# The `likeLTD` software: an illustrative analysis, explanation of the model, results of performance tests and version history

David J. Balding, Chris Steele
UCL Genetics Institute
Darwin Building, Gower Street
London WC1E 6BT
`d.balding@ucl.ac.uk`

July 4, 2014

This version coincides with Release 5-4, which includes a new optimisation wrapper function, `enumerate`, that adds geometric progression of optimisation, and a progress bar. This particularly improves the optimisation of difficult cases. Small improvements to the output reports are also included. See Section 1.3.

The previous version (Release 5-3) improved the allele and output reports, which now produce .doc files, with the same functionality as previous reports. These include improved layout and presentation, and modifications to the allele report. See Sections 1.2 & 1.4.

The version before (Release 5-2) introduced the function `get.likely.genotypes` that returns the most probable genotypes for each locus, and the most probable whole-profile genotype. See Section 1.5 for more information.

**Abstract**

`likeLTD` ("likelihoods for Low Template DNA profiles") is an `R` package for computing likelihoods for DNA profiles. It is particularly suited for low-template and/or degraded DNA when alleles from some contributors may be subject to dropout. It can handle multiple profiled possible contributors and up to two unprofiled contributors, in addition to the queried contributor. The package also provides input files for an example analysis (the "Hammer Case" described below).

This document shows how to install and run `likeLTD` using an illustrative example. It also describes the model underlying `likeLTD`, for example explaining the "uncertain" category for allele designations, and the dropout and degradation models. We present results of running `likeLTD` on a range of single-contributor and mixed DNA profiles subject to modifications, such as introduction of artificial dropout and dropin. Some of the material here, and some other analyses, are published in Balding (2013).

In all the tests described here, unless otherwise stated we have used Version 5.4.0 of `likeLTD`, with a standard allele frequency database of around 200 UK Caucasians, $F_{ST} = 0.02$ and a sampling adjustment adj $= 1$ (these are described briefly below).

# Contents


**1 Installation and example `R` script** **4**

1.1 Input . . . 4
1.2 Allele report . . . 5
1.3 Arguments and optimisation . . . 6
1.4 Output report . . . 7
1.5 Genotype probabilities . . . 9

**2 Overview and description of the model** **12**

2.1 Overview . . . 12
2.1.1 The contributors of DNA . . . 12
2.1.2 The parameters . . . 13
2.1.3 Key features of `likeLTD` . . . 13
2.2 Single-locus LR with dropout . . . 14
2.3 Effect of an "uncertain" allele designation . . . 15
2.4 Additional contributors . . . 16
2.5 Multi-dose dropout model . . . 17
2.6 Degradation model . . . 19
2.7 Dropin . . . 19
2.8 Maximising the penalised likelihood . . . 20
2.9 Computing time and memory requirements . . . 21

**3 Performance tests of `likeLTD`** **23**

3.1 Validation of likelihood ratios . . . 23
3.2 A one-contributor CSP with dropout and dropin . . . 24
3.3 Two unprofiled contributors . . . 27
3.4 Three unprofiled contributors . . . 31
3.5 Complex Case . . . 31
3.6 Modifying the model . . . 33
3.6.1 Including a dropin term . . . 33
3.6.2 Uncertain allele calls . . . 34
3.6.3 Varying the dropout power parameter . . . 35
3.6.4 Varying the dropout locus adjustments . . . 35
3.6.5 Varying the degradation model . . . 35
3.7 Conclusions . . . 36

**4 Acknowledgments and version history** **36**

**A Allele report for Hammer case** **42**

**B Output file for Hammer case** **48**

# 1 Installation and example R script

Installing likeLTD (only needs doing once on any computer) and loading it (once per R session) are both very simple.

```
install.packages("likeLTD")
require(likeLTD)
```

The install.packages command may generate a request for you to choose a site from which to download the package. Choose any site near you.

The example analysis that comes with likeLTD is called the Hammer Case. The DNA profiles come from Table 2 of Gill et al. (2007), who introduced the software LoComatioN which in some respects is similar to likeLTD and also to LRmix, available within the Forensim R package. The crime scene profile (CSP) consists of two profiling runs at each of 10 loci. These, and reference profiles from a queried contributor Q and two victims (K1 and K2), are available in input files hammer-CSP.csv and hammer-reference.csv. likeLTD allows "uncertain" allele calls but this designation was not used by Gill et al. (2007) and so there are no alleles labelled as uncertain in this example.

There is a total of six alleles, all of them unreplicated, that are not attributable to any of Q, K1 or K2. No more than two of these occurs at any one locus. This suggests a comparison of the following two hypotheses (where $H_p$ stands for the prosecution hypothesis, and $H_d$ stands for the defence hypothesis) for the contributors of DNA to the sample:

$$\begin{aligned} H'_p: &\quad \text{Q + K1 + K2 + U1} \\ H'_d: &\quad \text{X + K1 + K2 + U1} \end{aligned}$$

where X and U1 denote unprofiled, unrelated individuals not related to any of Q, K1 and K2.

## 1.1 Input

We now show how to calculate likelihoods under $H'_p$ and $H'_d$ using likeLTD. The first command below finds out where your system has stored the Hammer Case files, and saves that location in datapath. For your own analyses, you will need to create your own CSP and reference files, in the same format as hammer-CSP.csv and hammer-reference.csv. It is usually most convenient to create these files in a specific directory, and then set that to be the working directory for R using the command setwd() or using the R menu option (its location varies across operating systems). For example if your case files are

in the directory `C:/Users/JoeBloggs/Cases/JoeBloggs1` then you enter the command `setwd("C:/Users/JoeBloggs/Cases/JoeBloggs1")`. In that case you can set `datapath = "."` in place of the first command below. A default allele frequency database file is provided with `likeLTD`. To use your own database file instead (must be in same format) set `databaseFile` to the filename, including path if not in the working directory. If you wish to choose a different individual to be Q, or to add or omit one of the other profiled contributors (a K, in the notation used here, standing for "known") then you must create a new reference file.

```
datapath = file.path(system.file("extdata", package="likeLTD"),"hammer")

# File paths and case name for allele report
admin = pack.admin.input(
            cspFile = file.path(datapath, 'hammer-CSP.csv'),
            refFile = file.path(datapath, 'hammer-reference.csv'),
            caseName = "hammer"
             )
```

Values are required for `cspFile` and `refFile`, but `caseName` can be omitted above in which case it defaults to `"dummy"`. Two other arguments have been omitted so that their default values will be used: `databaseFile = NULL` and `outputPath = getwd()`.

## 1.2 Allele report

```
# Next we generate an allele report
allele.report(admin)
```

The allele report is a .doc that will be created in the current working directory (set `outputPath` to specify a different directory). The .doc file can be converted to .pdf by opening with MS Word or another document editor and saving as a pdf. The report generated by the above command (`hammer-Allele-Report-1.doc`) is shown in Appendix A. It summarises the input data, highlights rare alleles, and suggests values for key parameters (and hence the suitable hypotheses), in particular specifying the number of unprofiled contributors required to explain the observed CSPs under the $H_p$, and whether modelling dropin is necessary. Here, it indicates that one unknown contributor is sufficient under $H_p$ to explain the observed alleles not attributable to Q/X or K1 or K2 by including dropin. While it is never possible to specify an upper bound on the number of known contributors, specifying more than the minimum required usually has negligible impact on the resulting likelihood ratio (LR). Cowell et al. (2013) illustrate this with an example

in which a log(LR) of 14.09 with three contributors barely changes as the number of contributors increases, reaching 14.04 with eight contributors.

## 1.3 Arguments and optimisation

Based on the allele report we specify the required hypotheses by setting a list of arguments containing the following items:

`nUnknowns`: The number of unknown contributors under the prosecution hypothesis (non-negative integer). `likeLTD` automatically adds an unknown contributor (X) under the defence hypothesis.

`doDropin`: Whether to model dropin or not (logical: TRUE or FALSE).

`ethnic`: The ethnic category of the queried contributor. The default database comes with "EA1","EA3" and "EA4", corresponding to UK residents of Caucasian, Afro-Caribbean and South Asian origin respectively. If you use your own allele frequency database you will choose your own category labels (required even if there is only one category).

`adj`: Sampling adjustment (scalar). See Section 3.2.

`fst`: $F_{ST}$ adjustment for distant relatedness (coancestry) of Q and X (scalar). See Section 3.2.

`relatedness`: Relatedness coefficients for Q and X (vector of length two): the probabilities that they have one and two alleles identical by descent from recent common ancestors (e.g. parents or grandparents). The setting used below is for Q and X unrelated; for siblings use `relatedness = c(0.5,0.25)`.

```
# Enter arguments
args = list(
        nUnknowns = 1,
        doDropin = FALSE,
        ethnic = "EA1",
        adj = 1,
        fst = 0.02,
        relatedness = c(0,0)
        )
```

```
# Create hypotheses
hypP = do.call(prosecution.hypothesis, append(admin,args))
hypD = do.call(defence.hypothesis, append(admin,args))

# Get parameters for optimisation
paramsP = optimisation.params(hypP)
paramsD = optimisation.params(hypD)

# Run optimisation
results = evaluate(paramsP, paramsD)
```

The entries in `args` shown above are defaults, except for `nUnknowns` which defaults to 0. The function `do.call` calls the function given in its first argument. `prosecution.hypothesis` and `defence.hypothesis` are both functions defined within `likeLTD`, which generate the necessary objects for $H_p$ and $H_d$ respectively. The `evaluate` function is likewise defined within `likeLTD`, and is a wrapper function for the `DEoptim::DEoptim` function that performs optimisation (see Section 2.8), providing a geometric progression of convergence, and a progress bar displaying current WoE. The progress bar can be disabled by setting the argument `progBar = FALSE`, which is necessary if you do not have graphical capabilities e.g. running from command line on a server. The `evaluate` function now splits the convergence into a number of chunks, with each subsequent chunk having more stringent convergence tolerance and an increased crossover rate (a parameter for `DEoptim`); the combination of these two behaviours means that the parameter space is searched more thoroughly to start with, in an effort to improve optimisation of tough cases. The number of chunks to run is determined by the difficulty of converging the first chunk. The function `optimisation.params` sets the parameter values needed for `DEoptimLoop`. These values can be altered if required but the default settings should be adequate for most analyses.

The object returned by `evaluate` is a list of three element: `Pros`, `Def` and `WoE`. Both `Pros` and `Def` have the same structure as the object returned by `DEoptim` (see `help(DEoptim)`), with each corresponding to the prosecution and defence results respectively. `WoE` gives the WoE for each chunk run by `evaluate` in bans, the final WoE can be obtained through the command `results$WoE[length(results$WoE)]`.

## 1.4 Output report

```
# Generate output report
output.report(hypP,hypD,results)
```

The results are given in the output file `hammer-Evaluation-Report-1.doc` (the numbering of the filename increments automatically, or a custom filename may be specified with `file="fileName.doc"`) which again summarises the input data, similar to the allele report, but also states the hypotheses compared and gives single-locus and overall likelihood ratios (LR) in favour of the prosecution hypothesis relative to the defence hypothesis. Logarithms (base 10) are also given for the LRs, and we will refer to the $\log_{10}$(LR) values as the weight of evidence (WoE) measured in bans. The output file for the Hammer case analysis is given in Appendix B. The overall WoE was found to be 10.9 bans. The WoE is $> 0$ (favours $H'_p$ over $H'_d$) at every locus except D18 (WoE = -0.5 bans). The most incriminating locus is D19 (WoE = 2.6 bans), where the two alleles of Q are rare, replicated in the CSP, and not shared with either K1 or K2.

Dropout rate estimates for each contributor subject to dropout, and each profiling run, are also given in the output file. These estimates are often not precise, particularly under $H'_d$ where there are two unprofiled contributors, but this is not important for assessing the WoE against Q. Note that under $H'_d$, X and U1 are indistinguishable; we usually take X to be the one with dropout rates most similar to Q, but the labelling in the output file is arbitrary. The single-allele dropout rates for Q/X are estimated at 11%/12% and 0%/1% in the two replicates. The dropout rate estimates for K1 under $H'_p/H'_d$ are 46%/44% for replicate a and 3%/5% for b. The degradation value for K1 ($\gamma_{K1}$) is about 0.8% under both hypotheses, while $\gamma_Q$ and $\gamma_X$ are both about 0.4%, indicating an increase in dropout rate with fragment length, an effect of degradation, particularly for K1.

Every CSP allele attributable to K2 could also come from K1 or Q, and so under $H'_p$ there is no evidence for DNA from K2. However, under $H'_d$ the DNA of Q is not present, leaving three CSP alleles that can be attributed to K2 but not K1. Nevertheless, `likeLTD` estimates 100% dropout of the alleles of K2 in both replicates and under both hypotheses. This is because the three alleles attributable to K2 under $H'_d$ are all replicated, whereas seven other alleles of K2 do not appear at all, indicating very high dropout, and so `likeLTD` finds that attribution of the three alleles to K2 is unlikely. Although we cannot exclude K2 from contributing any DNA to the sample, these results indicate that including K2 in the analysis brings no explanatory power and so has negligible impact on the WoE implicating Q as a contributor. It isn't necessary to exclude K2 from the analysis, because `likeLTD` has automatically done this by estimating dropout at 100%, but there may be a slight improvement in the likelihood optimisation in running the analysis again without K2, due to fewer nuisance parameters to be estimated.

## 1.5 Genotype probabilities

LikeLTD can provide a list of the most probable genotypes at each locus for each unprofiled contributor, using function `get.likely.genotypes` (new in Version 5.2). By default only single-locus genotypes with probability $> 0.1$ are returned; this can be altered using the argument `prob`. The most probable whole-profile genotype, and its probability are also returned (Figure 1). The genotypes and their probabilities add nothing to the assessment of weight of evidence against an alleged contributor of DNA, but can be useful for searches in a database.

```
# Get the most likely single-contributor genotypes
gens = get.likely.genotypes(hypD,paramsD,results$Def)
```

The returned list object is organised into a series of levels, as shown in Figure 1.

