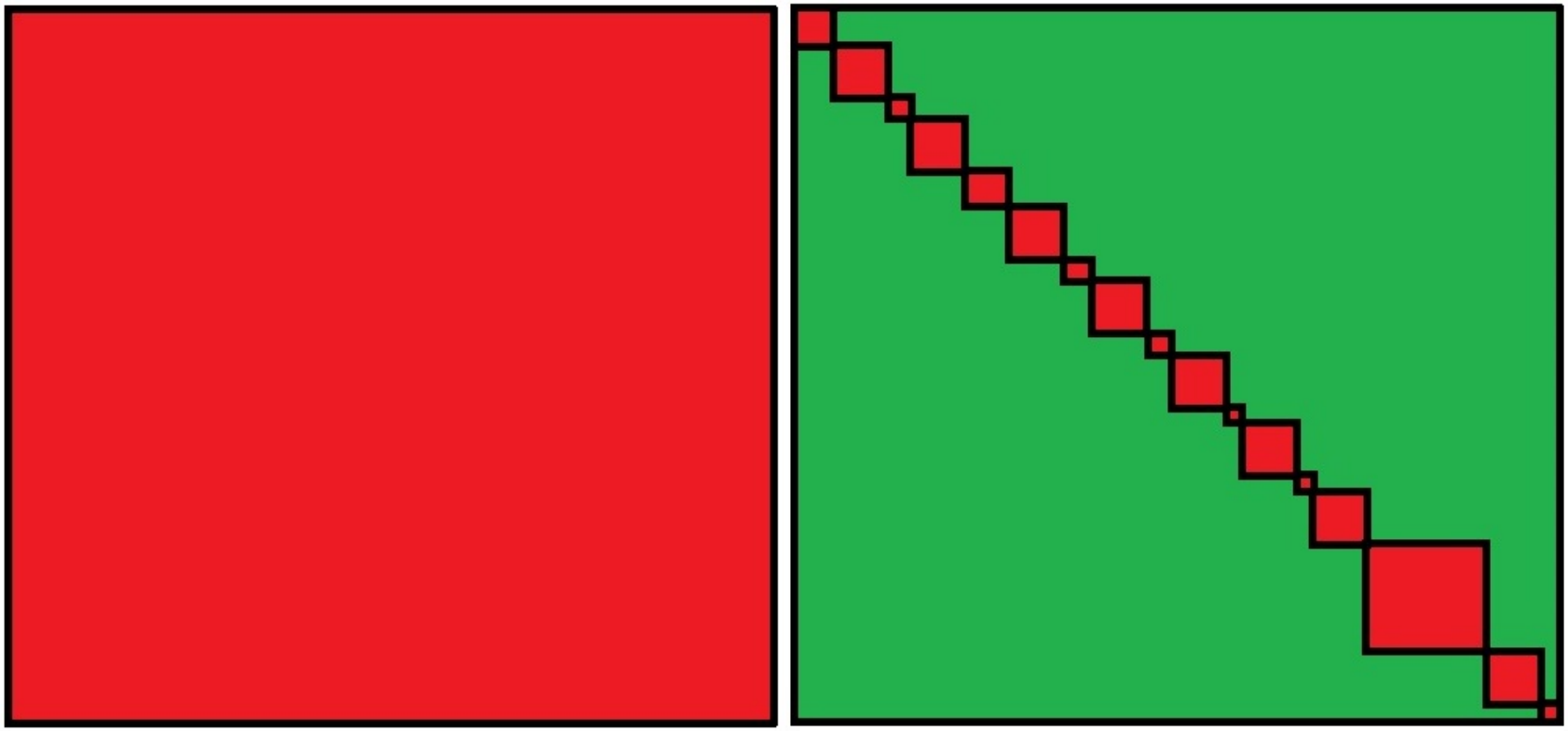

Figure 2: Left: All to all record comparison. Right: Example of resulting blocking partitions.

Blocking

- Blocking partitions similar records into partitions/blocks.
- ER is only performed within each blocks.

Traditional Blocking

- A deterministic (fixed) partition is formed based upon the data.
- A partition is created by treating certain fields that are thought to be nearly error-free as fixed.
- Benefits: simple, easy to understand, and fast to implement.
- Downsides: the blocks are treated as error free, which is not usually accurate and can lead to errors in the ER task that cannot be accounted for.

Example: Blocking on date of birth year.

Probabilistic Blocking

- A probability model is used to cluster the data into blocks/partitions.

Example: Fellegi-Sunter (1969), or Locality Sensitive Hashing

Under both blocking approaches, record pairs that do not meet the blocking criteria are automatically classified as non-matches.

