Bouncing Around

Run the Model

```
rm(list=ls())
set.seed(2)
R <- NULL
show_progress <- T
fast <- F
S <- 1000
burn <- S * .1
iter_count <- seq_len(S)

cd <- readRDS("../data/comparison_elsalvador_smallP.rds")
nA <- cd$n1
nB <- cd$n2
levels <- cd[[4]]
P <- prod(levels)
var_names <- cd$compFields[,1]
ptm <- proc.time()
Zchain_fabl <- BKSimple_hash2(cd, S = S, R = R, show_progress = T, all_patterns = FALSE)
#>  Simulation: 1% complete Simulation: 2% complete Simulation: 3% complete Simulation: 4% complete Sim
elapsed_fabl <- proc.time() - ptm
Zchain_fabl[[4]]
#> elapsed
#>  124.47
Zhat_fabl <- LinkRecordsBK(Zchain_fabl[[1]], nA, 1, 1, 2, Inf)
```

Exploring "Bouncing" Matches

The object `Zhat` below contains the Bayes estimate for each component Z_j and the posterior probability of that decision. Here, $n_A = 4420$, so everything where $Z_j > 4420$ indicates a nonmatch.

I single out examples where the posterior probability is less than 0.5, because those are situations where record j was matched to some record more than 50% of the time, but it was not matched to the same record consistently enough for the Bayes estimate.

```
Zhat <- cbind(Zhat_fabl[[1]], Zhat_fabl[[2]])
bouncing_matches <- which(Zhat[, 2] < .5)
examples <- Zhat[bouncing_matches, ]
rownames(examples) <- bouncing_matches
colnames(examples) <- c("Zhat", "probability")

examples
#>      Zhat probability
#> 253  4673       0.470
#> 254  4674       0.080
#> 255  4675       0.087
```

```
#> 256  4676        0.106
#> 257  4677        0.098
#> 258  4678        0.094
#> 259  4679        0.102
#> 263  4683        0.082
#> 264  4684        0.105
#> 265  4685        0.092
#> 266  4686        0.130
#> 267  4687        0.132
#> 268  4688        0.133
#> 269  4689        0.133
#> 270  4690        0.132
#> 272  4692        0.139
#> 274  4694        0.293
#> 275  4695        0.338
#> 278  4698        0.469
#> 281  4701        0.109
#> 282  4702        0.113
#> 283  4703        0.106
#> 374  4794        0.490
#> 510  4930        0.467
#> 567  4987        0.084
#> 568  4988        0.140
#> 658  5078        0.341
#> 659  5079        0.494
#> 674  5094        0.373
#> 688  5108        0.311
#> 695  5115        0.319
#> 787  5207        0.486
#> 812  5232        0.477
#> 827  5247        0.482
#> 1066 5486        0.496
#> 1072 5492        0.494
#> 1073 5493        0.486
#> 1074 5494        0.436
#> 1094 5514        0.489
#> 1096 5516        0.490
#> 1097 5517        0.493
#> 1098 5518        0.419
#> 1265 5685        0.477
```

Examples of "Bouncing Matches"

In this first example, record $253 \in ER$ is declared a nonmatch in the sampler with probability 0.47. So it is declared "a match" with probability .53, but that is split across 65 records!

```
probs_for_bouncing_matches <- lapply(bouncing_matches, function(x){
  Zchain_fabl[[1]][x, ] %>%
    table()/S
})
names(probs_for_bouncing_matches) <- bouncing_matches

probs_for_bouncing_matches[1]
```

```
#> $`253`
#> .
#>   244   247   254   291   338   384   396   408   425   475   480   487   495
#> 0.001 0.001 0.001 0.001 0.012 0.001 0.001 0.001 0.001 0.011 0.001 0.001 0.001
#>   514   534   557   559   581   595   631   632   633   635   636   637   638
#> 0.001 0.007 0.002 0.002 0.001 0.001 0.002 0.002 0.001 0.002 0.001 0.001 0.001
#>   639   640   642   645   652   653   655   660   662   663   664   665   666
#> 0.003 0.001 0.001 0.001 0.001 0.001 0.004 0.002 0.001 0.001 0.002 0.002 0.002
#>   667   670   671   673   676   681   683   685   686   687   689   690   691
#> 0.002 0.001 0.001 0.003 0.001 0.002 0.001 0.001 0.002 0.224 0.001 0.002 0.001
#>   692   693   695   696   697   698   699   704   705   709   710   713   717
#> 0.001 0.191 0.001 0.001 0.002 0.001 0.001 0.001 0.003 0.002 0.002 0.001 0.001
#>  4421
#> 0.470
```