It may also be desirable to obtain the probabilities of joint genotypes, rather than genotypes of single contributors. In this case, the argument `joint` can be handed to `set.likely.genotypes`, and if set to `TRUE`, the joint genotypes and probabilities will be returned rather than the single-contributor genotypes and probabilities. For joint genotypes the default probability threshold for single-locus genotypes is 5%. This value can be altered for the single-contributor or joint cases by setting `prob`.

Note some of the locus-specific genotypes used to contruct the whole profile genotype may have smaller probabilities than the threshold, and will therefore not be displayed in the locus-specific results.

```
# Return joint genotypes and probabilities
gens = get.likely.genotypes(hypD,paramsD,results$Def,joint=TRUE)

# Return joint genotypes with per-locus probability greater than 3%
gens = get.likely.genotypes(hypD,paramsD,results$Def,joint=TRUE,prob=0.03)
```

The returned object here is organised similarly to the single-contributor object, with the dependence on contributor removed from the organising heirarchy, as shown in Figure 2.

If there are three unprofiled contributors to the CSP, the function will return either a genotype list for each contributor if `joint=FALSE`, or a genotype list with six columns (two alleles for each contributor) if `joint=TRUE`. If there is only a single contributor to the CSP, the results will be identical regardless of the value of `joint`, although they will be diplayed slightly differently (if `joint=FALSE` the dependence on contributor will still be displayed, but as there is only one contributor this has no real effect).

```
[[1]]
[[1]][[1]]
[[1]][[1]]$D2
[[1]][[1]]$D2$genotypes
[1] "17" "20"

[[1]][[1]]$D2$probabilities
[1] 0.9460538


[[1]][[1]]$D21
[[1]][[1]]$D21$genotypes
[1] "29"   "32.2"

[[1]][[1]]$D21$probabilities
[1] 0.8806254


[[1]][[1]]$TH01
[[1]][[1]]$TH01$genotypes
     [,1] [,2]
[1,] "6"  "9.3"
[2,] "8"  "9.3"
[3,] "6"  "8"

[[1]][[1]]$TH01$probabilities
[1] 0.3113742 0.3109442 0.2923899



[[1]][[2]]
[[1]][[2]]$D2
[[1]][[2]]$D2$genotypes
     [,1] [,2]
[1,] "17" "24"
[2,] "20" "24"

[[1]][[2]]$D2$probabilities
[1] 0.3745314 0.2656215


[[1]][[2]]$D21
[[1]][[2]]$D21$genotypes
     [,1] [,2]
[1,] "29" "31"
[2,] "31" "32.2"

[[1]][[2]]$D21$probabilities
[1] 0.5306968 0.2493529


[[1]][[2]]$TH01
[[1]][[2]]$TH01$genotypes
     [,1]  [,2]
[1,] "6"   "9.3"
[2,] "8"   "9.3"
[3,] "6"   "8"
[4,] "9.3" "9.3"

[[1]][[2]]$TH01$probabilities
[1] 0.2776487 0.2232152 0.2017299 0.1318768



$topGenotypes
$topGenotypes$genotypes
$topGenotypes$genotypes[[1]]
     [,1] [,2]
D2   "17" "20"
D21  "29" "32.2"
TH01 "6"  "9.3"

$topGenotypes$genotypes[[2]]
     [,1] [,2]
D2   "17" "24"
D21  "29" "31"
TH01 "6"  "9.3"


$topGenotypes$probabilities
$topGenotypes$probabilities[[1]]
[1] 0.2594117

$topGenotypes$probabilities[[2]]
[1] 0.05518619
```

Figure 1: An example output of `get.likely.genotypes` for a two-contributor, three-locus CSP, obtaining marginal genotype probabilities. The first section of the results shows the single locus genotypes for single contributors and is broken into two subsections, one for each contributor subject to dropout (designated with `[[1]][[1]]` for the first contributor, and `[[1]][[2]]` for the second contributor). Each of these subsections is then further divided into locus sections (designated `$locusName`), which are then each split into genotypes and probabilities for that locus and that contributor. The probabilities correspond to the rows of the genotypes matrices. The second section (designated `$topGenotypes`) shows the most probable whole-profile genotype for each contributor, and is split into genotypes and probabilities subsections, which are both further subdivided into contributors subject to dropout (e.g. `$topGenotypes$probabilities[[2]]` indicates the probability of the most probable genotype for the second contributor, which is displayed in `$topGenotypes$genotypes[[2]]`).

```
$joint
$joint$D2
$joint$D2$genotypes
     [,1] [,2] [,3] [,4]
[1,] "17" "24" "17" "20"
[2,] "20" "24" "17" "20"
[3,] "24" "25" "17" "20"
[4,] "23" "24" "17" "20"
[5,] "19" "24" "17" "20"

$joint$D2$probabilities
[1] 0.35124052 0.23496616 0.07399688 0.07061620 0.06122807


$joint$D21
$joint$D21$genotypes
     [,1] [,2]   [,3] [,4]
[1,] "29" "31"   "29" "32.2"
[2,] "31" "32.2" "29" "32.2"
[3,] "30" "31"   "29" "32.2"
[4,] "31" "32.2" "29" "29"

$joint$D21$probabilities
[1] 0.49921332 0.16146173 0.08830646 0.08789113


$joint$TH01
$joint$TH01$genotypes
     [,1]  [,2]  [,3] [,4]
[1,] "6"   "9.3" "8"  "9.3"
[2,] "8"   "9.3" "6"  "9.3"
[3,] "9.3" "9.3" "6"  "8"
[4,] "6"   "8"   "6"  "9.3"
[5,] "6"   "9.3" "6"  "8"
[6,] "6"   "6"   "8"  "9.3"

$joint$TH01$probabilities
[1] 0.15636400 0.15062049 0.13187676 0.11295039 0.10912104 0.09935616


$topGenotypes
$topGenotypes$genotype
     [,1] [,2]  [,3] [,4]
D2   "17" "24"  "17" "20"
D21  "29" "31"  "29" "32.2"
TH01 "6"  "9.3" "8"  "9.3"

$topGenotypes$probability
[1] 0.02741748
```

Figure 2: An example output of `get.likely.genotypes` for a two-contributor CSP, obtaining joint genotype probabilities. The first section (`$joint`) displays the joint genotype probabilities at each locus for those genotypes with probability greater than `prob` (`$joint$locusName$genotypes`), as well as their associated probabilities (`$joint$locusName$probabilities`). Once again the probabilities correspond to rows in the genotypes matrices. The second section (`$topGenotypes`) displays the most likely whole-profile joint genotype (`$topGenotypes$genotype`) as well as its associated probability (`$topGenotypes$probability`).

# 2 Overview and description of the model

## 2.1 Overview

`likeLTD` is an R package for evaluating likelihoods for a crime scene DNA profile (CSP) given the reference profiles of possible contributors of DNA and a specified number of unprofiled contributors. It is particularly appropriate for low template DNA (LTDNA) profiles in which allelic dropout is considered possible for some or all contributors. `likeLTD` is described, and results of some illustrative analyses presented, in Balding (2013) (much, but not all, of the material in that paper is also found here). It is a development of earlier algorithms described in Balding and Buckleton (2009).

We consider a single crime stain which may have been profiled multiple times from replicate PCRs of the original sample. Forensic DNA profiling predominantly assays autosomal short tandem repeat (STR) loci, using technology in which an allele in the profiled sample is represented by a peak in an electropherogram (epg). See Butler (2010) for background on forensic DNA profiling, and Buckleton et al. (2004); Balding (2005) for introductions to statistical methods for evaluating DNA profile evidence. (We plan a new edition of Balding (2005) giving more attention to low-template profiles, to appear early in 2014.)

### 2.1.1 The contributors of DNA

It is assumed that we are interested in comparing the likelihood given that a profiled individual Q as a contributor with the likelihood when Q is replaced with an unprofiled individual X. The ratio of those two likelihoods, each maximised over the nuisance parameters, is the likelihood ratio (LR). There can be up to two further unprofiled possible contributors of DNA, U1 and U2, and multiple profiled uncontested contributors (K1, K2, ...). The U are always assumed to be subject to dropout, but if all alleles of Q or any of the K are observed in every replicate, then the dropout rate for that individual is fixed at zero.

There can be several LRs of interest, considering X of different ethnicities and different relatedness with Q (the more genetically similar X is to Q, the smaller the LR). `likeLTD` allows X to be related to Q with two relatedness coefficients. In addition, we use an $F_{ST}$ adjustment to allele fractions that allows for possible remote shared ancestry of Q with X. Within `likeLTD`, this adjustment only affects the alleles of Q and does not take into account any other profiled contributors. We assume U1 and U2 to be mutually unrelated, and they and the K are all assumed unrelated to X.

Because the relatedness coefficients and $F_{ST}$ account for the positive correlations across

loci due to shared ancestry of Q and X, it is reasonable to compute full-profile LRs by multiplication of single-locus LRs, which is standard practice in the assessment of DNA profile evidence (Buckleton et al., 2004). We thus focus below on the single-locus case.

### 2.1.2 The parameters

Some parameters are defined in terms of a reference individual who is X under the defence hypothesis. Under the prosecution hypothesis, the reference individual is Q if Q is subject to dropout, otherwise the first K subject to dropout, if any, otherwise the first U, if any. If there are no contributors subject to dropout, then there is no need for a reference individual – the parameters defined in terms of this individual are not used.

The "nuisance" parameters, which must be eliminated under each multi-locus likelihood before taking their ratio, are

- the dropout rates (one per replicate) for the reference individual;
- the contributions of DNA, relative to that from the reference individual (one parameter for each contributor subject to dropout other than the reference individual); the relative contribution from an individual is used to compute dropout rates – see below;
- the parameters of the dropout model: locus adjustment (one per locus), power parameter (one), and degradation parameters (one for each contributor subject to dropout);
- a dropin parameter (optional, see below).

`likeLTD` maximises a (penalised) likelihood over these parameters using the `R DEoptim` function.

### 2.1.3 Key features of `likeLTD`

Some key features of `likeLTD`:

- It can accept "uncertain" allele calls, in addition to present/absent, which mitigates the "cliff edge" effect of calls that are restricted to present/absent.
- It combines information across all DNA profiling runs, thus avoiding the need for a "consensus" profile (Gill et al., 2000).
- It allows a different dropout rate for each contributor in each profiling runs.

- The dropout probability for a given dose of DNA, relative to unit dose (for the reference individual), uses the model of (Tvedebrink et al., 2009).
- Dropout rates can increase with fragment length, based on the model of Tvedebrink et al. (2012b).
- As a consequence of estimating the dropout rate for contributors, a potential contributor can be considered in an hypotheses without implying that their DNA is present, because the contribution of DNA from that individual can be estimated at zero.
- Because the penalised likelihoods are maximised over the nuisance parameters, combining information over alleles, loci, replicates and individuals, there is little need for external calibration data. This is only required for a few hyperparameters – the parameters of the penalty functions and a rate parameter of the dropout model (for which we use results from Tvedebrink et al. (2009), see below). The underlying parameters are allowed flexibility to best fit the CSP data under each hypothesis, constrained by penalty functions that depend on these hyperparameters.
- `likeLTD` does not use peak height information directly; it is used indirectly by the forensic scientist when making the present/uncertain/absent designations. Peak heights would provide more information and hence greater statistical efficiency for many CSPs, but typically require extensive calibration data specific to the profiling protocol used for the CSP. Because DNA evidence is so powerful, statistical efficiency is not usually an urgent priority; robustness is usually more important and the results below show that `likeLTD` has good robustness properties. In the presence of multiple replicates there is often very little loss of information from using present/uncertain/absent rather than peak heights, but if only a single profiling run is available the loss of information can be substantial.

## 2.2 Single-locus LR with dropout

Consider first a single profiling run in which the CSP showed two alleles, A and B. If the contributors under the competing hypotheses are

$$H_p^1\text{: Q} \qquad \text{and} \qquad H_d^1\text{: X},$$

and Q ≡ AB, where "≡" denotes "has genotype", then the LR under usual assumptions (Buckleton et al., 2004; Balding, 2005) is

$$\text{Q} \equiv \text{AB, CSP} = \text{AB:} \qquad \text{LR} = \frac{(1+F_{ST})(1+2F_{ST})}{2(F_{ST}+(1-F_{ST})p_A)(F_{ST}+(1-F_{ST})p_B)} \tag{1}$$

where the $p$ are population allele fractions. We henceforth assume that sampling and $F_{ST}$ adjustments have been made to the $p$ as described below, so that we can ignore $F_{ST}$ and the LR simplifies to $1/(2p_Ap_B)$.

Dropout refers to any allele of a hypothesized contributor that is not observed in the CSP. If CSP = A, and low epg peak heights suggest that dropout is possible, then under a standard model (Balding and Buckleton, 2009; Gill et al., 2012) the LR can be written as

$$\text{Q} \equiv \text{AB, CSP} = \text{A:} \qquad \text{LR} = \frac{D(1-D)}{p_A^2(1-D_2) + 2p_A(1-p_A)D(1-D)} \tag{2}$$

where $D$ denotes the probability of dropout for a heterozygote allele, while $D_2$ denotes the probability of a homozygote dropout. The numerator is the probability that the B allele of Q has dropped out ($D$), while the A has not ($1-D$). In the denominator, either X is AA and there has been no dropout (1st term), or (2nd term) X is heterozygous but the non-A allele has dropped out.

Logically, $D$ in the numerator of the LR is different from $D$ in the denominator. However, typically a similar range for $D$ is supported under both hypotheses and they are often taken to be equal for illustrative calculations (Gill et al., 2007).

## 2.3 Effect of an "uncertain" allele designation

If now we assume CSP = A[B], where [] denotes an "uncertain" allele designation, while again Q ≡ AB, then the LR becomes

$$\text{CSP} = \text{A[B]:} \qquad \text{LR} = \frac{1-D}{p_A^2(1-D_2) + 2p_Ap_B(1-D) + 2p_A(1-p_A-p_B)D(1-D)}. \tag{3}$$

In the numerator we only know that Q's A allele has not dropped out ($1-D$), whereas we do not know if B has dropped out and so the corresponding term is one. In the denominator, the three terms correspond to X ≡ AA, AB and AZ, where Z is any allele other than A or B.

