

# Example session for Weight-based deduplication

Andreas Borg, Murat Sariyar

January 11, 2012

This document shows an example session using the package *RecordLinkage*. A single data set is deduplicated using an EM algorithm for weight calculation. Conducting linkage of two data sets differs only in the step of generating record pairs.

## 1 Generating record pairs

The data to be deduplicated is expected to reside in a data frame or matrix, each row containing one record. Example data sets of 500 and 10000 records are included in the package as `RLData500` and `RLData10000`.

```
> data(RLdata500)
> RLdata500[1:5,]
```

	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

For deduplication, `compare.dedup` is to be used. In this example, blocking is set to return only record pairs which agree in at least two components of the subdivided date of birth, resulting in 810 pairs. The argument `identity` preserves the true matching status for later evaluation.

```
> pairs=compare.dedup(RLdata500,identity=identity.RLdata500,blockfld=list(c(5,6),c(6,7),c(
> summary(pairs)
```

Deduplication Data Set

500 records

571 record pairs

49 matches

522 non-matches

0 pairs with unknown status

## 2 Weight calculation

Weights are calculated by means of an EM algorithm. This step is computationally intensive and might take a while. The histogram shows the resulting weight distribution.

```
> pairs=emWeights(pairs)

> hist(pairs$Wdata, plot=FALSE)

$breaks
[1] -15 -10 -5  0  5 10 15 20 25 30 35
[12] 40 45

$counts
[1] 352 13  0  0  5 26 42 123  9  0  0
[12]  1

$intensities
[1] 0.1232924694 0.0045534151 0.0000000000
[4] 0.0000000000 0.0017513135 0.0091068301
[7] 0.0147110333 0.0430823117 0.0031523643
[10] 0.0000000000 0.0000000000 0.0003502627

$density
[1] 0.1232924694 0.0045534151 0.0000000000
[4] 0.0000000000 0.0017513135 0.0091068301
[7] 0.0147110333 0.0430823117 0.0031523643
[10] 0.0000000000 0.0000000000 0.0003502627

$mids
[1] -12.5 -7.5 -2.5  2.5  7.5 12.5 17.5
[8] 22.5 27.5 32.5 37.5 42.5

$xname
[1] "pairs$Wdata"

$equidist
[1] TRUE

attr(,"class")
[1] "histogram"
```

## 3 Classification

For determining thresholds, record pairs within a given range of weights can be printed using `getPairs`<sup>1</sup>. In this case, 24 is set as upper and  $-7$  as lower

---

<sup>1</sup>The output of `getPairs` is shortened in this document.

threshold, dividing links, possible links and non-links. The summary shows the resulting contingency table and error measures.

```
> getPairs(pairs,30,20)
```

	id	fname_c1	fname_c2	lname_c1	lname_c2	by
23	457	URSULA	BIRGIT	MUELLER	<NA>	1940
24						
25	467	ULRIKE	NICOLE	BECKRR	<NA>	1982
26	472	ULRIKE	NICOLE	BECKER	<NA>	1982
27						
28	183	ULRICH	<NA>	MUELLER	<NA>	1962
29	444	SILKE	<NA>	MUELLER	<NA>	1962
30						
31	25	MATTHIAS	<NA>	HAAS	<NA>	1955
32	107	MATTHIAS	<NA>	HAAS	<NA>	1955
33						
34	106	ANDRE	<NA>	MUELLER	<NA>	1976
35	175	ANDRE	<NA>	MUELLER	<NA>	1976
36						

	bm	bd	Weight
23	6	15	25.14137
24			
25	8	4	
26	8	4	25.14137
27			
28	6	19	
29	6	14	24.20333
30			
31	7	8	
32	8	8	24.11923
33			
34	2	25	
35	1	25	24.11923
36			

```
> pairs=emClassify(pairs, threshold.upper=24, threshold.lower=-7)
> summary(pairs)
```

Deduplication Data Set

500 records

571 record pairs

49 matches

522 non-matches

0 pairs with unknown status

Weight distribution:

[-15,-10]	(-10,-5]	(-5,0]	(0,5]	(5,10]
352	13	0	0	5
(10,15]	(15,20]	(20,25]	(25,30]	(30,35]
26	42	123	9	0
(35,40]	(40,45]			
0	1			

15 links detected  
198 possible links detected  
358 non-links detected

alpha error: 0.000000  
beta error: 0.002786  
accuracy: 0.997319

Classification table:

		classification		
true status		N	P	L
	FALSE	358	163	1
	TRUE	0	35	14

Review of the record pairs denoted as possible links is facilitated by `getPairs`, which can be forced to show only possible links via argument `show`. A list with the ids of linked pairs can be extracted from the output of `getPairs` with argument `single.rows` set to `TRUE`.

```
> possibles <- getPairs(pairs, show="possible")
> possibles[1:6,]

  id  fname_c1 fname_c2 lname_c1 lname_c2  by
1  17 ALEXANDER      <NA>  MUELLER      <NA> 1974
2 193 CHRISTIAN      <NA>  MUELLER      <NA> 1974
3
4  61      ANDRE      <NA>  FISCHER      <NA> 1943
5 254 STEFANIE      <NA>  FISCHER      <NA> 1943
6
  bm bd  Weight
1  9  9
2  8  9 21.691086
3
4  6 25
5 11 25 21.691086
6

> links=getPairs(pairs,show="links", single.rows=TRUE)
> link_ids <- links[, c("id1", "id2")]
> link_ids

  id1 id2
290 290 466
```

50 50 234  
87 87 117  
145 145 240  
286 286 383  
289 289 399  
297 297 388  
357 357 414  
313 313 457  
467 467 472  
183 183 444  
25 25 107  
106 106 175  
370 370 478  
127 127 142

>