Familial searching

Maarten Kruijver
m.v.kruijver@vu.nl
VU University, Amsterdam

23 April, 2015
DNA databases often identify suspects, but what if there is no match?

- If there is no match, one can look for a relative of the donor of a crime scene profile through Familial Searching.
- ‘Finding Criminals Through DNA of Their Relatives’ – Bieber, Brenner, Lazer (Science, 2006)
- Bieber et al. showed that familial searching is feasible using a simulation study.
Familial searching is the process of looking for close relatives of an offender in a DNA database

- Basic principles
- The search process
- Strategies for generating the candidate list
- Simulation studies of search strategies
- Wrap-up
- Computational aspects
- Exercises
Familial searching works by conducting kinship tests on a large scale.

- For each database profile, compute a Kinship Index (KI) with the case profile.
- When searching for full siblings, use the Sibling Index (SI). Parent index (PI) for parent/offspring searches.
- For crime scene profile $x$ and database profile $y$,

$$ SI = \frac{P(x,y \mid \text{full siblings})}{P(x,y \mid \text{unrelated})}, $$

$$ PI = \frac{P(x,y \mid \text{parent/offspring})}{P(x,y \mid \text{unrelated})}. $$
The familiar interpretation and caveats of likelihood ratios apply to kinship indices

### Interpretation

- Kinship Index is a likelihood ratio
- Larger LR means stronger evidence
- Bayes rule: posterior odds = prior odds x likelihood ratio
- LR is sufficient: it completely conveys the strength of the evidence with regard to the two hypotheses. It does not matter, for example, whether 10 or 15 loci were typed if LRs are equal

### Caveats

- Large LR does not imply large posterior odds, since prior odds may be very small
- Large posterior odds do not imply large posterior probability, since the two competing hypotheses need not be exhaustive. For instance, a parent/offspring pair often has a large SI
- ‘Law of truly large numbers’: any outrageous thing is likely to happen at some point. Large LRs may very well be false positives
Bieber et al. find that a true relative often has the largest kinship index of all database members

- A number of test searches are simulated
- For a number of case profiles, a search is simulated
- 62% of the parent/offspring searches had the true relative ranked first
- Searching is feasible. How do we proceed?

**Finding the genetic needle in a large haystack.** The probability of identifying a close relative (i.e., parent/child) of a known offender by kinship searching is shown. Crime scene evidence would be searched against each profile in a simulated offender DNA database. A parent/child would be identified 62% of the time as the very first lead, and 99% of the time among the first 100 leads. Although these familial searching methods do not invariably distinguish parent/child from siblings, they have a high chance of identifying close relatives, if they exist, among the database samples with the highest LRs.
The familial searching process

- **Case profile**
  - Case profile does not match any database profile

- **Database LRs**
  - Compute LR for kinship (full siblings or parent/offspring) with all database members

- **Candidate list**
  - Top $k$: select the top $k$ LRs, for some $k$
  - Fixed LR: select DB members with LR > $t$, for some threshold $t$
  - Profile-dependent: select DB members with LR > $t_\alpha$, s.t. $P(LR > t_\alpha) = \alpha$ for true sibs
  - Conditional: select DB members such that posterior probability $\geq \alpha$

- **Eliminate false positives**
  - Lineage markers (Y-STR typing) can eliminate many false positives
  - Typing additional loci can also eliminate false positives

- **Final candidate list**
  - The final candidate list is investigated tactically by the authorities
There is a trade-off between length of the candidate list and power of detection

Generating the candidate list:

- Select a number of database profiles with the highest LRs
- Trade-off between workload (eliminating false positives) and probability of detection (PoD)

Workload and PoD per case are driven by:

- Case profile (rare alleles or common alleles?)
- Search strategy and tuning parameters (k, t, α)
- Database size (N)
Likelihood ratio distributions depend on the case profile

- For 1,000 simulated SGMplus profiles, the SI-distribution is obtained with respect to a true full sibling and an unrelated person.

Distribution differs a lot between case profiles. Large effect on TPR and FPR.

Variation is caused by rarity of the profile. Profiles with rare alleles are especially amenable to familial searching.

Effect on search strategies is discussed next.
We discuss four search strategies

• Top $k$:
  o select the top $k$ LRs for some fixed $k$;
  o workload is fixed, PoD depends on case profile.

• Fixed LR threshold:
  o select DB members with LR > $t$, for some fixed threshold $t$;
  o both workload and PoD depend on case profile;
  o optimal in the long run.

• Profile-dependent:
  o select DB members with LR > $t_α$, s.t. $P(LR > t_α) = α$ for true sibs;
  o PoD is fixed, workload depends on case profile.

• Conditional:
  o select DB members such that posterior probability that relative is on candidate list, conditional on a relative being present and the observed LRs is at least $α$;
  o both workload and PoD depend on case profile, but average PoD ≥ $α$. 
Conditional

- Suppose the database contains at most one relative
- Prior probability that database contains a relative: $\pi_D$
- Prior probability that database member $i$ is a relative: $\pi_i$
- Likelihood ratio for database member $i$: $r_i$

- Posterior probability that database member $i$ is the relative:

$$\frac{r_i \pi_i}{\sum_{k=1}^{N} r_k \pi_k + 1 - \pi_D}$$

- Strategy: select database members such that the sum of the posterior probabilities is at least $\alpha$
Simulation: top \( k \) strategy (1,000 profiles)

- Large variation in PoD for fixed \( k \)
- Increasing \( k \) gives quickly diminishing returns in terms of PoD
- Using 15 instead of 10 loci makes it possible to increase DB size \( \sim 10 \) times, while retaining the PoD
Simulation: fixed LR strategy (1,000 profiles)