Figure 3 (solid curves) shows the LRs (1) through (3) as a function of dropout rate $D$ for a locus with $p_A = p_B = 0.1$ (after adjustments). The computation of $D_2$ from $D$ is discussed in Section 2.5. As expected, the LR for CSP = A[B] (red curve) is always intermediate between those for CSP = AB (black) and CSP = A (green). When $D$ is high the red and green curves are similar, because in the presence of high dropout both an uncertain and an absent designation for B convey little information about whether or not X has a B allele. However when $D$ is small the two LRs differ substantially as CSP = A is inconsistent with X ≡ AB, whereas CSP = A[B] is consistent with both X ≡ AA and X ≡ AB.

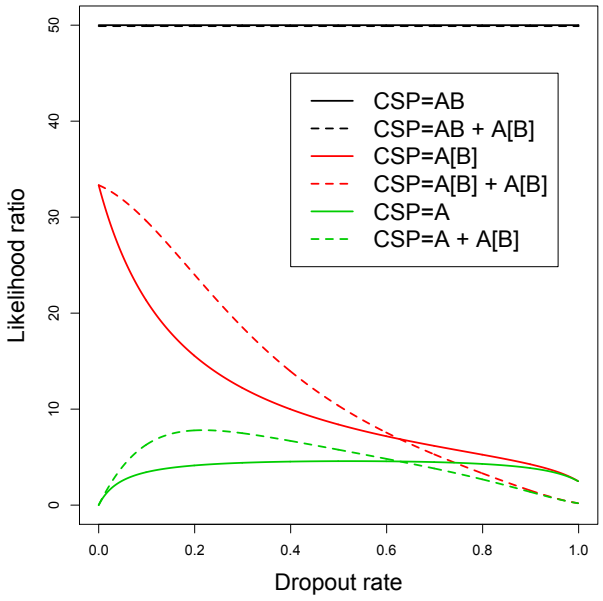



Figure 3: **Single-locus, single-contributor LRs for three CSPs with 1 profiling run (solid curves) and three with 2 runs (dashed curves).** In the legend box, + separates the two runs and [ ] denotes an uncertain allele call. Allele A is observed in every case, whereas the designation of allele B in the first run varies over present, uncertain, and absent; it is uncertain in the second run. The LRs are expressed as a function of the dropout rate $D$, assumed to be the same for all alleles. For purposes of illustration, $D$ has the same value in numerator and denominator.

Next consider the LRs when the three CSPs considered above constitute only the first run, and there is a second run that gives A[B] in each case (Figure 3 dashed curves). We assume the same $D$ for both runs. When CSP = AB + A[B], we must have X ≡ AB (we ignore dropin here, see Section 2.7) and the LR is again (1). The LRs in the other two cases are

$$\text{CSP} = \text{A} + \text{A[B]}: \quad \text{LR} = \frac{D(1-D)^2}{p_A^2(1-D_2)^2 + 2p_Ap_BD(1-D)^2 + 2p_A(1-p_A-p_B)D^2(1-D)^2},$$

$$\text{CSP} = \text{A[B]} + \text{A[B]}: \quad \text{LR} = \frac{(1-D)^2}{p_A^2(1-D_2)^2 + 2p_Ap_B(1-D)^2 + 2p_A(1-p_A-p_B)D^2(1-D)^2}.$$

Note that we assume that the different runs are independent, conditional on the genotypes of all contributors (Curran et al., 2005).

We see from Figure 3 that observing A[B] in the 2nd run increases both LRs when $D$ is small but decreases them when $D$ is large. In fact, when $D$ is very high, observing either A or A[B] in just one run yields LR > 1, favouring $H_p^1$, whereas two such observations in independent runs gives LR < 1, against $H_p^1$. This is because X ≡ AA under $H_d^1$ then provides a better explanation of the replicate observations than $H_p^1$, since homozygotes are much less likely to drop out than a heterozygote allele.

## 2.4 Additional contributors

LRs such as (2) and (3) can be rewritten more generally as

$$\text{LR} = \frac{P(\text{CSP}|\text{Q} \equiv \text{AB})}{\sum_{g\in\Gamma} p_g P(\text{CSP}|\text{X} \equiv g)} \tag{4}$$

where $\Gamma$ denotes the set of possible genotypes, while $p_g$ denotes the population fraction of genotype $g$. Equation (4) makes explicit the requirement to sum over all possible genotypes for the unprofiled contributor X. When there is an additional unprofiled contributor U1, we proceed in the same way as for X. We sum over all possibilities for each unknown genotype, multiplying each term by the genotype probability:

$$\mathrm{LR} = \frac{\sum_{g\in\Gamma} P(\mathrm{CSP}|\mathrm{Q}\equiv \mathrm{AB}, \mathrm{U1}\equiv g)}{\sum_{g1,g2\in\Gamma} p_{g1}p_{g2}P(\mathrm{CSP}|\mathrm{X}\equiv g1, \mathrm{U1}\equiv g2)}. \tag{5}$$

Each term in these sums follows the same well-established and simple rules that have been used for Q and X above, now applied additionally to the current genotype for U1.

Individuals contribute different amounts of DNA to the mixed-source sample, and multiple individuals can have one or two copies of a given allele. Thus we need to model the dropout probability at an allelic position as a function of the total amount of DNA from all contributors with that allele.

## 2.5 Multi-dose dropout model

At each step in the likelihood calculation, there is a set of individual hypothesized contributors of DNA to the CSP and each has a genotype (tentatively assigned for the unprofiled contributors). Dropout refers to any allele of these hypothesized contributors that is not observed in the CSP. The dropout model used here (Tvedebrink et al., 2009) can be written

$$\frac{D(k)}{1-D(k)} = (\alpha_s k)^{\beta}, \tag{6}$$

where $s$ indicates the locus. Note that $\beta$ is $\beta_1$ in the notation of Tvedebrink et al. (2009), and $\alpha_s$ is proportional to $\exp(\beta_{0,s}/\beta_1)$. Here we choose the scale by fixing the mean over loci of $\alpha_s$ at one. The estimates of $\beta_{0,s}$ obtained by Tvedebrink et al. (2009), from experimental non-degraded LTDNA profiled at the ten loci of the SGM+ system, imply an SD for $\alpha_s$ of 0.141. Because they may depend sensitively on the experimental protocol employed, we do not use the Tvedebrink et al. (2009) estimates directly, but instead estimate the $\alpha_s$ under each hypothesis from the observed CSP. To keep the estimates realistic, we impose a gamma distribution prior on the $\alpha_s$, here assigning mean = 1 and SD = 0.141, although a different SD may be appropriate for example in highly-degraded samples. For the global parameter $\beta$, we adopt a normal distribution with mean $-4.35$ and SD 0.38. These both derive from the work of Tvedebrink et al. (2009), but the SD value was not published in the paper and was supplied in personal correspondence with Dr Tvedebrink.

Figure 4 illustrates $D(k)$ as a function of $D(1)$ for several values of $k$, evaluated by substituting $\alpha_s^{\beta} = D(1)/(1-D(1))$ in (6). We see for example that if $D(1) = 0.5$, then

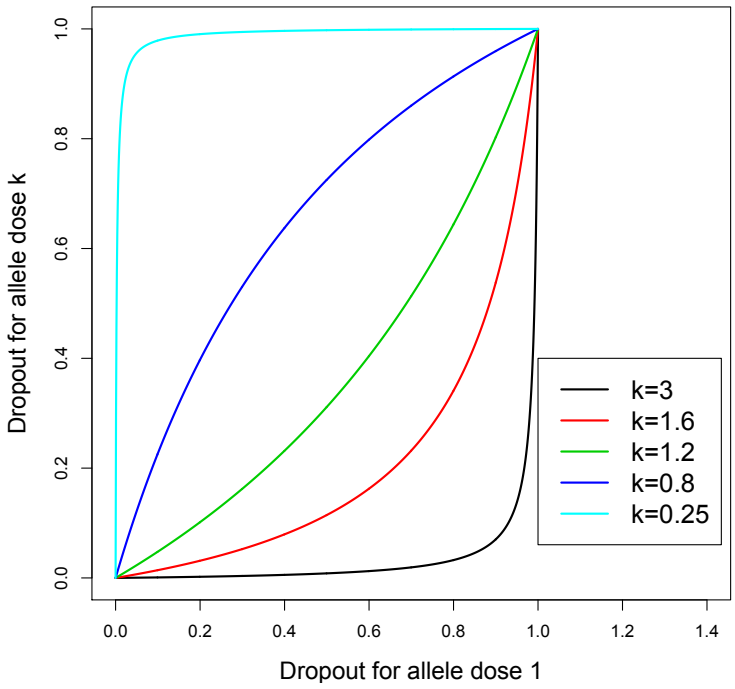



Figure 4: **Dropout probabilities for dose $k$ of DNA ($y$ axis) against those for a unit dose ($x$ axis).** The values of $k$ are shown in the legend box. For this figure the power parameter $\beta$ was fixed at $-4.35$.

$D(1.2) \approx 0.3$ and $D(0.8) \approx 0.7$, and so just a 20% change in DNA dose can have a large impact on dropout probabilities.

We take $k = 1$ to correspond to a single heterozygote allele of the reference individual (see Section 2.1.2). Thus $D(1)$ is the heterozygote dropout probability, and $D(2)$ is the homozygote dropout probability, for the reference individual. For $k$ large

$$\frac{D(2k)}{D(k)^2} \approx \left(\frac{2}{\alpha_s k}\right)^{\beta} > 1,$$

so a homozygote dropout can be *more* likely than independent dropout of both alleles, which is implausible. However, this inequality only applies for low dropout probabilities (Tvedebrink et al., 2012a) and so this defect of the model is unimportant in practice.

The problem of calculating likelihoods for LTDNA profiles was not addressed by Tvedebrink et al. (2009); they validated their model by comparing theoretical and empirical dropout rates. To achieve this, they estimated the amount of DNA from each contributor using the heights of peaks due only to that contributor over the whole profile. This is problematic for calculating LTDNA likelihoods, because it ignores information from allele peaks with multiple contributors, and requires alleles of individual contributors to be distinguished, which is frequently not possible. Here, we directly specify the likelihood for each replicate in terms of $D(k)$ at every allelic position, with $k$ calculated according to the contributions of DNA and the genotypes of all the hypothesised contributors. We thus use present/uncertain/absent information at every allelic position to provide information about amounts of DNA.

## 2.6 Degradation model

DNA degrades over time, at a rate that depends on temperature, humidity and environmental exposure. In forensic DNA profiling, degradation is manifested in higher dropout rates for alleles with large fragment lengths. Our model for the effect of degradation is based on that of Tvedebrink et al. (2012b), who posited a geometric distribution for the effective amount of DNA as a function of allele fragment length. Thus, the average allele dose $k$ from the $i$th contributor subject to dropout is modified at an allele with fragment length $l$ base pairs (centred to have mean zero) according to

$$k' = k(1+\gamma_i)^{-l},$$

where $\gamma_i > 0$. Shorter fragments ($l < 0$) correspond in effect to an enhanced allele dose, while longer fragments generate a smaller effective allele dose. An STR allele consists of flanking regions in addition to the tandem repeats, and so the repeat number that characterises the allele is not a good proxy for fragment length, which can be obtained for many DNA profiling systems at `www.cstl.nist.gov/div831/strbase/`.

In the spirit of shrinkage regression methods, `likeLTD` incorporates a weak penalty (exponential, mean 0.02) on each $\gamma_i$. An example below (Section 3.6) illustrates the effect of this penalty, which is a slight tendency to shrink the parameter estimates towards zero, which is usually negligible but avoids inflated values when there is very little information (for example, in the presence of high dropout or substantial masking).

## 2.7 Dropin

Dropin refers to an allele in the CSP that is not included in the genotype of any hypothesized contributor, profiled or unprofiled. Dropin alleles can arise from individuals contributing a very low level of DNA to the sample, perhaps generated from fragments of the DNA molecule that persist for some time after the death and decay of a cell. Forensic scientists often restrict "dropin" to lab-based contamination which can be measured by control runs and is usually found to be rare. However, we cannot usually determine the source of any dropin (Gill et al., 2000), so for the purposes of evaluation we do not distinguish distinguish dropin according to its source. It follows that lab-based estimates of drop-in rates are of limited usefulness.

Each dropin allele does come from an individual, but it may arise from very few and possibly degraded cells so that very little of this individual's DNA is reflected in CSP peaks. It is computationally inefficient to sum over all possible genotypes, as in (5), for such low-level contributors, and so we allow the possibility of modelling dropin more simply, as independent Bernoulli trials (Curran et al., 2005). Dropin is non-dropout of an

allele of a low-level contributor, and so we model the dropin probability as a constant ($c$) times the non-dropout rate for each replicate. As for the $\gamma_i$, we impose a weak penalty on $c$ (exponential, mean 0.5) to discourage solutions with $c$ large, reflecting background information that dropin is usually rare.

If there are sufficient unprofiled contributors to explain all observed alleles, it is unnecessary to model dropin explicitly. If dropin is not modelled, any proposed allocation of alleles to the unprofiled contributors that does not explain every CSP allele is impossible, and the corresponding likelihood calculations can be avoided. We are unable to model the effect of degradation on dropin probability, because dropins are too rare to estimate $\gamma_i$, so it is preferable to avoid use of the dropin model when possible. We suggest that the dropin model be used when there are one or two unreplicated alleles in a 10-locus profile that cannot be accounted for by the hypothesized contributors.

## 2.8 Maximising the penalised likelihood

To compute the LR, it is necessary to deal with the "nuisance" parameters under each hypothesis. These are: the $D(1)$ (one per replicate), the dropout model parameters $\alpha_s$ (one per locus) and $\beta$, the contributions of DNA, relative to the reference individual, and the $\gamma_i$ (one of each for every contributor subject to dropout), and possibly a dropin parameter $c$ (see above). `likeLTD` seeks to maximise a penalised likelihood over these parameters, with penalties on $\alpha_s$, $\beta$, $\gamma_i$ and $c$ as described above. These penalties can be thought of as prior distributions, but we do not use a Bayesian approach since we maximise over unknown parameters rather than integrate. The primary purpose of the penalty is to avoid the maximisation algorithm from exploring unrealistic regions of the parameter space.

