Evaluation Metrics for Blocking/Entity Resolution

Ajay Krishnamurthy - STA 325: Homework 2

General instructions for homeworks: Your code must be completely reproducible and must compile. No late homeworks will be accepted.

Reading Read the paper Binette and Steorts (2022) to get an overview of entity resolution. You'll want to refer to this during the course of the semester as it's meant to be a quick reference regarding the concepts that we will be covering. For more details, refer to the book by Christen (2012).

Advice: Start early on the homeworks and it is advised that you not wait until the day of as these homeworks are meant to be longer and treated as case studies.

Commenting code Code should be commented. See the Google style guide for questions regarding commenting or how to write code https://google.github.io/styleguide/Rguide.xml.

R Markdown Test

0. Open a new R Markdown file; set the output to HTML mode and "Knit". This should produce a web page with the knitting procedure executing your code blocks. You can edit this new file to produce your homework submission.

Total points on assignment: 2 (reproducibility) + 23 points for the assignment = 25 total points.

1. (4 points) What are the four main challenges of entity resolution?

Costly manual labelling, limited treatment of uncertainty, scalability/computational efficiency, and unreliable evaluation.

2. (4 points, 1 point each) Suppose there are 10 records in a data set. a.) What are the total number of brute-forte comparison needed to make all-to-all record comparisons?

The total number of brute-force comparisons will be 10 choose 2, which is 45 comparisons.

b.) Repeat this for 100 records, 1000 records, 10,000 records.

For 100 records, the total number of brute-force comparisons will be 100 choose 2, which is 4950.

For 1000 records, the total number of brute-force comparisons will be 1000 choose 2, which is 4.995×10^5.

For 10000 records, the total number of brute-force comparisons will be 10000 choose 2, which is 4.9995×10^7.

c.) What do you observe about the number of comparisons that need to be made?

I observe that the number of comparisons are roughly quadratic with the number of records. In other words, if there are n records, there will be roughly n^2 comparisons.

3. (9 points) Consider the following record linkage data set with 1,000,000 total records that are matched between two databases. Assume that 500,000 are true matches. Assume a classified (or method) finds 600,000 record pairs as matches, and of these 400,000 correspond as true matches. The number of TP + FP + TN + FN = 50,000,000.

a. (4 points) Given the information above, find the following information in the confusion matrix: TP, FP, TN, and FN.

$TP = 400,000$

$FP = 600,000 - 400,000 = 200,000$

$FN = 500,000 - 400,000 = 100,000$

$TN = 50,000,000 - (400,000 + 200,000 + 100,000) = 49,300,000$

b. (1 point) Calculate the accuracy. Comment on the reliability of this metric for this problem.

Accuracy = $\frac{TP+TN}{TP+FP+FN+TN} = \frac{400,000+49,300,000}{400,000+49,300,000+200,000+100,000} = 0.994$

Accuracy in this case is very high at 99.4%, but this is not necessarily a reliable metric for this problem. This is because there is a class imbalance in the data set - there are far more true negatives (49.3 million) compared to the true positives (400,000). In such cases, accuracy tends to be misleading, as it can still be high even if the classifier does poorly on the positive class. In this case, the large number of true negatives dominates the accuracy, but the method may still miss many true matches, which is not reflected well by the accuracy metric. For example, even if our predictor simply guessed negative for everything, we would have high accuracy because of the class imbalance.

c. (1 point) Calculate the precision.

Precision = $TP/(TP + FP) = \frac{400,000}{400,000+200,000} = \frac{2}{3}$

d. (1 point) Calculate the recall.

Recall = $TP/(TP + FN) = \frac{400,000}{400,000+100,000} = \frac{4}{5}$

e. (1 point) Calculate the f-measure.

f-measure = 2 * precision * recall/(precision + recall) = $\frac{2 \cdot \frac{2}{3} \cdot \frac{4}{5}}{\frac{2}{3}+\frac{4}{5}} = \frac{\frac{16}{15}}{\frac{22}{15}} = \frac{8}{11}$

f. (1 point) Comment on the reliability of the precision, recall, and f-measure for this problem.

Precision measures the proportion of predicted matches that are actually true matches. In this case, it is $\frac{2}{3}$, or 66.67%. This means that of all the records the method classified as matches, two-thirds are correct. Precision is a reliable metric for evaluating the quality of the matches identified by the method, especially when we want to minimize false positives.

Recall measures the proportion of actual true matches that were identified by the method. In this case, it is $\frac{4}{5}$, or 80%. This indicates that 80% of the true matches were found. Recall is useful when missing true matches is more critical than incorrectly labeling non-matches as matches (i.e., we want to minimize false negatives).

F-Measure is the harmonic mean of precision and recall, balancing both metrics. In this case, it is $\frac{8}{11}$, or approximately 72.73%. The F-Measure is a good overall indicator of the method's performance, especially when both precision and recall are important.

Given the class imbalance and the fact that precision and recall focus on true positives and false positives/negatives, they provide more reliable insights for this problem than accuracy.