Example: Traditional blocking

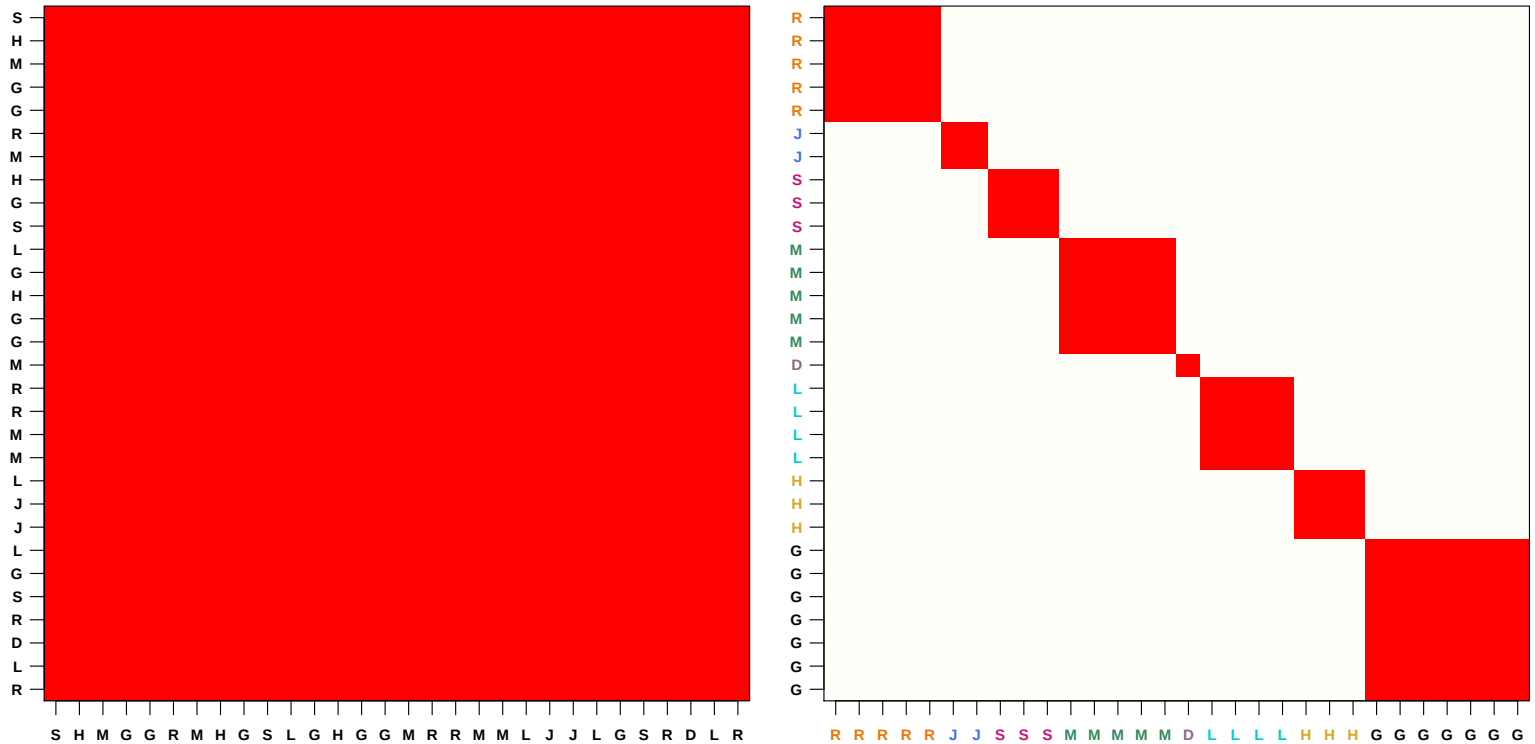

Figure 3: All-to-all record comparisons (left) versus partitioning records into blocks by lastname initial and comparing records only within each partition (right).

Example: RLdata500

```
library(RecordLinkage)
data(RLdata500)
head(RLdata500)
```

```
##   fname_c1 fname_c2 lname_c1 lname_c2   by bm bd
## 1  CARSTEN     <NA>     MEIER     <NA> 1949  7 22
## 2     GERD     <NA>     BAUER     <NA> 1968  7 27
## 3   ROBERT     <NA> HARTMANN     <NA> 1930  4 30
## 4   STEFAN     <NA>     WOLFF     <NA> 1957  9  2
## 5     RALF     <NA>   KRUEGER     <NA> 1966  1 13
## 6  JUERGEN     <NA>    FRANKE     <NA> 1929  7  4
```

RLdata500 (Continued)

```
# Total number of all to all record comparisons
choose(500,2)
```

```
## [1] 124750
```

RLdata500 (Continued)

```r
# Block by last name initial
last_init <- substr(RLdata500[,"lname_c1"], 1, 1)
head(last_init)
```

```
## [1] "M" "B" "H" "W" "K" "F"
```

```r
# Total number of blocks
length(unique(last_init))
```

```
## [1] 20
```

RLdata500 (Continued)

```r
# Total number of records per block
recordsPerBlock <- table(last_init)
head(recordsPerBlock)
```

```
## last_init
##  A  B  D  E  F  G
##  5 56  2  6 38 12
```

```r
# Block sizes can vary
summary(as.numeric(recordsPerBlock))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2.00    5.75    8.00   25.00   40.00  115.00
```

RLdata500 (Continued)

```
# Total number of records pairs per block
sapply(recordsPerBlock, choose, k=2)
```

```
##     A     B     D     E     F     G     H     J     K     L     M
##    10 1540     1    15   703    66   496    28 1035    78 2850
##     T     V     W     Z
##     1    21 1326    10
```

```
# Reduction on comparison space
sum(sapply(recordsPerBlock, choose, k=2))
```

```
## [1] 14805
```

RLdata500 (Continued)

What is the overall dimension reduction form the orginal space to the reduced space induced by blocking?

Recall the original space of comparisons was

```
choose(500,2)
```

```
## [1] 124750
```

We have reduced the number of comparisons to

```
sum(sapply(recordsPerBlock, choose, k=2))
```

```
## [1] 14805
```

Calculate the RR.

Pairwise Evaluation Metrics

Calculate the Pairwise Precision and Recall.

Case Study to XX

Merge this into an exercise or a homework assignment.