We use the `R DEoptim` function to maximise the penalised likelihoods, which is a genetic algorithm utilising differential evolution optimisation to find the global minimum of a function (since we want to maximise and not minimise, it is necessary to multiply the log-likelihoods by $-1$).

The results from `DEoptim` consist of two lists, `DEoptim` and `member`. `DEoptim` consists of four parts:

`bestmem`: The parameter values that gave the maximum likelihood.

`bestval`: The negative log likelihood at these parameter values.

`nfeval`: The number of evaluations of the objective function that were carried out during optimisation.

`iter`: The number of generations for the optimation.

while `member` consists of:

`lower`: The lower bounds of the parameters used.

`upper`: The upper bounds of the parameters used.

`bestvalit`: A vector containing the maximum likelihood at each generation.

`bestmemit`: A matrix containing the parameters that gave the maximum likelihood at each generation.

`pop`: The set of parameter values generated for the last generation.

`storepop`: The sets of parameters generated for previous generations.

See the `DEoptim` help page for more information.

Rather than estimate the nuisance parameters, they can be set to fixed values if desired, by setting the upper and lower bounds. For example

```
tofix = "dropout"
value = 0.2
# Create index of which parameters are dropout parameters
index = grep(tofix,names(paramsP$upper))
# Set those parameters upper value to the fixed value for prosecution
paramsP$upper[index] = rep(value,times=length(index))
# Set those parameters lower value to the fixed value for prosecution
paramsP$lower[index] = rep(value,times=length(index))

# Repeat for the defence uppers and lowers
index = grep(tofix,names(paramsD$upper))
paramsD$upper[index] = rep(value,times=length(index))
paramsD$lower[index] = rep(value,times=length(index))
```

## 2.9 Computing time and memory requirements

The Hammer case analysis described above required 146 minutes to run, while the three contributor test (see Section 3.4) requires 350 minutes to run. These timings were made on a desktop computer with 15Gb of RAM, and an eight core Intel i7 processor (at 3.1GHz per core). Computing times may vary across machines, and in particular the memory requirement may become important for machines with less RAM. Most desktop computers will have enough memory to run cases with up to two unknown contributors

(the defence case, with three unknowns, uses between 100 & 200 Mb of memory), but if dropin is also modelled this can require in the region of 8Gb of RAM, and will therefore only be possible on large-memory machines.

As discussed previously the major determinant of runtime is now how many chunks `enumerate` runs for convergence, $n$. This is computed as $n = 4\lceil \log_2(\overline{x}) \rceil + r_d$ where $r_d$ is the number of `rcont` parameters handed to DEoptim under the defence case, and $\overline{x}$ is the mean of the set $\{\sigma_p, \sigma_d\}$ where $\sigma_p$ is the standard deviation of the first phase (iterations 1-75) of the first chunk of optimisation for the prosecution ($\sigma_d$ is the same, but for defence). A negative $n$ may be obtained if $\overline{x} < 1$, in which case $\overline{x}$ is set to 1.5 so $\lceil \log_2(\overline{x}) \rceil = 1$. We can see that the minumum number of chunks is 4, and that the number of chunks increases with both complexity and difficulty of optimisation.

The number of generations used for optimisation by `DEoptim` is not fixed in advance, but is determined by the function `evaluate`. This function determines a number of "chunks" to run, based on the standard deviation of the first chunk. Each chunk is a separate optimisation, with a geometric pattern in the crossing over rate (see `DEoptim::DEoptim.control`) and convergence tolerance. Each chunk checks for convergence after every 75 generations; convergence occurs if $\log_{10}(|1 - L_c/L_{c-75}|) \leq t$ where $L$ denotes likelihood, $c$ the current generation and $t$ the tolerance for the given chunk.

A key parameter determining the `DEoptim` run time is the size of the population (number of random starts per generation). Larger values tend to generate higher likelihoods, but there is a rapidly diminishing benefit with increasing population size, while run time increases approximately linearly. `likeLTD` sets the population size to be four times the number of parameters varied in the optimisation (NP). If close relatedness is invoked, a further incresase in population size is implemented. For the Hammer timings above this implies a population size of 80, while for the three contributor test (Section 3.4) the population size was 72. NP, and hence population size and run time, depend on the number of contributors subject to dropout, since there is an effective amount of DNA and a degradation parameter for each of these. Run time is also affected by the number of unprofiled contributors: these are always subject to dropout, and summation over their possible genotypes adds to run time and to the memory requirement. Below we show that `likeLTD` can handle up to four contributors subject to dropout, of which up to two can be unprofiled.

# 3 Performance tests of `likeLTD`

## 3.1 Validation of likelihood ratios

Laboratory procedures to measure a physical quantity such as a concentration can be validated by showing that the measured concentration consistently lies within an acceptable range of error relative to the true concentration. Such validation is infeasible for software aimed at computing an LR, because there is no underlying true value (no equivalent of the true concentration). The underlying truth that is being probed is whether or not an individual Q is a contributor of DNA to the crime scene profile (CSP), which is a yes/no and not a quantitative event. Even if I know all the contributors to a complex mixture there is still no "correct" value of the LR. Probabilities measure our certainty about the true state of nature, and there is no "true" level of certainty about an unknown fact, it depends on the data available (for example details of the CSP) and on modelling assumptions, e.g. about dropout and dropin.

For a given set of input data and modelling assumptions, there may be an exact value for the likelihood, but there are always other plausible modelling assumptions that can give different values. In practice, the different programs that are available or in development for the evaluation of LTDNA profiles do make different assumptions and obtain different answers, though of course the differences should be, and in practice are, small relative to the 10 or more orders of magnitude over which LRs range in practical applications. It follows that it is fruitless to insist on very high precision of likelihood calculations under any specific model. It was recognised by Turing over a half-century ago that evidence is best measured in units of base 10 logarithms, a unit that he called the ban. An error of 1 deciban ($\approx 26\%$ on the natural scale) should be regarded as negligible, because the implications of different, reasonable, modelling assumptions are often larger than this, and so bans should not be reported to one decimal place. Moreover, we show below that routine sampling and $F_{ST}$ adjustments can make a difference of between one and two bans, and these are designed to err substantially in favour of defendants, typically by at least several decibans.

There is a convenient analogy between the measurement of WoE and earthquakes. Both are logarithmic scales that should usually be reported to at most one decimal place, and typical values range up to about 10. There is no maximum for WoE but in practice in the UK any value above 9 bans (LR $>$ 1 billion) is reported as "over a billion".

While there is no "true" value of the LR, even for simulated cases, many checks on the performance of `likeLTD` are possible, using simulated and real profiles. We report below the performance of `likeLTD` under an extensive set of tests. For example, we check if an allele in the reference profile of Q is removed from the CSP, then the LR is reduced,

whereas if an allele not in Q's reference profile is removed this increases the resulting LR. We check that if we wrongly assume an extra contributor, either profiled or unprofiled, the LR computed by `likeLTD` is little changed, and similarly if a small number of alleles is modelled ether using a dropin term in the model, or as the alleles of an additional contributor with heavy dropout. We check that when large fragment length alleles are removed from the CSP, the degradation parameter estimate is high. When different numbers of dropouts occur in different replicates, we show that the estimates of the dropout rates for those replicates differ accordingly. We also check that `likeLTD` produces answers consistent with those generated by a similar program, in that any discrepancies can be qualitatively understood in terms of the different modelling assumptions of the programs. We further provide simulations to illustrate the search length required for stable estimates of the LR over replicate runs of `likeLTD`.

## 3.2 A one-contributor CSP with dropout and dropin

Three CSPs were analysed (Table 1(i)), to which Q (Table 1(i), row 1) is the only true contributor, but both dropout and dropin are introduced for CSPs 2 and 3. Thus $H_p^1$ (page 14) is the truth, and in the analyses below this is compared with $H_d^1$.

**CSP1:** Exactly the profile of Q (Table 1, row 1) appears in each replicate of CSP1 (rows 2,3). There would be no need to use `likeLTD` for this perfect single-contributor match, but we apply it here in order to check that `likeLTD` works correctly in this setting. The WoE (Table 1, row 1) is 11.3 bans. A naive application of the product rule using the allele counts in Table 1 gives 12.9 bans (Table 2), but using the allele fractions in the bottom two rows of the table, adjusted for coancestry ($F_{ST} = 0.02$) and sampling (adj = 1), reduces this to 11.43 bans.

The sampling adjustment "adj" is a value added to the database counts for the alleles of Q, to reduce the risk of understating the population frequency, particularly for rare alleles. Except in the first column of Table 2, we always use adj = 1. Balding (1995) advocated adj = 2 in the absence of an $F_{ST}$ adjustment, but since $F_{ST}$ has a big impact on low-frequency alleles, adj = 1 is adequate when an appropriate $F_{ST}$ adjustment is employed. Note that `likeLTD` only adjusts for coancestry between X and Q, by replacing each population allele fraction estimate $p$ (after sampling adjustment) with:

$$\begin{array}{rl} (1-F_{ST})p/(1+F_{ST}) & \text{for alleles not in the profile of Q} \\ (F_{ST}+(1-F_{ST})p)/(1+F_{ST}) & \text{for a heterozygote allele of Q} \\ (2F_{ST}+(1-F_{ST})p)/(1+F_{ST}) & \text{for a homozygote allele of Q} \end{array}$$

$F_{ST}$ values comparing different parts of mainland Europe are typically $< 0.005$, and so 0.02 is a relatively large value, except perhaps for Q and X both from a small population

(i)

| Locus | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|
| Q | 15,16 | 15,18 | 11,11 | 17,24 | 13,13 | 29,31 | 10,16 | 13,14 | 6,9.3 | 22,24 |
| CSP1a | 15,16 | 15,18 | 11 | 17,24 | 13 | 29,31 | 10,16 | 13,14 | 6,9.3 | 22,24 |
| CSP1b | 15,16 | 15,18 | 11 | 17,24 | 13 | 29,31 | 10,16 | 13,14 | 6,9.3 | 22,24 |
| CSP2a | 15,16 | 15,18 | 11 | 17,24 | 13 | 29,31 | 10,16 | 13,14 | 6,9.3 | 22 |
| CSP2b | 15,16,17 | 15,18 | 11 | 17 | 13 | 29,31 | 10,16 | 13,14 | 6,9.3 | 22,24 |
| CSP3a | 15,16 | 15,18 | 11 | 17,24 | 13,15 | 29,31 | 10,16 | 13,14 | 6,9.3 | 22 |
| CSP3b | 15,16,17 | 15,18 | 11 | 17 | 13 | 29,31 | 16 | 13,14 | 6,9.3 | 22 |
| count | 116,106 (396) | 38,80 (384) | 130 (418) | 90,35 (388) | 118 (384) | 74,28 (384) | 2,56 (384) | 102,133 (396) | 77,151 (384) | 75,47 (384) |
| fraction | 0.302,0.278 | 0.117,0.221 | 0.341 | 0.244,0.108 | 0.338 | 0.206,0.0918 | 0.027,0.161 | 0.268,0.343 | 0.214,0.398 | 0.209,0.139 |

(ii)

| | | | | **Contributor 1** | | | **Contributor 2** | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CSP | WoE | Hyp | L | DO a | DO b | *gamma* | DO a | DO b | *gamma* | DI | Power |
| CSP1 | 11.3 | $H_p^1$ | 4.6 | | | | | | | 0.00 | −4.35 |
| | | $H_d^1$ | −6.8 | 0.00 | 0.00 | 0.00 | | | | 0.00 | −4.35 |
| | 11.3 | $H_p^2$ | 4.6 | 1.00 | 1.00 | 0.00 | | | | | −4.35 |
| | | $H_d^2$ | −6.8 | 1.00 | 1.00 | 0.00 | 0.00 | 0.00 | 0.00 | | −4.35 |
| CSP2 | 11.2 | $H_p^1$ | −0.4 | 0.03 | 0.03 | 0.00 | | | | 0.07 | −4.37 |
| | | $H_d^1$ | −11.6 | 0.03 | 0.03 | 0.00 | | | | 0.09 | −4.37 |
| | 11.3 | $H_p^2$ | 0.1 | 0.05 | 0.02 | 0.00 | 1.00 | 0.99 | 0.01 | | −4.37 |
| | | $H_d^2$ | −11.2 | 1.00 | 0.99 | 0.01 | 0.05 | 0.02 | 0.00 | | −4.37 |
| CSP3 | 8.8 | $H_p^1$ | −3.6 | 0.03 | 0.11 | 0.00 | | | | 0.15 | −4.38 |
| | | $H_d^1$ | −12.3 | 0.00 | 0.14 | 0.00 | | | | 0.09 | −4.37 |
| | 9.4 | $H_p^2$ | −2.9 | 0.05 | 0.13 | 0.00 | 0.96 | 0.98 | 0.01 | | −4.40 |
| | | $H_d^2$ | −12.3 | 0.97 | 0.99 | 0.01 | 0.04 | 0.12 | 0.00 | | −4.38 |

Table 1: **One contributor profile simulation experiments.** (i) Observed alleles for three crime scene profiles (CSP), each profiled in duplicate (a and b). Also shown are database counts for the alleles of Q (locus total) and the allele fractions after sampling and $F_{ST}$ adjustments (adj = 1, $F_{ST} = 0.02$). (ii) Weight of evidence (WoE) and corresponding parameter estimates, from `likeLTD` analyses of the three CSPs. DO and DI denote dropout and dropin, while gamma and Power are degradation/dropout parameters described in the text as $\gamma$ and $\beta$. The labelling of Contributors 1 & 2 is arbitrary. Under $H_p^1$ there is only one contributor who is not subject to dropout, so no dropout or degradation results are shown.

| **Fst** | **0.00** | **0.00** | **0.01** | **0.02** | **0.03** |
|---|---|---|---|---|---|
| adj | 0 | 1 | 1 | 1 | 1 |
| WoE (bans) | 12.3 | 12.3 | 11.7 | 11.3 | 10.9 |

Table 2: Overall weight of evidence (WoE) for CSP1.

| n. contrib. | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA | WoE |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2.4 | 2.9 | 2.3 | 2.8 | 2.1 | 2.9 | 3.4 | 2.2 | 2.3 | 2.9 | 4.1 |
| 2 | 2.4 | 2.9 | 2.3 | 2.8 | 2.1 | 2.9 | 3.4 | 2.2 | 2.3 | 2.9 | 4.1 |

Table 3: Single-locus LRs and overall weight of evidence for CSP1 when X, the alternative source of the DNA, is a brother of Q. Results are shown for both 1- and 2-contributor analyses.

isolate such as a remote island. $F_{ST} = 0.03$, or in extreme cases $F_{ST} = 0.05$, may be more appropriate for small, isolated or heterogeneous populations, such as many migrant populations. See Balding (2005) for further discussion and for more details of $F_{ST}$ adjustments based on the multinomial-Dirichlet distribution.