4. (6 points) We will revisit the Italian Survey on Household and Wealth (SHIW) from class, which is a sample survey 383 households conducted by the Bank of Italy every two years (2008 and 2010). The data set is anonymized to remove first and last name (and other sensitive information).

a. (0 points) Please load the data set in the way that we did in class and block based upon gender.
b. (1 point) Plot the size of the blocks and comment on how many their are and their relative size.
c. (1 point) Calculate the reduction ratio and interpret its meaning.
d. (2 points) Calculate the precision and recall. Interpret the meaning of each.
e. (1 point) Would this be a reasonable approach for blocking. Explain.
f. (1 point) Would blocking on gender be recommended for entity resolution. Explain.

a.

```
# Installing and librarying pacman, plus loading necessary packages
if(!require("pacman")) {
install.packages("pacman")
library(pacman)
}
p_load(RecordLinkage, blink, italy, tidyverse, assert)
```

```
# Loading italy data for 2008 and 2010 and combining them without the id
library(italy)
library(assert)
data(italy08)
data(italy10)
knitr::opts_chunk$set(echo = TRUE,
                      fig.width=4,
                      fig.height=3,
                      fig.align="center")
head(italy08)
```

```
##        id PARENT SEX ANASC NASCREG CIT ACOM4C STUDIO Q QUAL SETT IREG
## 1 1040021      1   2  1948      16   1      0      5 1    2    3   16
## 2 1040022     10   2  1952      16   1      0      7 1    2    3   16
## 3 1110521      1   1  1972      20   1      2      5 1    1    4   20
## 4 1110522      3   1  1935      20   1      2      2 3    6    5   20
## 5 1110523      3   2  1941      20   1      2      3 3    6    5   20
## 6  119401      1   1  1941       7   1      0      4 3    6    5    7
```

```
head(italy10)
```

```
##        id PARENT SEX ANASC NASCREG CIT ACOM4C STUDIO Q QUAL SETT IREG
## 1 1040021      1   2  1948      16   1      0      5 3    6    5   16
## 2 1040022     11   2  1952      16   1      0      7 1    2    3   16
## 3 1110521      1   2  1941      20   1      2      3 3    6    5   20
## 4 1110522      2   1  1935      20   1      2      2 3    6    5   20
## 5 1110523      6   1  1972      20   1      2      5 1    1    4   20
## 6  119721      1   2  1948      16   1      2      2 2    5    4   17
```

```
id08 <- italy08$id
id10 <- italy10$id
id <- c(italy08$id, italy10$id) # combine the id
italy08 <- italy08[-c(1)] # remove the id
italy10 <- italy10[-c(1)] # remove the id
italy <- rbind(italy08, italy10)
head(italy)
```

```
##   PARENT SEX ANASC NASCREG CIT ACOM4C STUDIO Q QUAL SETT IREG
## 1      1   2  1948      16   1      0      5 1    2    3   16
## 2     10   2  1952      16   1      0      7 1    2    3   16
## 3      1   1  1972      20   1      2      5 1    1    4   20
## 4      3   1  1935      20   1      2      2 3    6    5   20
## 5      3   2  1941      20   1      2      3 3    6    5   20
## 6      1   1  1941       7   1      0      4 3    6    5    7
```

```
# Blocking by gender
blocksByGender <- italy$SEX
recordsPerBlock <- table(blocksByGender)
```

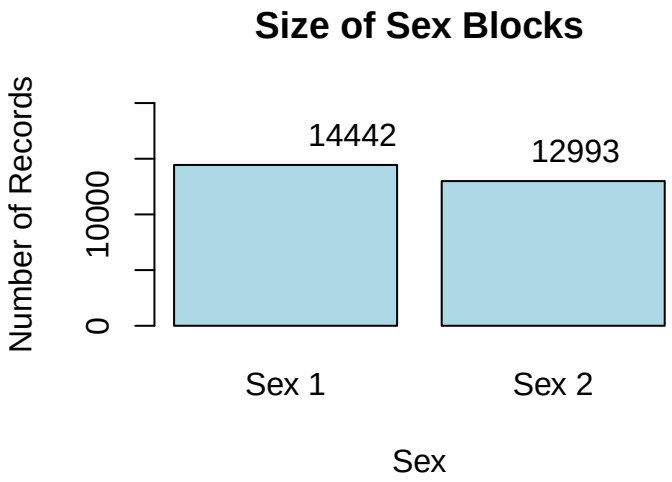
```r
head(recordsPerBlock)
```

```
## blocksByGender
##     1     2
## 14442 12993
```

b.

```r
block_labels <- c("Sex 1", "Sex 2")
# Get the number of records in each block
block_sizes = c(recordsPerBlock[[1]], recordsPerBlock[[2]])
# Create a bar plot
barplot(block_sizes,
        names.arg = block_labels,
        col = "lightblue",
        main = "Size of Sex Blocks",
        xlab = "Sex",
        ylab = "Number of Records",
        ylim = c(0, 20000)) # Setting y-axis limit slightly above the largest value for better view

# Add text labels showing the size of each block on top of the bars
text(x = c(1, 2),
     y = block_sizes,
     label = block_sizes,
     pos = 3,
     cex = 1,
     col = "black")
```



There are two blocks corresponding to two gender/sex categories (1 and 2).