- Large variation in PoD for fixed $t$
- Variance of # candidates is larger for smaller thresholds
- Number of false positives increases linearly with database size
Simulation: conditional (1,000 profiles)

- Uniform prior on database members
- For each case profile, simulate 1,000 times with a full sibling added to the database
- Number of candidates has high variance
- When all LRs are small, many false leads have to be excluded
The fixed-LR strategy is optimal in the long run

- Fixed-LR strategy is most efficient in the long-run: lowest FPR for given TPR.
- How many more false positives with top k or profile-dependent threshold?
- Take top 168 strategy as point of reference in fixed DB (N=100,000). Tuning parameters (t, α) such that the average PoD coincides with top 168.
- Fixing workload is cheap; fixing PoD is not.
Wrap-up

• Workload and PoD per case may depend on case profile
• A fixed-LR threshold is optimal in the long run
• Fixing workload is cheap; fixing PoD not
Computational aspects

• Suppose we have a case profile and work with a fixed SI threshold of 500. How to compute TPR and FPR for this threshold?

Efficient computations with the likelihood ratio distribution

Maarten Kruijver *

Department of Mathematics, VU University, De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands

ABSTRACT

What is the probability that the likelihood ratio exceeds a threshold \( t \), if a specified hypothesis is true? This question is asked, for instance, when performing power calculations for kinship testing, when computing true and false positive rates for familial searching and when computing the power of discrimination of a complex mixture. Answering this question is not straightforward, since there is a huge number of possible genotypic combinations to consider. Different solutions are found in the literature. Several authors estimate the threshold exceedance probability using simulation. Corradi and Ricardi [1] propose a discrete approximation to the likelihood ratio distribution which yields a lower and upper bound on the probability. Nothnagel et al. [2] use the normal distribution as an approximation to the likelihood ratio distribution. Petron et al. [3] introduce an algorithm that can be used for exact

\[
\text{TPR}(t) = P(KI > t \mid H_p)
\]

\[
\text{FPR}(t) = P(KI > t \mid H_d)
\]

R Package: DNAprofiles
The KI follows a probability distribution

- First obtain distribution of KI per locus

Table 1
Probability distribution of the likelihood ratio for pairwise kinship at a locus with a fixed individual. See [16] for a version including subpopulation correction.

| $G_1$ | $G_2$ | $P(G_2|G_1, H_p)$ | $P(G_2|G_1, H_a)$ | LR |
|-------|-------|------------------|------------------|----|
| $a/a$ | $z/z$ | $k_0 p_z^2$      | $p_z^2$          | $k_0$ |
| $a/a$ | $a/z$ | $k_0 2p_a p_z + k_1 p_z$ | $2p_a p_z$ | $k_0 + k_1 p_a^{-1}$ |
| $a/a$ | $a/a$ | $k_0 p_a^2 + k_1 p_a + k_2$ | $p_a^2$ | $k_0 + k_1 p_a^{-1} + k_2 p_a^{-2}$ |
| $a/b$ | $z/z$ | $k_0 p_z^2$      | $p_z^2$          | $k_0$ |
| $a/b$ | $a/z$ | $k_0 2p_a p_z + k_1 1 \frac{1}{2} p_z$ | $2p_a p_z$ | $k_0 + k_1 1 \frac{1}{2} p_a^{-1}$ |
| $a/b$ | $a/a$ | $k_0 p_a^2 + k_1 \frac{1}{2} p_a$ | $p_a^2$ | $k_0 + k_1 \frac{1}{2} p_a^{-1}$ |
| $a/b$ | $b/z$ | $k_0 2p_b p_z + k_1 \frac{1}{2} p_z$ | $2p_b p_z$ | $k_0 + k_1 \frac{1}{2} p_b^{-1}$ |
| $a/b$ | $b/b$ | $k_0 p_b^2 + k_1 \frac{1}{2} p_b$ | $p_b^2$ | $k_0 + k_1 \frac{1}{2} p_b^{-1}$ |
| $a/b$ | $a/b$ | $k_0 2p_a p_b + k_1 \frac{1}{2} (p_a + p_b) + k_2$ | $2p_a p_b$ | $k_0 + k_1 \frac{1}{2} (p_a^{-1} + p_b^{-1}) + k_2 \frac{1}{2} p_a^{-1} p_b^{-1}$ |

- Then estimate or compute the exceedance probability
We use algorithms to compute or estimate probabilities (TPRs, FPRs)

- Estimation:
  - Sample a large number of profiles according to hypothesis;
  - Compute LR for each profile;
  - Estimate exceedance probability by empirical fraction of LRs above threshold.

- Estimation is problematic when probability is small, since no sample will be above threshold.

- Importance sampling is an alternative to regular sampling. Idea: sample from a different hypothesis (Hp instead of Hd) and correct for the bias.

- Exact computation is possible when number of markers is not too large.
Example

```r
library(DNAprofiles)

# load allele freqs
data(freqsNLsgmplus)

# sample one profile
x <- sample.profiles(N = 1, freqs = freqsNLsgmplus)

# obtain distribution function of si under Hp
hp <- ki.cdf(x = x, hyp.1 = "FS", hyp.2 = "UN", hyp.true = "FS", freqs.ki = freqsNLsgmplus)

# obtain distribution function of si under Hd
hd <- ki.cdf(x = x, hyp.1 = "FS", hyp.2 = "UN", hyp.true = "UN", freqs.ki = freqsNLsgmplus)

# plot the functions for a number of thresholds

# plot Hp
plot(t, hp(t), log="x", type="l")

# add Hd
lines(t, hd(t), lty=2)
```
Example
Questions?