*Brother alternative:* Repeating the 1-contributor analysis but now assuming that X is an unprofiled brother of Q gives 4.1 bans (Table 3). The single locus LRs computed by `likeLTD` follow very closely the usual formula for a sibling (Balding, 2005), which in the heterozygote (homozygote) case is:

$$\text{LR} = \frac{4}{1 + p_a + p_b + 2p_a p_b} \qquad \left(\frac{4}{(1+p_a)^2}\right)$$

Note that the single-locus LR can never exceed 4 for the brother alternative. Also, there is no explicit $F_{ST}$ adjustment in the formula above because the allele fractions $p$ have been adjusted as described above: this provides a reasonable approximation to the adjustment based on the multinomial-Dirichlet distribution.

*Assuming two contributors:* If we wrongly guessed that there were two contributors to CSP1, we could compare the hypotheses

$$H_p^2 : \text{Q} + \text{U1} \qquad \text{with} \qquad H_d^2 : \text{X} + \text{U1}$$

where X and U1 denote unprofiled individuals that are unrelated to each other and to Q. In this case `likeLTD` correctly estimates near 100% dropout for U1 under both hypotheses (Table 1(ii), row 2) and the WoE is almost unchanged from assuming one contributor.

Similarly, when X is a brother of Q, the single-locus LRs and overall WoE computed by `likeLTD` are almost identical to the 1-contributor case (Table 3).

**CSP2:** The two replicates of CSP2 differ from those of CSP1 due to one dropin and two dropouts (Table 1(i), rows 4,5). The two dropouts both affect loci with large fragment lengths, consistent with the effects of DNA degradation. Because the dropin allele is at a heterozygous locus where the two alleles of Q are replicated, and the two dropout alleles each appear in the other replicate, the evidence implicating Q remains very powerful and the WoE is only slightly reduced, to 11.2 bans (Table 1, rows 5,6). The dropout parameter estimates are similar over the four replicate/hypothesis combinations. The $\gamma_Q/\gamma_X$ estimates are, as expected, moderately large at around 0%. Since the dropin allele must have come from somebody, it is possible to analyse CSP2 as a two-contributor profile without a dropin term in the model, which implies very heavy dropout for the second contributor, U1. We see from (Table 1(ii), rows 7,8) that this analysis gives a slightly stronger WoE (11.3 bans), closer to the WoE for CSP1. The dropout rates for U1 are, as expected, very high, and $\gamma_{U1}$ is also high (about 1 because the dropin allele is at a short fragment length locus, so when attributed to U1 it gives further support to the pattern of dropout increasing with DNA fragment length.

**CSP3:** One further dropin and two more dropouts have been introduced into CSP3 relative to CSP2 (Table 1(i), rows 6,7). The extra dropin is at a locus for which Q is homozygous, so while it must be a dropin under $H_p^1$, that is not so under $H_d^1$. The two additional dropouts again both affect loci with large fragment lengths, and this time an allele of Q has dropped out in both replicates (FGA 24), which has a substantial impact on the evidence implicating Q as a contributor. While the evidence remains powerful, the overall WoE is now reduced by at least two bans compared with CSP2 (Table 1(ii), rows 9–12). CSP3b is affected by three dropouts, whereas CSP3a by only one, and consequently the dropout rate estimates are much higher for CSP3b than for CSP3a. The difference between modelling CSP3 as a one-contributor profile with two dropins, and a two-contributor profile with substantial dropout for one of the contributors, is now more important (0.6 bans). The two models differ in several respects. The two-contributor model gives the stronger WoE, and the dropin model can be regarded as a simple approximation that reduces computation time, and is more conservative for the examples considered here.

## 3.3 Two unprofiled contributors

Table 5 presents results from `likeLTD` analyses of two-contributor profiles shown in Table 4. The contributors are Q and an individual treated here as unknown, while the

| Locus | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|
| Q | 15,16 | 15,18 | 11,11 | 17,24 | 13,13 | 29,31 | 10,16 | 13,14 | 6,9.3 | 22,24 |
| CSP4a | 15,16,17 | 15,18 | 11,13 | 17,20,24 | 10,13 | 29,31,32.2 | 10,11,13,16 | 13,14 | 6,8,9.3 | 19,22,24,25 |
| CSP4b | 15,16,17 | 15,18 | 11,13 | 17,20,24 | 10,13 | 29,31,32.2 | 10,11,13,16 | 13,14 | 6,8,9.3 | 19,22,24,25 |
| CSP5a | 15,16 | 15,18 | 11,13 | 17,24 | 10,13 | 29,31,32.2 | 10,11,16 | 13,14 | 6,8,9.3 | 19,22,24 |
| CSP5b | 15,16,17 | 15,18 | 11,13 | 17,24 | 13 | 29,31,32.2 | 10,16 | 13,14 | 6,9.3 | 22,24,25 |
| CSP6a | 15,16,17 | 15,18 | 11,13 | 17,20 | 10,13 | 29,31,32.2 | 10,11,13 | 13,14 | 6,8,9.3 | 19,25 |
| CSP6b | 15,16,17 | 15,18 | 11,13 | 17,20,24 | 10,13 | 29,32.2 | 11,13 | 13,14 | 6,8,9.3 | 19,25 |
| CSP7a | 15,17 | 15,18 | 11,13 | 17,20 | 10,13 | 29,31,32.2 | 10,11,13 | 13,14 | 6,8,9.3 | 19,25 |
| uncertain | (16) | (14),(17) | (10) | (16),(19) | (12) | | | (12) | (5) | (18),(24) |
| CSP7b | 15,17 | 15,18 | 11,13 | 17,20,24 | 10,13 | 29,32.2 | 11,13 | 13,14 | 6,8,9.3 | 19,25 |
| uncertain | | (17) | (12) | (19) | | (31.2) | (10) | (12) | (5) | (24) |
| CSP8a | 15 | 15,18 | 11 | 17,20,24 | – | 29,31 | 10,11,16 | 13,14 | 6,8 | 19,22 |
| CSP8b | 16,17 | 15,18 | 13 | 17,20 | 10,13 | 31 | 10,16 | 13 | 6,8 | 22,24,25 |

Table 4: **Two contributor profile simulation experiments.** The crime scene profiles correspond to those of Q and one unprofiled contributor, modified by dropout and (for CSP7) stutter generating uncertain allele calls. The layout is similar to Table 1 except that for CSP7 there are additional rows indicating alleles classified as uncertain.

| | | | | Contributor 1 | | | Contributor 2 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CSP | WoE | Hyp | L | DO a | DO b | *gamma* | DO a.1 | DO b.1 | *gamma*.1 | DI | Power |
| CSP4 | 6.2 | $H_p^2$ | −7.5 | 0.00 | 0.00 | 0 | | | | | −4.35 |
| | | $H_d^2$ | −13.8 | 0.00 | 0.00 | 0 | 0.00 | 0.00 | 0 | | −4.35 |
| CSP5 | 9.4 | $H_p^2$ | −9.9 | 0.51 | 0.69 | 0 | | | | | −4.37 |
| | | $H_d^2$ | −19.4 | 0.54 | 0.72 | 0 | 0.00 | 0.00 | 0 | | −4.37 |
| CSP6 | 5.2 | $H_p^2$ | −11.9 | 0.26 | 0.40 | 0 | 0.00 | 0.00 | 0 | | −4.40 |
| | | $H_d^2$ | −17.1 | 0.67 | 0.77 | 0 | 0.00 | 0.00 | 0 | | −4.39 |
| CSP7 | 4.1 | $H_p^2$ | −11.6 | 0.29 | 0.47 | 0 | 0.00 | 0.00 | 0 | | −4.38 |
| | | $H_d^2$ | −15.7 | 0.65 | 0.71 | 0 | 0.00 | 0.00 | 0 | | −4.37 |
| CSP8 | 2.5 | $H_p^1$ | −23.1 | 0.25 | 0.32 | 0 | | | | 0.93 | −4.22 |
| | | $H_d^1$ | −25.6 | 0.24 | 0.18 | 0 | | | | 0.52 | −4.41 |
| | 4.5 | $H_p^2$ | −21.4 | 0.43 | 0.43 | 0 | 0.63 | 0.63 | 0 | | −4.15 |
| | | $H_d^2$ | −25.9 | 0.65 | 0.65 | 0 | 0.42 | 0.42 | 0 | | −4.29 |

Table 5: Weight of evidence (WoE) and corresponding parameter estimates, from `likeLTD` analyses of the five CSPs of Table 4. See Table1(ii) for explanations.

hypotheses compared are those given above (page 26): $H_p^2$ (the truth) and $H_d^2$. Note that under $H_d^2$ the labels X and U1 for the two unprofiled contributors are arbitrary: in the discussion below we call X the one with dropout rates most similar to Q.

**CSP4:** shows exactly the alleles of the two contributors in both replicates, with no dropout or dropin. The dropout rates, and $\gamma_Q$ and $\gamma_X$, are all correctly estimated to be close to zero. The WoE is 6.2 bans, reduced by over five bans from the case of a single-contributor CSP matching Q (CSP1), because of the additional uncertainty created by the masking effect of the alleles of U1.

**CSP5:** introduces random 50% dropout for the alleles of U1 not shared with Q. Because of the reduced masking effect, the WoE is much higher than for CSP4, now >9 bans. The dropout rates for X are correctly estimated as close to zero. The actual numbers of dropouts of the alleles of U1 are 4 in CSP5a and 5 in CSP5b, and this is reflected in the dropout rate estimates of 35% for CSP5a and 47% for CSP5b. These estimates cannot be precise because of the masking of the alleles of U1 by those of Q.

**CSP6:** Here, the opposite scenario is considered of a random 50% dropout of the alleles of Q not shared with U1. As expected, this reduces the WoE relative to CSP4, by one ban. At locus FGA both alleles of Q have dropped out in both replicates; the WoE is $-0.74$ at this locus and is $> 0$ at all other loci (not shown). There are again four dropouts in CSP6a but five in CSP6b, which is reflected in different dropout rates over the two replicates. The dropouts were predominantly at the loci with large fragment lengths (D2, D18 and FGA) and so $\gamma_Q$ is moderately high, at 0.6%.

**CSP7:** A new difficulty is introduced: in addition to 50% dropout for the alleles of Q, 50% of the alleles of U1 generate stutter peaks that have a non-negligible probability of masking an allele of Q. Each of these peaks is classified as "uncertain" for the `likeLTD` analysis, irrespective of whether or not Q has an allele at that position. This additional ambiguity in the CSPs reduces further the WoE, to 4.1 bans. Once again the dropout and $\gamma_i$ parameter estimates are broadly reasonable, noting that high precision is not possible here because of the masking effects of the stutters and alleles of U1.

**CSP8:** Random 50% dropout affects the alleles of both Q and U1. Once again all the dropout and $\gamma_i$ estimates are reasonable for the two-contributor analysis, and the WoE is 4.5 bans. Even though the CSPs were created assuming two contributors, there are now only three replicate/locus combinations (out of 20) at which $> 2$ alleles were observed. Thus, it would be possible, though not recommended, to analyse this case assuming one contributor plus dropin. The results (Table 5(ii) two rows) again show that this simplified analysis gives a conservative result, with the WoE now 2.5 bans.

We repeated the computation of the likelihoods for CSP7 and CSP8 (2 contributors) varying the population size used by `R DEoptim` (Table 6). In each case we find no problem with convergence of `DEoptim` and essentially no variation in the result for different population sizes, and so for these cases a population size in `R DEoptim` equal to the number of parameters (NP) appears to be adequate.

## 3.4 Three unprofiled contributors

We generated an example with three unprofiled contributors using the profiles from the Hammer Case (see above) but omitting K2 and treating K1 as if unprofiled. The hypotheses compared were then

$$H_p^3 : \text{Q} + \text{U1} + \text{U2} \qquad \text{with} \qquad H_d^3 : \text{X} + \text{U1} + \text{U2}.$$

With this more challenging optimisation problem, the population size in `R DEoptim` equal to NP is inadequate (Table 6), but population sizes of 4NP and 6NP give congruent results under $H_d$, indicating that 4NP gives a good compromise between computing time and quality of the optimisation achieved by `R DEoptim`.

## 3.5 Complex Case

We simulated a complex CSP with four replicate profiling runs. The genotypes of the four contributors (Q, K1, U1 and U2) were generated randomly assuming HWE and allele fractions from the Caucasian database (EA1). The probabilities of dropout were 0.2, 0.4, 0.6 and 0.7 for Q, K1, U1 and U2, respectively, each constant over runs. Dropout of the two alleles of a homozygote, and/or multiple contributors with the same allele, were treated as independent events (so the allele dropped out only if each copy from each contributor dropped out according to the probabilities above). This makes dropout at such alleles more likely than predicted under the multi-dose dropout model, generating an inflated dropout rate estimates in `likeLTD`

At any locus and in any replicate, with probability 0.95 there were no dropins, otherwise exactly one dropin, chosen according to the allele fractions.