The block for "Sex 1" contains 14,442 records, and the block for "Sex 2" contains 12,993 records. The sizes are somewhat comparable, with "Sex 1" having slightly more records than "Sex 2" by about 11%.

c.

```r
# Total number of records (before blocking)
total_records <- sum(block_sizes)

# Total possible comparisons without blocking (pairwise comparisons)
total_comparisons_without_blocking <- choose(total_records, 2)
```

```r
# Total possible comparisons within each block (after blocking by gender)
comparisons_after_blocking <- sum(sapply(block_sizes, function(n) (choose(n, 2))))

# Calculate the reduction ratio
reduction_ratio <- 1 - (comparisons_after_blocking / total_comparisons_without_blocking)

reduction_ratio
```

```
## [1] 0.4986234
```

The reduction ratio of 0.4986 (approximately 0.5) means that blocking by gender has reduced the number of comparisons by about 50% compared to performing comparisons on the entire dataset without any blocking.

Without blocking, we would need to perform comparisons for every possible pair of records, which is computationally expensive, especially with a large dataset. After blocking by gender, only records within the same gender group are compared, which reduces the number of comparisons by roughly half. In this case, the number of comparisons has been reduced by about 50%, meaning that only half the original comparisons are needed. A reduction ratio of 0.5 is a moderate improvement, and it shows that blocking based on gender does reduce the amount of comparisons we need significantly, but it is not an extremely high reduction like 0.9 or more. The reduction is valuable but could be improved with additional or more specific blocking criteria.

d.

```r
# Function to calculate precision for blocking
precision <- function(block.labels, IDs) {
  # Contingency table of block labels and IDs
  ct <- xtabs(~block.labels + IDs)

  # Number of true positives
  TP <- sum(choose(ct, 2))

  # Number of positives (TP + FP)
  P <- sum(choose(rowSums(ct), 2))

  # Precision
  return(TP / P)
}

# Function to calculate recall for blocking
recall <- function(block.labels, IDs) {
  # Contingency table of IDs and block labels
  ct <- xtabs(~IDs + block.labels)

  # Number of true positives
  TP <- sum(choose(ct, 2))

  # Number of true links (TP + FN)
  TL <- sum(choose(rowSums(ct), 2))

  # Recall
  return(TP / TL)
}
```

```r
italy$ID <- id
```

```r
block_labels <- italy$SEX
IDs <- italy$ID  # Adjust this according to your actual dataset structure

# Calculate precision and recall
precision_value <- precision(block_labels, IDs)
recall_value <- recall(block_labels, IDs)

# Output precision and recall values
precision_value
```

```
## [1] 3.599727e-05
```

```r
recall_value
```

```
## [1] 0.9113109
```

Precision: 3.599727e-05 (or 0.0036%) Precision is extremely low, meaning that only 0.0036% of the record pairs classified as matches by the blocking method are actually true matches. This indicates that the majority of the predicted matches are false positives, i.e., records that were incorrectly classified as matches. Therefore, although the method is finding a large number of matches, most of them are incorrect. Recall: 0.9113109 (or 91.13%) Recall is quite high, meaning that 91.13% of the actual true matches were correctly classified as matches by the blocking method. This indicates that the method is very good at finding the true matches, as it correctly identifies most of them. Overall Interpretation: High recall but very low precision means that the method is casting a very wide net and successfully capturing most of the true matches, but it is also capturing a large number of false matches (false positives). While recall is strong, the low precision shows that the method's output includes a lot of incorrect matches, which makes it unreliable for accurate entity resolution.

e.

No, blocking solely by gender would not be a reasonable approach for blocking in this case. While it may reduce the number of comparisons to some extent, it results in very low precision, which means that the vast majority of the predicted matches are incorrect. Blocking should help minimize false positives by grouping records more effectively, but since many records within the same gender group may not be actual matches, this approach fails to achieve that goal. Furthermore, gender is a very coarse attribute with limited variety (only two categories in most cases), meaning that blocking based on it alone does not create enough distinct blocks to narrow down the number of comparisons sufficiently.

f.

No, blocking on gender alone would not be recommended for entity resolution. As seen from the precision value, blocking based only on gender leads to a large number of false positives. Gender is not specific enough to separate entities effectively, resulting in a high volume of incorrect matches. In entity resolution, it is better to use more discriminating attributes—such as combinations of name, date of birth, or location—that can more reliably distinguish between different entities. Gender can be used as one of multiple blocking criteria, but relying on it alone will not provide good results. A good entity resolution algorithm should ideally have both high recall and high precision, ensuring that most of the predicted matches are correct. In this case, blocking by gender alone is not an effective strategy.