Example Session for Package RecordLinkage

Andreas Borg

14. Januar 2009

Load example data:

```
> data(RLdata500)
```

The example data set has the fields:

fname_c1 First name, first component

fname_c2 First name, second component

lname_c1 Last name, first component

lname_c2 Last name, second component

by Year of birth

bm Month of birth

bd Day of birth

List some records:

```
> RLdata500[1:5, ]
```

```
     fname_c1   fname_c2 lname_c1  lname_c2 by     bm   bd
[1,] "CEM"      ""       "KRAUSE"  ""       "1997" " 2" " 12"
[2,] "NICK"     ""       "HUEBNER" ""       "1996" " 4" " 30"
[3,] "J"        ""       "MEYER"   ""       "1979" " 2" " 26"
[4,] "FILIZ"    ""       "AKKOC"   ""       "1983" " 6" " 11"
[5,] "PATRICIA" ""       "POLMANS" ""       "1989" " 5" "  4"
```

For deduplication, `compare_dedup` is to be used. In our example, blocking gives all record pairs which agree in at least two components of the date of birth. 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(5, 7)))
```

```
[1] "blocking beginnt"
[1] "blockstr"
[1] "tapply"
[1] "delete.NULLs"
[1] "lapply"
```

```
[1] "unlist"
[1] "pair_ids"
[1] "blockstr"
[1] "tapply"
[1] "delete.NULLs"
[1] "lapply"
[1] "unlist"
[1] "pair_ids"
[1] "blockstr"
[1] "tapply"
[1] "delete.NULLs"
[1] "lapply"
[1] "unlist"
[1] "pair_ids"
[1] "blocking beendet"
[1] "vor unique"
[1] "nach unique"
[1] 810
[1] "merge"
[1] "nach merge"
[1] "Vergleich"
[1] "Trainingsdaten ziehen"
[1] "Daten zusammenführen"
```

```
> summary(pairs)
```

```
Deduplication Project
500 records
0 training pairs
810 validation pairs

0 matches in training set
0 non-matches in training set

46 matches in validation set
764 non-matches in validation set
```

Calculate weights with EM algorithm:

```
> pairs = emWeights(pairs)
```

```
[1] "Datenvorbereitung"
   user  system elapsed
      0       0       0
[1] "Fuzzy umrechnen"
   user  system elapsed
  0.000   0.000   0.002
[1] "Patterns zählen, em vorbereiten"
   user  system elapsed
  0.000   0.000   0.003
[1] "Häufigkeiten schätzen"
```

```
> hist(pairs$Wdata)
```

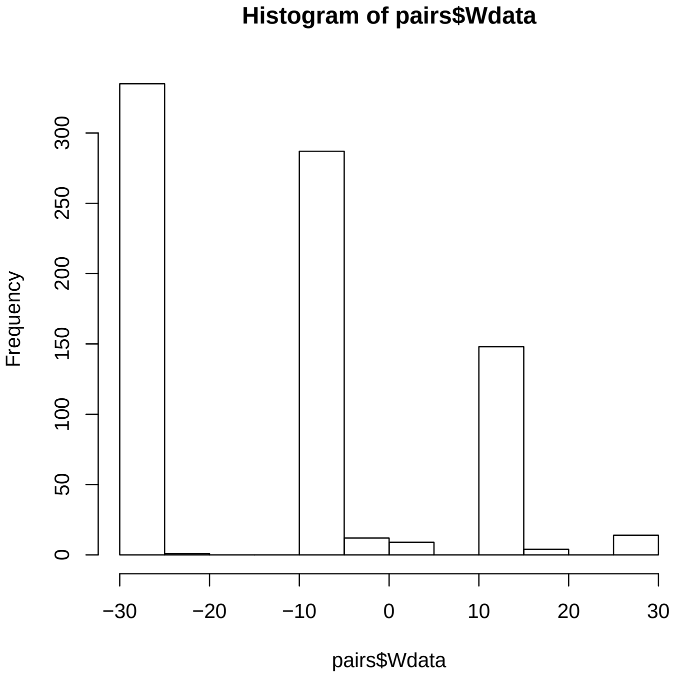

Abbildung 1: Weights histogram.

```
   user  system elapsed
  0.010   0.000   0.012
[1] "EM ausführen"
   user  system elapsed
  70.92    0.29   71.79
[1] "Der Rest"
   user  system elapsed
  0.000   0.000   0.001
```

A histogram gives information on weight distribution, see figure 1.

For determining thresholds or clerical review, record pairs within a given range of weights can be printed using `print.range`

```
> print.range(pairs, 15, 10)
```

Based on the output, 11 is set as upper and lower threshold in this case, dividing links from non-links. The summary shows that 36 matches were correctly classified while 10 matches were not detected.

```
         V1        V2   V3         V4   V5   V6 V7  V8
25 11.60721  ANNETTE <NA>       DITZ <NA> 2002  1   1
26          ANNETWTE <NA>       DITZ <NA> 2002  1   1
27 11.60721   NIKLAS <NA>   HEUTINK <NA> 2002  7  26
```

```
28           NIKLNAS <NA>   HEUTINK <NA> 2002  7  26
29 11.52404 MATTHIAS <NA>   HORBACH <NA> 1975  9  35
30          MATTHIAS <NA>   HORBACH <NA> 1975  9  15
31 11.52404   AGATHE <NA>    GLADER <NA> 1977  8  79
32            AGATHE <NA>    GLADER <NA> 1977  8  29
33 11.52404   FABIAN <NA>     BRUNS <NA> 1987  6 922
34            FABIAN <NA>     BRUNS <NA> 1987  6  22
35 11.52404 PATRICIA <NA>   POLMANS <NA> 1989  5   4
36          PATRICIA <NA>   POLMANS <NA> 1989  5  14
37 10.99097    DANER <NA>    GLASSL <NA> 1975  5  13
38           TORSTEN <NA>     FIALA <NA> 1975  2  13
39 10.99097  GOWSIYA <NA>   MATZNER <NA> 1975  1  15
40          MATTHIAS <NA>   HORBACH <NA> 1975  9  15
41 10.99097  MARTINA <NA>   WIENEKE <NA> 1975 12  21
42               IDA <NA> KALEMBACH <NA> 1975  8  21
43 10.99097    JULIA <NA>    FOLMER <NA> 1975  9   5
44            MORITZ <NA>    WIERER <NA> 1975 11   5
```

```
> pairs = emClassify(pairs, threshold_lower = 11)
> summary(pairs)
```

```
Deduplication Project
500 records
0 training pairs
810 validation pairs

0 matches in training set
0 non-matches in training set

46 matches in validation set
764 non-matches in validation set

0 links detected
36 possible links detected
774 non-links detected

alpha error: 1.000000
beta error: 0.000000
accuracy: 0.987080


Classification table:

           prediction
true status FALSE <NA>
      FALSE   764    0
      TRUE     10   36
```

Detected links can be extracted for further processing:

```
> links = print.results(pairs, show = "links")
```