At locus D2 there are three alleles that cannot be explained by Q or K1 (17,22,23), all three of which are replicated and so cannot be explained by drop-in, thus indicating the presence of at least two unprofiled contributors. We therefore compared the following hypotheses:

$$H_p^4 : \text{Q} + \text{K1} + \text{U1} + \text{U2} \qquad \text{with} \qquad H_d^4 : \text{X} + \text{K1} + \text{U1} + \text{U2}.$$

(a)

| Factor | WoE | Lp | Ld | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4.1 | −11.58 | −15.68 | 0.6 | 4.1 | 1.8 | 3.8 | 1.8 | 11.1 | 8.6 | 2.3 | 3.5 | 0.6 |
| 2 | 4.1 | −11.58 | −15.68 | 0.6 | 4.1 | 1.8 | 3.8 | 1.8 | 11.1 | 8.6 | 2.3 | 3.5 | 0.6 |
| 4 | 4.1 | −11.58 | −15.68 | 0.6 | 4.1 | 1.8 | 3.8 | 1.8 | 11.1 | 8.6 | 2.3 | 3.5 | 0.6 |
| 6 | 4.1 | −11.58 | −15.68 | 0.6 | 4.1 | 1.8 | 3.8 | 1.8 | 11.1 | 8.6 | 2.3 | 3.5 | 0.6 |

(b)

| Factor | WoE | Lp | Ld | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 4.5 | −21.41 | −25.88 | 2 | 11.6 | 0.7 | 3.2 | 0.5 | 10.3 | 46.5 | 2.9 | 0.2 | 4.3 |
| 2 | 4.5 | −21.41 | −25.88 | 2 | 11.6 | 0.7 | 3.2 | 0.5 | 10.3 | 46.5 | 2.9 | 0.2 | 4.3 |
| 4 | 4.5 | −21.41 | −25.88 | 2 | 11.6 | 0.7 | 3.2 | 0.5 | 10.3 | 46.4 | 2.9 | 0.2 | 4.3 |
| 6 | 4.5 | −21.41 | −25.88 | 2 | 11.6 | 0.7 | 3.2 | 0.5 | 10.3 | 46.4 | 2.9 | 0.2 | 4.3 |

(c)

| Factor | WoE | Lp | Ld | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 8.3 | −17.1 | −25.38 | 0.1 | 0.1 | 9.9 | 3.7 | 1.6 | 17.4 | 3.5 | 90.9 | 1.3 | 4.7 |
| 2 | 8.3 | −17.1 | −25.38 | 0.1 | 0.1 | 10.0 | 3.7 | 1.6 | 17.3 | 3.5 | 90.6 | 1.3 | 4.7 |
| 4 | 8.3 | −17.1 | −25.38 | 0.1 | 0.1 | 10.0 | 3.7 | 1.6 | 17.2 | 3.5 | 90.6 | 1.3 | 4.7 |
| 6 | 8.3 | −17.1 | −25.38 | 0.1 | 0.1 | 9.9 | 3.6 | 1.6 | 17.5 | 3.6 | 90.8 | 1.3 | 4.7 |

Table 6: Summary of results from `likeLTD` analyses for two 2-contributor CSPs: (a) CSP7, (b) CSP8, and (c) a 3-contributor CSP (Section 3.4), using different population sizes in the `R DEoptim` function. The first column gives the factor by which the number of parameters (NP) is multiplied to define the population size in `R DEoptim`. WoE is in bans, computed as the difference between Lp and Ld which are the overall log likelihoods under the prosecution and defence hypotheses. The LR is reported for individual loci.

| Hp | K1 | Q | U1 | U2 |
|---|---|---|---|---|
| Dropout a | 0.558 | 0.271 | 0.726 | 0.730 |
| Dropout b | 0.559 | 0.272 | 0.727 | 0.730 |
| Dropout c | 0.653 | 0.357 | 0.798 | 0.801 |
| Dropout d | 0.624 | 0.328 | 0.777 | 0.780 |

| Hd | K1 | X | U1 | U2 |
|---|---|---|---|---|
| Dropout a | 0.542 | 0.669 | 0.635 | 0.605 |
| Dropout b | 0.541 | 0.668 | 0.634 | 0.604 |
| Dropout c | 0.633 | 0.746 | 0.717 | 0.690 |
| Dropout d | 0.602 | 0.721 | 0.690 | 0.661 |

Table 7: Dropout rate estimates for the complex case (Section 3.5)

The dropout estimates returned by `likeLTD` tend to overestimate the known values slightly under $H_p^4$, due to the the independent assignment of drop-out as discussed above. Under $H_d^4$, the differences in DNA contribution from the three unprofiled contributors are, as might be expected, not well distinguished.

The log likelihoods returned were -55.3 under $H_p$ and -62.44 leading to a WoE of 7.1 bans. Here, $H_p^4$ is true and so the strong WoE in favour of this hypothesis is encouraging. However, the WoE has been tempered by the presence of substantial masking from the one profiled and two unprofiled other contributors of DNA.

## 3.6 Modifying the model

In this section we introduce various modifications to the model underlying the Hammer Case analysis of Section 1.

### 3.6.1 Including a dropin term

In the original analysis we did not explicitly model dropin because the CSPs show at most three alleles not attributable to the victims, so they are consistent with two unknown contributors and no dropin. We now investigate the effect of introducing an explicit dropin term, retaining U1 so that the algorithm could account for the unreplicated alleles either as dropin or as alleles of U1. Allowing the possibility of dropin had a noticeable impact on the overall WoE (Table 8). The most important single-locus difference was an increase of 5 decibans at D2. This is the only locus at which more alleles were observed in CSPa than in CSPb: overall there are 11 more alleles observed in CSPb than in CSPa, and

| Hammer | Standard | Drop-in | Unc | Beta | | locAdj $\sigma^2$ | | deg |
|---|---|---|---|---|---|---|---|---|
| | | | | -3.97 | -4.73 | 0 | 0.32 | 0 |
| D3 | 1.0 | 1.1 | 1.0 | 1.1 | 1.1 | 1.1 | 1.1 | 1.3 |
| vWA | 1.1 | 1.1 | 1.1 | 1.1 | 1.1 | 1.1 | 1.1 | 1.1 |
| D16 | 1.5 | 1.4 | 1.3 | 1.4 | 1.4 | 1.4 | 1.4 | 1.5 |
| D2 | 0.9 | 1.4 | 0.9 | 1.4 | 1.3 | 1.5 | 1.3 | 1.5 |
| D8 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 | 0.9 |
| D21 | 1.5 | 1.6 | 1.5 | 1.6 | 1.6 | 1.6 | 1.6 | 1.5 |
| D18 | −0.5 | −0.5 | −0.5 | −0.5 | −0.5 | −0.4 | −0.5 | −0.9 |
| D19 | 2.6 | 2.6 | 1.7 | 2.6 | 2.6 | 2.6 | 2.6 | 2.5 |
| TH01 | 0.9 | 0.8 | 0.9 | 0.8 | 0.8 | 0.9 | 0.8 | 0.8 |
| FGA | 1.1 | 1.2 | 0.7 | 1.2 | 1.2 | 1.1 | 1.2 | 1.3 |
| Overall | 10.9 | 11.6 | 9.5 | 11.6 | 11.6 | 11.7 | 11.6 | 11.4 |

Table 8: Single-locus and overall WoE values for analyses of the Hammer case imposing various modifications to the dropout model.

consequently the dropout rate for U1 in CSPa is high (estimate 95% under $H_p'$, compared with 38% for CSPb). This makes the unreplicated D2 23 allele in CSPa better explained as a dropin. However, the fact that the dropin model does not adequately account for the evidence for a high dropout rate in CSPa reflects a limitation of the model, and a reason to prefer the standard analysis above that does not include a dropin term.

### 3.6.2 Uncertain allele calls

The capability of `likeLTD` to treat an allelic position as "uncertain" was not available in `LoComatioN` and so this designation was not made in the Hammer Case CSPs. Additional features of `likeLTD` not incorporated in `LoComatioN` include dropout rate estimates per replicate and per contributor, and modelling of DNA degradation.

To illustrate its impact on WoE, we recoded seven CSP alleles as "uncertain". Overall, this weakens the evidence but it remains very strong at 9.5 bans (Table 8). However, the consequence of changing an allele call to "uncertain" varies, depending for example on whether it affects an allele of Q or a profiled contributor or both. For each allele the change in the computed WoE matches intuition we briefly describe some examples:

- Locus D16, CSPa, allele 11. This is an allele of Q, not shared with K1, and so changing its status from present to uncertain reduces the WoE, but only slightly because the allele is called in CSPb: the single-locus WoE is decreases slightly from

1.5 bans to 1.3 bans (Table 8, column 4).

- Locus D8, CSPa, allele 11 and CSPb allele 15. These are both alleles of K1 not shared with Q, and both are called as alleles in the other run. The WoE is unchanged at 0.9 bans.

- Locus D19, CSPa and CSPb, allele 15.2. This is a rare allele, included in the profile of Q but not K1 or K2, and so changing its status in both CSPs substantially reduces the WoE, by 9 decibans.

There are also indirect effects on all loci, because the changes in allele calls impact the support for dropout parameter values.

In the following subsections, we investigate the effects on WoE of various modifications to the dropout model (6).

### 3.6.3 Varying the dropout power parameter

We performed new analyses assigning $\beta = -3.97$ and $\beta = -4.73$. These values represent one SD above and below the central estimate of $-4.35$ reported by Tvedebrink et al. (2009). The overall WoE values are identical at 11.6 bans, and 11.6 bans, respectively.

### 3.6.4 Varying the dropout locus adjustments

The locus-adjustment parameter $\alpha_s$ allows for differences in dropout rates over loci, beyond the dependence on fragment length that is captured with the degradation model. In the standard analysis, `likeLTD` imposes a gamma distribution penalty term with both parameters equal to 50, implying a prior variance in $\alpha_s$ of 0.14. We first fixed $\alpha_s$ at one (variance = 0), so that the dropout probability of an allele of a given fragment length is the same for all loci. The overall WoE was 11.7 bans. The converse change was to weaken the gamma penalty by changing its prior SD from 0.14 to 0.32 (the two parameters of the gamma distribution penalty changed from both = 50 to both = 10). This reduced the overall WoE to 11.6 bans.

### 3.6.5 Varying the degradation model

Degradation is manifested in higher dropout rates for large fragment length alleles. In the SGM+ system, this has greatest effect for the alleles at D2, D18 and FGA. In Table 8, we note a reduced number of observed alleles relative to other loci at D18 and FGA, but not D2. To investigate the effect of modelling degradation in `likeLTD`, we repeated the analysis but with the degradation parameter $\gamma_i = 0$ for every contributor $i$, which

implies no change in dropout probability with fragment length. This increased the WoE by nearly 6 decibans at locus D2 relative to the standard analysis, but decreased it by 4 decibans at D18.

In contrast with fixing the $\gamma_i$, we removing the penalty term (Table 8). The estimates of $\gamma_Q$, $\gamma_X$, $\gamma_{K1}$ and $\gamma_{U1}$ were almost identical to those given in Appendix A, rows 3 and 6, confirming that the penalty term has little effect in the presence of good information. For K2 we have essentially no information, and $\gamma_{K2}$ was estimated at 0.000 and 0.000 with the penalty, but at 0.001 and 0.003 without it, illustrating that the penalty can prevent inflated $\gamma_i$ estimates in the presence of little information.

The overall effect on WoE of both the failure to model degradation and the removal of the penalty term was close to zero (Table 8).

### 3.7 Conclusions

As discussed above there is no "gold standard" test of a likelihood calculation for LTDNA profiles, but we have shown here a good performance of `likeLTD` in analysing a wide range of crime scene DNA profiles, involving complex mixtures, uncertain allele designations, dropin and dropout, degradation, stutter, and relatedness of alternative possible contributors. `likeLTD` behaves gives the same answer as well-established formulas in simple settings. The parameter estimates and WoE change in a coherent and interpretable manner under artificial modifications of the CSPs, and are robust to major modifications of the modelling assumptions.

As well as providing strong WoE in favour of true contributors in simulation experiments, we found in examples that `likeLTD` identified no support for the presence of DNA even when superficially there appeared to be some support. This, together with the decline in WoE as dropins and dropouts were introduced, illustrate that, while powerful, `likeLTD` does not overstate evidential strength, due to a number of features in the algorithm including $F_{ST}$ and sampling adjustments. The introduction of "uncertain" allele calls mitigates the "cliff edge" problem of yes/no allele calls required by other approaches.

Full results are available from the authors, including all input files that enable others to replicate any of the analyses reported here or to make additional tests.

## 4 Acknowledgments and version history

The underlying mathematical model and its implementation in the `likeLTD` `R` code were developed by DJB. Input into the model came from John Buckleton, as described in Balding and Buckleton (2009). A number of other academics and forensic scientists have

given feedback and encouragement, among them Norah Rudin and Kirk Lohmueller in California, Torben Tvedebrink and Niels Morling in Denmark, Peter Gill (Norway), Hinda Haned (Netherlands), and Roberto Puch-Solis (UK).

Since Version 4.0, DJB has been helped to develop the `R` code by Adrian Timpson, and more recently Chris Steele has helped improve coding and implement the tests described in this document.

The early work in developing Version 5.0 was done by Adrian Timpson, the bulk of the recoding was done by Mayeul d'Avezac of the Research Software Development team in UCL Information Services Division, and some final enhancements were implemented by Chris Steele.

There has been no external funding for this project, although DJB has benefitted from fees paid to UCL Consultants Ltd for expert witness work. His employer University College London, and in particular the UCL Genetics Institute, have supported the project by continuing to pay him a salary during the many months of work time that he has devoted to it.

- **Version 1**
  - Release 1-0, 19/1/10. The initial code had separate files LR1unk.R and LR2unk.R for 1 and 2 unprofiled contributors. Each included functions LRnumer() and LRdenom()
  - Release 1-1, 23/1/10. Restructured code for LR1unk.R to make it more similar to LR2unk.R
  - Release 1-2, 26/3/10. Fixed small bug reported by Kirk Lohmueller, affecting the assignment of allfracs in 3 places
  - Release 1-3, 24/5/10. Changed way dropin is modelled.
- **Version 2**
  - Release 2-0, 21/6/10. Merged previous LR1unk.R and LR2unk.R into a single file LTDNALR.R with the functions LRnumer() from those files renamed as LRnumer1() and LRnumer2(), respectively, and similarly for LRdenom().
  - Release 2-1. The change introduced in V2.1 has since been undone in V3.0, by introduction of a better way to deal with rare alleles
- **Version 3**
  - Release 3-0, 12/10/11. The previous functions LRnumer1(), LRnumer2(), LRdenom1() and LRdenom2() were all replaced by a single function `likeLTD`.

