Forensic Science International: Genetics 14 (2015) 125–131

Contents lists available at ScienceDirect

Forensic Science International: Genetics

journal homepage: www.elsevier.com/locate/fsig

A series of recommended tests when validating probabilistic DNA profile interpretation software

Jo-Anne Bright [a,b,*], Ian W. Evett [c], Duncan Taylor [d,e], James M. Curran [b], John Buckleton [a]

[a] ESR, Private Bag 92021, Auckland 1142, New Zealand
[b] University of Auckland, Department of Statistics, Private Bag 92019, Auckland 1142, New Zealand
[c] Principal Forensic Services Ltd, United Kingdom
[d] Forensic Science South Australia, 21 Divett Place, SA 5000, Australia
[e] School of Biological Sciences, Flinders University, GPO Box 2100, Adelaide, SA 5001, Australia

ARTICLE INFO

Article history:
Received 25 May 2014
Received in revised form 10 September 2014
Accepted 23 September 2014

Keywords:
Forensic DNA
DNA interpretation
Continuous models
STRmix™
LRmix
Lab Retriever

ABSTRACT

There has been a recent push from many jurisdictions for the standardisation of forensic DNA interpretation methods. Current research is moving from threshold-based interpretation strategies towards continuous interpretation strategies. However laboratory uptake of software employing probabilistic models is slow. Some of this reluctance could be due to the perceived intimidating calculations to replicate the software answers and the lack of formal internal validation requirements for interpretation software. In this paper we describe a set of experiments which may be used to internally validate in part probabilistic interpretation software. These experiments included both single source and mixed profiles calculated with and without dropout and drop-in and studies to determine the reproducibility of the software with replicate analyses. We do this by way of example using three software packages: STRmix™, LRmix, and Lab Retriever. We outline and demonstrate the profile examples where the expected answer may be calculated and provide all calculations.

1. Introduction

The amplification of short tandem repeats (STRs) by the polymerase chain reaction (PCR) is the dominant method of forensic DNA analysis. DNA evidence interpretation is widely accepted in criminal investigations, however there are still recent examples where existing practices have been questioned [1,2]. Prevalent DNA interpretation strategies have been shown to result in likelihood ratios that vary between different laboratories by orders of magnitude [3,4].

There is pressure from many jurisdictions for standardisation including within Australasia, the US [5], and Europe [6]. The International Society for Forensic Genetics (ISFG) have published recommendations for the interpretation of mixed DNA profiles [7], Y STR profiles [8] and profiles that may have allelic dropout or drop-in [9]. In 2012, the editors of Forensic Science International: Genetics (affiliated with ISFG) encouraged more research and the creation of statistical software packages to advance the development and implementation of generally accepted standards for forensic genetics [10].

Current research is steering away from threshold-based interpretation strategies and towards continuous interpretation strategies [11–14]. The introduction of semi and fully continuous (probabilistic) methods has the potential to increase the efficiency of forensic laboratories, and improve the consistency and transparency of the reported results. These probabilistic programmes calculate a likelihood ratio (*LR*). The *LR* is widely considered to be the most powerful and relevant measure of the weight of evidence [15]. It is the ratio of the probability of the evidence (E) given each of two competing hypotheses, H_p, which typically aligns with the prosecution hypotheses, and H_d, which aligns with the defence hypothesis:

$$LR = \frac{Pr(E|H_p)}{Pr(E|H_d)}$$

Semi-continuous methods (also known as discrete or drop models) can optionally incorporate a probability for dropout ($Pr(D|R)$) and/or a probability for drop-in ($Pr(C|S)$), where R is the information used to assign the probability of dropout and S the information used to assign the probability of drop-in. We will

* Corresponding author at: ESR, Private Bag 92021, Auckland 1142, New Zealand. Tel.: +64 8153940; fax: +64 98496046.
E-mail address: Jo.bright@esr.cri.nz (J.-A. Bright).

http://dx.doi.org/10.1016/j.fsigen.2014.09.019
1872-4973/

sometimes omit the conditioning information in future use of these terms. These semi-continuous methods do not use peak heights when generating possible genotype sets and do not model artefacts such as stutter. In these models, peaks must be assigned as stutter or labelled as allelic by an analyst prior to interpretation. One programme [13] allows an ambiguous peak designation of either stutter or allele which seems to be a real advantage. Lab Retriever implements this approach by characterizing the ambiguous peak as a masking allele, and we have found that the same approach may be applied to LRmix (see Appendix 3 for a more complete description of this implementation). The probability of the evidence given all possible genotype sets is then calculated [15,16].

Fully continuous methods assign a probability density for the observed profile or profiles (if interpreting multiple amplifications or replicates) given all the possible genotype combinations. A continuous approach uses nearly all the information within a profile, including peak height, and models the uncertainty in the behaviour of peaks including back stutter. Continuous systems therefore do not require the analyst to specify whether they believe a peak in a back stutter position is allelic or stutter (or a composite of the two) as these possibilities will be considered in the analysis. Both semi and fully continuous based methods are software-based solutions because of their complexity. The replication of the complex calculations by hand is potentially daunting for laboratories and this factor is plausibly preventing uptake of the models.