More egregiously, record $254 \in ER$ is declared a nonmatch with probability 0.03, and is declared a match 97% of the time! However, that is split across many many records, so the Bayes esimate is declares a nonmatch.

```
probs_for_bouncing_matches[2]
#> $`254`
#> .
#>   229   237   238   273   275   286   288   295   299   300   301   317   329
#> 0.005 0.008 0.003 0.002 0.006 0.009 0.001 0.009 0.002 0.005 0.003 0.002 0.004
#>   331   350   351   352   353   361   374   375   377   381   389   391   393
#> 0.009 0.002 0.008 0.007 0.001 0.003 0.004 0.003 0.002 0.003 0.001 0.004 0.002
#>   395   397   403   429   431   432   433   435   441   446   455   461   487
#> 0.006 0.003 0.005 0.004 0.004 0.003 0.001 0.003 0.006 0.006 0.003 0.001 0.001
#>   488   489   490   505   568   583   587   589   628   635   643   644   645
#> 0.001 0.002 0.003 0.001 0.004 0.002 0.004 0.003 0.003 0.069 0.067 0.069 0.067
#>   646   648   654   657   672   679   683   686   697   701   703   715  4421
#> 0.080 0.074 0.001 0.079 0.001 0.001 0.080 0.076 0.064 0.001 0.060 0.001 0.033
```

Looking at the Records

Here, we inspect one record in ER and the multiple records it is matched to.

```
CDHES <- readRDS("../data/CDHES.rds")
ER <- readRDS("../data/ER.rds")

ER[254, ]
#>        lastname firstname record_id dataset day month year    dept muni
#> 254 CHICAS DIAS   DOROTEO       792   CDHES  15    12 1981 MORAZAN <NA>
```

```
fleeting_matches <- names(probs_for_bouncing_matches[[2]]) %>%
  as.numeric() %>%
  .[1:10]

CDHES[fleeting_matches, ]
#>     lastname     firstname record_id dataset day month year    dept muni
#> 229   CHICAS       SEGUNDO       996   ER-TL   8    12 1981 MORAZAN <NA>
#> 237   CHICAS       RUPERTO      1007   ER-TL   8    12 1981 MORAZAN <NA>
#> 238   CHICAS         MIRNA      1008   ER-TL   8    12 1981 MORAZAN <NA>
#> 273     DIAS       MELESIO      1063   ER-TL   8    12 1981 MORAZAN <NA>
#> 275     DIAS        MARTIR      1065   ER-TL   8    12 1981 MORAZAN <NA>
#> 286     DIAS       ROJELIA      1089   ER-TL   8    12 1981 MORAZAN <NA>
#> 288     DIAS       LORENSO      1093   ER-TL   8    12 1981 MORAZAN <NA>
#> 295   CHICAS ARTURO JIDIO      1103   ER-TL   8    12 1981 MORAZAN <NA>
#> 299     DIAS         PAULA      1112   ER-TL   8    12 1981 MORAZAN <NA>
#> 300     DIAS       DOMINGA      1113   ER-TL   8    12 1981 MORAZAN <NA>
```

Reasons for "Bouncing Matches"

Some of this occurs because of the construction of the comparison vectors. Because of the way that Sadinle designed the string distance metric and the thresholds for the comparison vectors, both "CHICAS" and "DIAS" are coded as full agreements with "CHICAS DIAS." This code shows the comparison vectors for all these pairs, but since BRL uses a one-hot-encoding of the comparison vector, its a little hard to read, so I'm commenting out. I can show you when we meet though if you want!

```
# index <- expand.grid(1:nA, 1:nB)
# index_cd <- which(index[, 2] == 254 & index[, 1] %in% fleeting_matches)
# cd[[1]][index_cd, ]
```

If the vectors are not made poorly, they can smooth over meaningful distinctions in the data, or make distinctions when they don't really exist. I don't want the paper to get into the weeds of how Sadinle makes his comparison vectors (there are different ways that may prove better), but this does seem to be a problem intrinsic to `BRL` and `fabl` frameworks.

(I still note that the majority of matches are made with very high confidence! This only occurs for a small portion of matches.)

Interpretation of Posterior Distribution

The Bayes estimate is designed to declare records that "bounce around" the matching space to be nonmatches. It seems like the overcounting is inherent to the method. I have seen that the posterior probabilities of the matches is well calibrated and meaningful, but it seems like the model does not give reasonable uncertainty quantification about the *number* of matches.

I am not too concerned about this because many times, this is not a relevant quantity. If you're estimating casualty counts, its super relevant. But I don't think the DNC (for example) is ever interested in estimating the number of individuals that can be linked to the voterfile; they're interested in getting a set of links, and uncertainty quantification on those links.

I'm curious to hear your thoughts on this!