There is now a distinct dropout rate for each replicate (DO). The dropout rate for other individuals is determined as a function of DO and the amount of DNA from that individual relative to the amount contributed by the reference individual (Q or U). We now strip out alleles with zero database frequency. If an allele of Q or CSP is not found in rownames(acbp) this allele is inserted into acbp with count 1. This has speeded up computations so that it now becomes feasible to allow three unprofiled contributors to the crime scene profile when Qcont=F, otherwise two unprofileds + Q. The model for dropout is now improved: the previous kdrop function has gone, and both dropout and dropin calculations are included in a new function Calclik(). Stutter alleles, or other apparent artefacts, can be entered as uncertain alleles allowing the possibility that they could be allelic.

- Release 3-1, 4/1/12. Previously the dropin parameter DI was the non-dropout rate for a hypothetical extra individual, but this is now modified so that the dropin rate for each replicate is DI times the non-dropout rate (1-DO) for that replicate. As before, if DI=0 then all CSP alleles must come from one of the specified contributors. We now allow any of the profiled possible contributors to be unaffected by dropout, including Q. This option should only be used if the individual's alleles are observed in the CSP in every replicate at every locus; otherwise an error is generated. Alleles of profiled possible contributors not subject to dropout are converted to uncertain and removed from the CSP in the preprocessing step and (except for Q) don't play any further role in `likeLTD`. There has been some rearrangement of the code so that more work is done in a preprocessing function that is called only once, rather than being repeated in every call to the main function. Some changes have been made to the way parameters are named and passed; function calls to previous versions of `likeLTD` will not work without modification.

- **Version 4**

  - Release 4-0, 19/3/12. The main innovation is to allow dropout rates to increase with fragment length. Thus, fragment lengths for each allele in the profiling system being employed must be supplied (in base-pairs, bp, centred so that 0 represents an average length). These are passed to `likeLTD` in column 2 of matrix afbp, which replaces vector allfracs in Version 3.1; column 1 is the previous allfracs, and specifies population allele fractions. The program uses the model of Tvedebrink et al. (2012b) and essentially the "dose" of DNA contributed by an individual at an allele is adjusted by a geometric function of

fragment length (increased for below-average fragment lengths, and decreased for above-average). The rate of the geometric distribution is a parameter deg (for degradation), which is a vector with one entry per contributor subject to dropout.

- Release 4-1, 8/5/12. Improvement to computation of number of simulations used when denNu=3 and also starting values for nupa and depa. Release of test document giving results from performance tests of `likeLTD`.
- Release 4-2, 26/6/12. These are mainly minor changes to improve the output and program clarity documentation. The test results document distributed with this code is also updated to include new test results. The most important change is an improved assignment of the simulation size for the likelihood approximation invoked for three unprofiled contributors (i.e. denNu=3). For one or two unknown contributors there should be no changes to results from Version 4.1. BB is now passed as a parameter rather than being assigned as a constant.
- Release 4-3, 10/8/12. Mostly just a few minor changes to documentation but there is one important bug fix that affected the likelihood calculations when DI $> 0$; thus any V4.2 runs that modelled dropin (Drin = TRUE in the wrapper) should be rerun with V4.3. Further improvements to output and to value for nsim.
- Release 4-4, 2/11/12. Two changes:
  - A new block of code can provide much faster computation when Nunp=2 or 3 and DI=0. The speed-up is greatest when the CSPs determine many alleles in the genotypes of the unprofiled contributors. The new code uses combinatorial functions that require the `R gtools` library; `library(gtools)` is now included in the Wrapper, but the package must first be installed using `install.packages("gtools")`. The result of the computation is unchanged from the original code that uses "for" loops. Both codes are kept, and the initial likelihood calculation is done once using each code in order to set flags indicating which is quickest; the faster code is then used for all subsequent calculations at that locus (there are separate flags for the calculations under $H_p$ and $H_d$). Because of this improvement, the previous code that performed a simulation-based approximation to the likelihood when Nunp=3 has been removed, and so `nsim` has been removed from the list of parameters passed to `likeLTD`.
  - Locus adjustment terms are now included in the dropout model, as in

Tvedebrink et al. (2009). However, rather than estimate the locus effects on dropout from external data, they are estimated from the input data for the profile being analysed. Because this may be relatively little information, a strong prior is imposed on the locus adjustments: gamma with both parameters equal so that the mean is 1. The default value of this parameter (lap) is 50, giving a prior standard deviation for the locus adjustments of 0.14, the same as the SD of the estimates of Tvedebrink et al. (2009).

Also the inverse of the exact match probability is output for comparison with the LR for the observed CSP: this is the the standard match probability that would apply if the CSP showed exactly the reference profile of Q, and it is assumed that there is only one contributor. The LR for any other CSP should not exceed the inverse of the match probability.

– Release 4-5, 2/11/12. The power parameter $\beta$ (eq. 6) has been fixed in previous versions at $-4.35$ (Tvedebrink et al., 2009). In this version it is updated in the simulated annealing, separately under $H_p$ and $H_d$, subject to a Gaussian prior/penalty with mean $-4.35$ and SD 0.38, the values obtained by Tvedebrink et al. (2009). This is a relatively minor and sensible change, and we have checked that it has little impact. However all the test results reported in this document are for V4-4 and not V4-5.

- **Version 5**

  – Release 5-0. This is a complete re-writing of the basic code, which is now established as an official `R` package on CRAN. The simulated annealing algorithm used in previous versions for parameter optimisation is replaced with a differential evolution algorithm for optimisation. The underlying likelihood model remains the same as version 4.5, however, significant speed improvements have been gained through re-factoring of R code (e.g. converting for loops into vector/matrix operations), re-writing computationally intensive steps in C, and implementing parallel computation of the C code. Steps that have been implemented in C code include the computation of genotype combinations for unknown contributors, computing allele doses for each genotype combination, dose adjustments for relatedness, heterozygocity, dropout and power. Uploading the package to CRAN comes with improved documentation, version control and ease of access.

  – Release 5-1. This update improved the calculation of the LR when close-relatedness is taken into account.

- Release 5-2. This update adds the function `get.likely.genotypes` that returns the most probable genotypes for each locus, and the most probable whole-profile genotoype. There is an option to return marginal genotype probabilities for each contributor subject to dropout, or joint probabilities for all contributors subject to dropout.
- Release 5-3. This update improves the generation of both allele and output reports. These are now output as .doc files instead of .pdf files, and will now scale with the numer of loci and the number of replicates correctly. The change to .doc files was motivated by client requests, and .pdf files can still be easily obtained by opening the .doc file in MS Word and saving as a pdf. There are additional improvements to the checks for unusual alleles (which will now recognize typos and alleles not present in the database), and to the suggestion of appropriate hypotheses to test.
- Release 5-4. This update improves the optimisation procedure, replacing the simple convergence threshold with a geometric progression of convergence. This includes a geometric progression of the `DEoptim::DEoptim.control CR` variable, which controls the crossover rate of the optimisation algorithm. The combination of these two means that the parameter space is more thoroughly searched in the initial stages, leading to improved optimisation for cases that are difficult to converge, which are a minority of cases. $L_p$ and $L_d$ are now optimised together (within each chunk), allowing for estimation of the progress of optimisation (and an associated progress bar). These changes are incorporated in the new optimisation function, `enumerate`. Small changes to the outputs are included, namely altered default file names (including the case name in the file name) and including which database file is used in the information section. Additionally

# References


Balding, D. J. 1995. Estimating products in forensic identification using DNA profiles. *J. Amer. Statist. Assoc.*, 90(431):839–844.

Balding, D. J. 2005. *Weight of evidence for forensic DNA profiles*. Wiley.

Balding, D. J. 2013. Evaluation of mixed-source, low-template dna profiles in forensic science. *Proc. Natn. Acad. Sci. USA*.

Balding, D. J. and Buckleton, J. 2009. Interpreting low template DNA profiles. *Forens. Sci. Intern.: Genet.*, 4:1–10.

Buckleton, J., Triggs, C., and Walsh, S. 2004. *DNA Evidence*. CRC Press.

Butler, J. 2010. *Fundamentals of Forensic DNA Typing*. Elsevier Academic Press.

Cowell, R. G., Graversen, T., Lauritzen, S., and Mortera, J. 2013. Analysis of DNA Mixtures with Artefacts. *ArXiv e-prints*.

Curran, J., Gill, P., and Bill, M. 2005. Interpretation of repeat measurement DNA evidence allowing for multiple contributors and population substructure. *Forensic science international*, 148(1):47–53.

Gill, P., Gusmão, L., Haned, H., Mayr, W. R., Morling, N., Parson, W., Prieto, L., Prinz, M., Schneider, H., Schneider, P. M., and Weir, B. S. 2012. DNA commission of the International Society of Forensic Genetics: Recommendations on the evaluation of STR typing results that may include drop-out and/or drop-in using probabilistic methods. *Forens. Sci. Intern.: Genet.*, 6(6):679–88.

Gill, P., Kirkham, A., and Curran, J. 2007. LoComatioN: A software tool for the analysis of low copy number DNA profiles. *Forens. Sci. Intern.*, 166:128–138.

Gill, P., Whitaker, J., Flaxman, C., Brown, N., and Buckleton, J. 2000. An investigation of the rigor of interpretation rules for STRs derived from less than 100 pg of DNA. *Forens. Sci. Intern.*, 112:17–40.

Tvedebrink, T., Eriksen, P., Asplund, M., Mogensen, H., and Morling, N. 2012a. Allelic drop-out probabilities estimated by logistic regression–further considerations and practical implementation. *Forens. Sci. Intern.: Genet.*, 6(2):263–7.

Tvedebrink, T., Eriksen, P., Mogensen, H., and Morling, N. 2009. Estimating the probability of allelic drop-out of STR alleles in forensic genetics. *Forens. Sci. Intern.: Genet.*, 3:222–226.

Tvedebrink, T., Eriksen, P., Mogensen, H., and Morling, N. 2012b. Statistical model for degraded DNA samples and adjusted probabilities for allelic drop-out. *Forens. Sci. Intern.: Genet.*, 6:97–101.

# A Allele report for Hammer case

---

# hammer-Allele-Report

hammer

---

# Data provided by forensic scientist

## Crime scene profiles (CSP)

| run | status | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | certain | 14,16 | 15,16,19 | 11,13,14 | 20,23,24,25 | 11,12,13,15 | 28,31 | | 12,14,15.2,17.2 | 6,8,9,9.3 | 22 |
| - | uncertain | | | | | | | | | | |
| 2 | certain | 14,16 | 15,16,17,19 | 11,13,14 | 20,24,25 | 11,12,13,15 | 28,29,30,31,31.2 | 13,14,16,17 | 12,13,14,15.2,17.2 | 6,8,9,9.3 | 22,23,25 |
| - | uncertain | | | | | | | | | | |

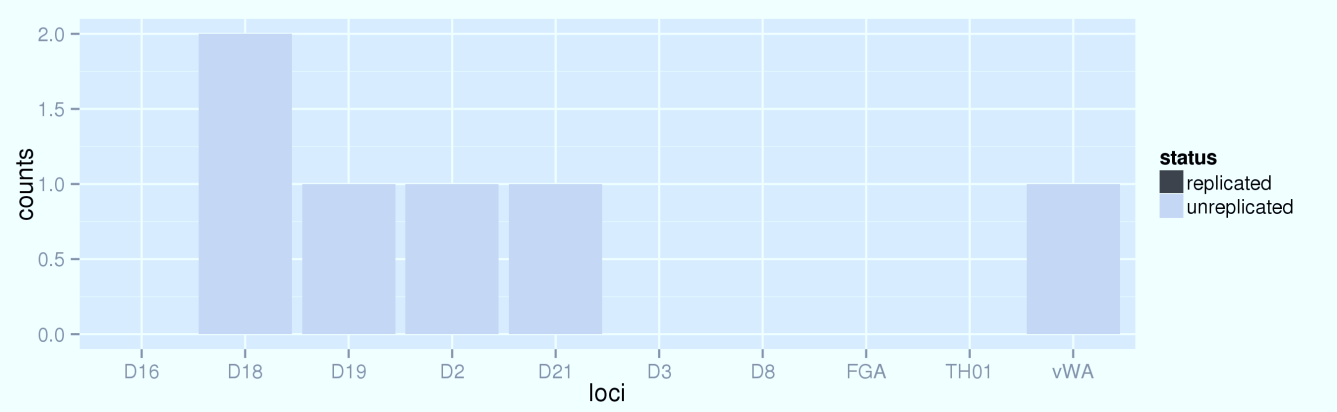
## Reference profiles

| profile | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|
| Suspect | **14,16** | **15,19** | **11,14** | **24,25** | **12,13** | **28,31** | *14,17* | **15.2,17.2** | **9,9.3** | **22**,*23* |
| Victim 1 | **16** | **15,16** | **13** | **20** | **11,15** | *29,30* | *17* | **12,14** | **6,8** | **22**,*25* |
| Victim 2 | 15,17 | **16,19** | **13**,12 | **25**,18 | **11,13** | *29,30* | *17*,15 | **14** | **6**,7 | **22**,20 |
| Other | | *17* | | *23* | | *31.2* | *13,16* | *13* | | |

Alleles that are **replicated**, *unreplicated* or absent in the crime scene profile, using the certain designations only.

# Summary

## Unattributable alleles




The number of 'certain' alleles that cannot be attributed to the known profile(s).

## Unusual alleles

| source | locus | allele | EA1.freq | EA3.freq | EA4.freq | error |
|---|---|---|---|---|---|---|
| Reference profiles | D19 | 17.2 | 0 | 2 | 1 | - |
| Crime scene certain | D19 | 17.2 | 0 | 2 | 1 | - |

Alleles are automatically checked against the database. An error will be reported if an allele is absent from the database, or present more than once, or if a locus is absent.