The requirements for the internal validation of interpretation software have not been previously outlined. We make a distinction between the requirements for developmental versus internal validation. This distinction is also made in the Quality Assurance Standards for Forensic DNA Testing Laboratories (QAS) [17]. Developmental validation is described as (in part) the determination of conditions and limitations of a new or novel DNA methodology for use on forensic samples. It does not explicitly reference interpretation software. Internal validation is data generated within a laboratory to demonstrate that established methods perform as expected. The QAS standards state that interpretation guidelines, including guidelines for mixture interpretation, shall be established using the findings of the internal validation. The Scientific Working Group on DNA Analysis Methods Validation Guidelines (2012) contain recommendations for the validation of DNA analysis methods including software [18], however no guidance is provided on what tests should be undertaken. Internal validation studies traditionally involve tests of accuracy and reproducibility.

There is a subset of profiles where the expected *LR* may be replicated relatively easily by hand. We use the word "expected" here with care. The expected answer from a software programme is the one that should be produced using the model in use. It would be wrong to substitute the words "true" or "correct" as the answer is one produced by a model and the reliability of the model is a matter of subjective judgement.

The performance and limitations of the software may be examined by comparing the software output with the expected answer. An understanding of the models behind the methods is essential for this process as is an understanding of the limitations of the different methods. Examples of where we can predict the expected answer include single source profiles, mixtures where the profile of a major contributor is unambiguous (major/minor) and mixtures of two contributors in equal proportions (balanced). In a simplification of the calculations, profiles may be analysed where no dropout or drop-in is assumed. In subsequent trials, the effect of adding dropout and drop-in on the *LR* can be determined. A performance requirement for any software product should be that its output is intuitively sensible to the analyst given the inputs. A large *LR* does not, in itself, satisfy this requirement.

In this paper we recommend a set of experiments that may be used in part to internally validate software. We suggest they are also essential during developmental validation. We do this by way of example using three software packages: STRmix™ [11], LRmix [19], and Lab Retriever [20]. We outline and demonstrate the profile examples where the expected answer may be calculated. We propose that these validation profiles should be artificially generated by means of a mathematical model rather than from profiling of made up samples. This approach has previously been advocated by Gill et al. [21]. In this way we can control the input variables exactly and produce profiles where the expected answer is known. Consider that it is effectively impossible to create an exact 1:1 mixture in vitro. These experiments would not investigate the full range of functionality of most probabilistic systems but test whether the allele probabilities, population genetic model, and genotype combinations are performing as expected. The mathematics used to construct the profiles in this paper are not the same model as used in any of the programmes except in as much that stutter models used to create the profiles and some decision rules for handling stutter and assigning probabilities for the application of all three programmes have evolved from the same empirical basis. We assert that there is no preselection bias in the creation of the profiles that assists or restricts one or more of these programmes. They are simply tidy single source, major:minor and balanced profiles. Recall the goal of this paper is to specify tests that could and, we argue, should be applied to any software. This is not a head-to-head test of software. In order to "know" the expected answer the profiles need to be simplified. Real samples seldom display this simplicity.

Lab Retriever and LRmix are semi-continuous models. They both require the user to enter a value for the probability of dropout and drop-in. They do not use peak heights and do not model stutter peak heights. LRmix and Lab Retriever input files do not require pre-processing of stutters peaks, but if not filtered, they usually lead to an underestimation of the *LR*, since the sample allele count increases. The peaks are alternatively treated as true and/or drop-in alleles.

STRmix™ uses a fully continuous model for DNA profile interpretation [11] modelling the mass of an allele by fitting an exponential degradation curve to each contributor within a profile [22]. A 'per allele' stutter ratio is applied [23,24]. STRmix™ assigns a weight to each possible genotype combination at a locus. The weights across all combinations at that locus sum to one. A single unambiguous genotype combination at any locus would therefore be assigned a weight of one. The weights are subsequently incorporated into the likelihood ratio calculation. STRmix™ and LRmix implement the same population genetic model namely the Balding and Nichols' equations also known as recommendation 4.2 of NRC II [25,26]. Lab Retriever implements approximations present in the Balding and Buckleton paper [13].

The desirability of using one interpretation model that can be applied across all template amounts has been emphasised repeatedly in the recent literature [6,27–29]. It is preferable that this model can be applied without restrictive workarounds such as locus dropping. All three software products examined here meet this condition. Maintaining two or more methods imposes a quality assurance and training load on an organisation that is unnecessary and counterproductive.

2. Method

2.1. Software

STRmix™ V2.0 is commercial software (www.strmix.esr.cri.nz). Lab Retriever and LRmix are both free open source software available on the internet. Version 1.2.4 of the Lab Retriever