## Approximate representation

| Contributor | Rep 1 | Rep 2 | Total |
|---|---|---|---|
| Suspect | 85 | 100 | 92 |
| Victim 1 | 75 | 100 | 88 |
| Victim 2 | 47 | 63 | 55 |

The fraction of an individual's alleles (as a percentage) that have been designated as 'certain' alleles in each replicate. This estimate is not used by likeLTD, and is intended to assist informal assessments of possible known contributors to the CSP. A more formal approach is to do a likeLTD run to compute the likelihood ratio (LR) for that individual contributor.

## Suggested parameter values

| nUnknowns | doDropin | Recommendation |
|---|---|---|
| 0 | TRUE | |
| 1 | FALSE | recommended |

Recommended values for 'nUnknowns', choose from 0,1 or 2 (likeLTD automatically adds and additional unknown X to the defence hypothesis in place of the queried profile Q).
Recommended values for 'doDropin', choose from 'TRUE' or 'FALSE'.
All the attributable alleles must either come from an unknown or dropin.

# System information

| Type | Details |
|---|---|
| Date report generated: | Thu Jun 26 17:36:47 2014 |
| Package | likeLTD |
| Title | Tools to determine DNA profile evidence. |
| Description | Tools to determine DNA profile Weight of Evidence. For further information see the likeLTD guide at the URL provided, or the paper under citation. |
| Depends | R (≥ 2.10), DEoptim, ggplot2, gtools, rtf |
| Suggests | svUnit, scales |
| Imports | gdata, tools, tcltk |
| Version | 5.3.3 |
| Date | 2013-03-15 |
| Author | David Balding, Adrian Timpson, Christopher Steele, Mayeul d'Avezac, James Hetherington. |
| Maintainer | Christopher Steele <c.steele.11@ucl.ac.uk> |
| License | GPL-3 |
| URL | https://sites.google.com/site/baldingstatisticalgenetics/ |
| Packaged | 2014-06-26 12:52:36 UTC; steele |
| Built | R 2.15.2; i686-redhat-linux-gnu; 2014-06-26 12:52:38 UTC; unix |

| | |
|---|---|
| sysname | Linux |
| release | 3.8.3-203.fc18.i686 |
| version | #1 SMP Mon Mar 18 13:20:52 UTC 2013 |
| nodename | ZyXEL2 |
| machine | i686 |
| login | steele |
| user | steele |
| effective_user | steele |

# B Output file for Hammer case

---

# hammer-Evaluation-Report

hammer

**Prosecution Hypothesis: Suspect (Q) + Victim 1 + Victim 2 + 1U**

**Defence Hypothesis: Unknown (X) + Victim 1 + Victim 2 + 1U**

---

# Data provided by forensic scientist

## Crime scene profiles (CSP)

| run | status | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | certain | 14,16 | 15,16,19 | 11,13,14 | 20,23,24,25 | 11,12,13,15 | 28,31 | | 12,14,15.2,17.2 | 6,8,9,9.3 | 22 |
| - | uncertain | | | | | | | | | | |
| 2 | certain | 14,16 | 15,16,17,19 | 11,13,14 | 20,24,25 | 11,12,13,15 | 28,29,30,31,31.2 | 13,14,16,17 | 12,13,14,15.2,17.2 | 6,8,9,9.3 | 22,23,25 |
| - | uncertain | | | | | | | | | | |

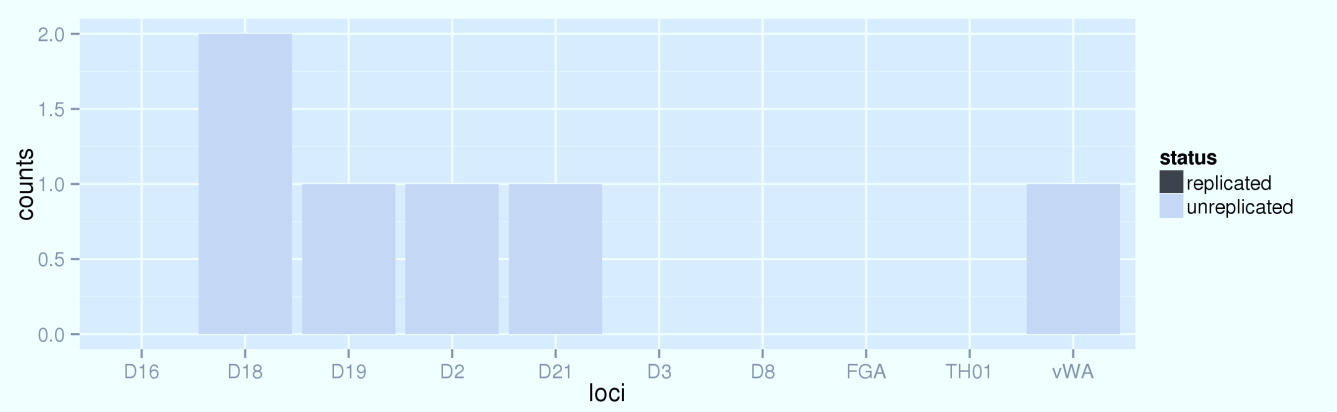
## Reference profiles

| profile | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|
| Suspect | **14,16** | **15,19** | **11,14** | **24,25** | **12,13** | **28,31** | *14,17* | **15.2,17.2** | **9,9.3** | **22**,*23* |
| Victim 1 | **16** | **15,16** | **13** | **20** | **11,15** | *29,30* | *17* | **12,14** | **6,8** | **22**,*25* |
| Victim 2 | 15,17 | **16,19** | **13**,12 | **25**,18 | **11,13** | *29,30* | *17*,15 | **14** | **6**,7 | **22**,20 |
| Other | | *17* | | *23* | | *31.2* | *13,16* | *13* | | |

Alleles that are **replicated**, *unreplicated* or absent in the crime scene profile, using the certain designations only.

# Summary

## Unattributable alleles




The number of 'certain' alleles that cannot be attributed to the known profile(s).

## Unusual alleles

| source | locus | allele | EA1.freq | EA3.freq | EA4.freq | error |
|---|---|---|---|---|---|---|
| Reference profiles | D19 | 17.2 | 0 | 2 | 1 | - |
| Crime scene certain | D19 | 17.2 | 0 | 2 | 1 | - |

Alleles are automatically checked against the database. An error will be reported if an allele is absent from the database, or present more than once, or if a locus is absent.

## Approximate representation

| Contributor | Rep 1 | Rep 2 | Total |
|---|---|---|---|
| Suspect | 85 | 100 | 92 |
| Victim 1 | 75 | 100 | 88 |
| Victim 2 | 47 | 63 | 55 |

The fraction of an individual's alleles (as a percentage) that have been designated as 'certain' alleles in each replicate. This estimate is not used by likeLTD, and is intended to assist informal assessments of possible known contributors to the CSP. A more formal approach is to do a likeLTD run to compute the likelihood ratio (LR) for that individual contributor.

## Likelihoods at each locus

| Likelihood | D3 | vWA | D16 | D2 | D8 | D21 | D18 | D19 | TH01 | FGA |
|---|---|---|---|---|---|---|---|---|---|---|
| Prosecution.log10 | 0.031 | -0.406 | -0.166 | -3.068 | 0.127 | -1.549 | -3.228 | -0.416 | 0.113 | -0.912 |
| Defence.log10 | -1.012 | -1.522 | -1.621 | -3.954 | -0.816 | -3.030 | -2.682 | -3.010 | -0.759 | -1.993 |
| Ratio.log10 | 1.043 | 1.116 | 1.454 | 0.886 | 0.943 | 1.481 | -0.546 | 2.594 | 0.872 | 1.080 |
| Ratio | 11.051 | 13.060 | 28.465 | 7.686 | 8.769 | 30.248 | 0.284 | 392.931 | 7.444 | 12.025 |

## Overall Likelihood

| calculation | estimate |
|---|---|
| Prosecution.log10 | -9.475 |
| Defence.log10 | -20.398 |
| Ratio.log10 | 10.923 |
| Ratio | 83800941044 |

## Theoretical maximum LR

| calculation | estimate |
|---|---|
| likelihood ratio | 218884512327670 |
| Log10 likelihood ratio | 14.340 |

## Dropout and degradation parameter estimates

| hypothesis | contributor | Dropout (Run 1) | Dropout (Run 2) | Degradation (overall) |
|---|---|---|---|---|
| Prosecution | Victim 1 (K1) | 0.463 | 0.026 | 0.009 |

| | | | | |
|---|---|---|---|---|
| Prosecution | Victim 2 (K2) | 1.000 | 1.000 | 0.000 |
| Prosecution | Suspect (Q) | 0.107 | 0.004 | 0.004 |
| Prosecution | U1 | 0.955 | 0.397 | 0.000 |
| Defence | Victim 1 (K1) | 0.436 | 0.045 | 0.008 |
| Defence | Victim 2 (K2) | 1.000 | 1.000 | 0.000 |
| Defence | X | 0.898 | 0.348 | 0.000 |
| Defence | U1 | 0.114 | 0.008 | 0.004 |

### Dropin parameter estimates

| hypothesis | dropin |
|---|---|
| Prosecution | - |
| Defence | - |

# User defined parameters

| Parameter | User input |
|---|---|
| caseName | hammer |
| outputPath | /home/steele/Dropbox/PhD/likeLTD/likeLTD-5.0/TestResults/hammer-standard |
| nUnknowns | 1 |
| ethnic | EA1 |
| adj | 1 |
| fst | 0.02 |
| relatedness1 | 0 |
| relatedness2 | 0 |
| doDropin | FALSE |

# Input files

| File | Used |
|---|---|

| | |
|---|---|
| CSP | hammer-CSP.csv |
| Reference | hammer-reference.csv |
| Database | lgc-allele-freqs-wbp.txt (Default) |

# Optimised parameters

## Prosecution parameters

| parameter | estimate | upper bound | lower bound |
|---|---|---|---|
| locusAdjustment1 | 0.945 | 1.500 | 0.500 |
| locusAdjustment2 | 1.010 | 1.500 | 0.500 |
| locusAdjustment3 | 1.003 | 1.500 | 0.500 |
| locusAdjustment4 | 1.098 | 1.500 | 0.500 |
| locusAdjustment5 | 0.981 | 1.500 | 0.500 |
| locusAdjustment6 | 0.942 | 1.500 | 0.500 |
| locusAdjustment7 | 0.931 | 1.500 | 0.500 |
| locusAdjustment8 | 1.009 | 1.500 | 0.500 |
| locusAdjustment9 | 1.009 | 1.500 | 0.500 |
| locusAdjustment10 | 0.873 | 1.500 | 0.500 |
| power | -4.329 | -2.000 | -6.000 |
| dropout1 | 0.107 | 1.000 | 0.000 |
| dropout2 | 0.004 | 1.000 | 0.000 |
| degradation1 | -2.064 | 0.000 | -20.000 |
| degradation2 | -17.727 | 0.000 | -20.000 |
| degradation3 | -2.355 | 0.000 | -20.000 |
| degradation4 | -17.124 | 0.000 | -20.000 |
| rcont1 | 0.634 | 100.000 | 0.000 |
| rcont2 | 0.000 | 100.000 | 0.000 |
| rcont3 | 0.303 | 100.000 | 0.000 |

## Defence parameters

| parameter | estimate | upper bound | lower bound |
|---|---|---|---|
| locusAdjustment1 | 0.946 | 1.500 | 0.500 |

| | | | |
|---|---|---|---|
| locusAdjustment2 | 0.998 | 1.500 | 0.500 |
| locusAdjustment3 | 1.012 | 1.500 | 0.500 |
| locusAdjustment4 | 1.091 | 1.500 | 0.500 |
| locusAdjustment5 | 0.994 | 1.500 | 0.500 |
| locusAdjustment6 | 0.936 | 1.500 | 0.500 |
| locusAdjustment7 | 0.878 | 1.500 | 0.500 |
| locusAdjustment8 | 1.016 | 1.500 | 0.500 |
| locusAdjustment9 | 1.021 | 1.500 | 0.500 |
| locusAdjustment10 | 0.907 | 1.500 | 0.500 |
| power | -4.403 | -2.000 | -6.000 |
| dropout1 | 0.898 | 1.000 | 0.000 |
| dropout2 | 0.348 | 1.000 | 0.000 |
| degradation1 | -2.108 | 0.000 | -20.000 |
| degradation2 | -17.633 | 0.000 | -20.000 |
| degradation3 | -17.576 | 0.000 | -20.000 |
| degradation4 | -2.363 | 0.000 | -20.000 |
| rcont1 | 1.736 | 100.000 | 0.000 |
| rcont2 | 0.000 | 100.000 | 0.000 |
| rcont3 | 2.609 | 100.000 | 0.000 |

# System information

| Type | Details |
|---|---|
| Date report generated: | Thu Jun 26 17:36:51 2014 |
| Package | likeLTD |
| Title | Tools to determine DNA profile evidence. |
| Description | Tools to determine DNA profile Weight of Evidence. For further information see the likeLTD guide at the URL provided, or the paper under citation. |
| Depends | R (≥ 2.10), DEoptim, ggplot2, gtools, rtf |
| Suggests | svUnit, scales |
| Imports | gdata, tools, tcltk |
| Version | 5.3.3 |
| Date | 2013-03-15 |
| Author | David Balding, Adrian Timpson, Christopher Steele, Mayeul d'Avezac, James Hetherington. |
| Maintainer | Christopher Steele <c.steele.11@ucl.ac.uk> |
| License | GPL-3 |

| | |
|---|---|
| URL | https://sites.google.com/site/baldingstatisticalgenetics/ |
| Packaged | 2014-06-26 12:52:36 UTC; steele |
| Built | R 2.15.2; i686-redhat-linux-gnu; 2014-06-26 12:52:38 UTC; unix |
| sysname | Linux |
| release | 3.8.3-203.fc18.i686 |
| version | #1 SMP Mon Mar 18 13:20:52 UTC 2013 |
| nodename | ZyXEL2 |
| machine | i686 |
| login | steele |
| user | steele |
| effective_user | steele |