



Validation of probabilistic genotyping software for use in forensic DNA casework: Definitions and illustrations



Hinda Haned ^{a,*}, Peter Gill ^{b,c}, Kirk Lohmueller ^d, Keith Inman ^e, Norah Rudin ^f

^a Netherlands Forensic Institute, Department of Human Biological traces, The Hague, The Netherlands

^b Norwegian institute of Public Health, Oslo, Norway

^c Department of Forensic Medicine, University of Oslo, Norway

^d Department of Ecology and Evolutionary Biology, University of California, Los Angeles, 621 Charles E. Young Drive South, Los Angeles, CA 90095-1606, United States

^e Department of Criminal Justice Administration, California State University, East Bay, 4069 Meiklejohn Hall, 25800 Carlos Bee Boulevard, Hayward, CA 94542, United States

^f 650 Castro Street, Suite 120-404, Mountain View, CA 94041, United States

ARTICLE INFO

Article history:

Received 18 June 2015

Received in revised form 26 November 2015

Accepted 27 November 2015

Keywords:

Validation

Likelihood ratio

Probabilistic

Genotyping

Strength of evidence

LRmix

ABSTRACT

A number of new computer programs have recently been developed to facilitate the interpretation and statistical weighting of complex DNA profiles in forensic casework. Acceptance of such software in the user community, and subsequent acceptance by the court, relies heavily upon their validation. To date, few guidelines exist that describe the appropriate and sufficient validation of such software used in forensic DNA casework. In this paper, we discuss general principles of software validation and how they could be applied to the interpretation software now being introduced into the forensic community. Importantly, we clarify the relationship between a statistical model and its implementation via software. We use the LRmix program to provide specific examples of how these principles can be implemented.

© 2015 The Chartered Society of Forensic Sciences. Published by Elsevier Ireland Ltd. All rights reserved.

1. Background and scope

A number of new computer programs have recently been developed to facilitate the interpretation and statistical weighting of complex DNA profiles in forensic casework (for a review, see [1]). Complex profiles may encompass a multitude of confounding factors resulting from DNA profiling of a low quantity and/or low quality biological sample. The resulting profile may contain multiple contributors, may lack information from the true contributors (allelic drop-out), may include extraneous information unrelated to the crime-sample information (allelic drop-in), and may suffer from degradation or inhibition [2].

It is now accepted throughout the world-wide forensic DNA community that a likelihood ratio (LR) approach is required to reliably interpret these types of profiles [3]. Accordingly, recent years have seen a proliferation of probabilistic models, implemented via software, offered to the community as solutions to this problem. Although these probabilistic models rely on different assumptions, and make use of different types of information, they all enable the evaluation of evidence within a LR framework. While these software programs have proven generally useful to facilitate the interpretation of complex DNA profiles, [4–7], no

generally accepted guidelines exist to establish their validity for use in forensic casework. Model validation for use in forensic casework is not straightforward because the true weight of the DNA evidence cannot be determined; indeed, the generated LR always depends on the model's assumptions, no 'gold standard' exists in the form of a true likelihood ratio that can serve as a comparison [8,9].

In this paper, we offer a set of definitions and examples that aim to provide guidance in validating software for casework use. We first introduce some general definitions of model and software validation taken from existing fields. We then propose a set of considerations for validating software for forensic use. We illustrate the application with the LRmix program [10], which has been validated for casework use and introduced into a courtroom setting.

2. Definition of validation

Forensic science is not the first discipline to face the challenges of model and software validation. Consequently, it is possible to learn from the experience of scientists working in different fields. We follow Rykiel [11] in his definition of model validation (originally applied to the field of ecological science). This paper is highly cited and is effectively regarded as a 'standard reference'. We regard model validation as a

* Corresponding author at: Netherlands Forensic Institute, Netherlands.

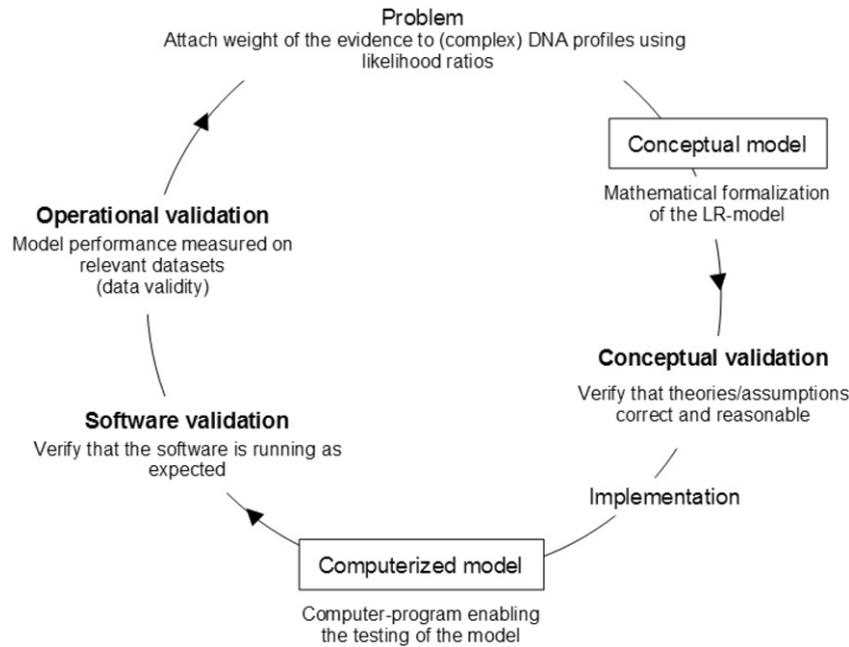


Fig. 1. Simplified representation of the model development and validation process. The diagram shows the different stages of conceptual, operational and software validation (modified from [11]).

process that results in an explicit statement about the behavior of the model (and subsequently the software). In the case of an interpretation model, such a statement would be: “The implementation of Model X in Software Y is valid for application in forensic casework subject to limitations described in the operational validation document”.

Model and software validation are inherently entangled, as software implementation is always needed to implement and use a model (see Fig. 1). However, the two concepts can be related in a simple way; the software is merely a vector for the model. As illustrated in Fig. 1, validated software can actually rely on an invalid model, for example, if the underlying theory or mathematics are shown to be flawed. The goal is to implement a valid model, but it is important to realize that correct implementation of the mathematics of a model by a piece of software provides no information about the validity of the model itself; conversely, demonstration of correct implementation is a critical part of validation.

2.1. Model validation

Model validation ensures that the model has been extensively checked to be sound and fit for purpose. This can be achieved through two steps: conceptual validation and operational validation [11].

2.1.1. Conceptual validation

Conceptual validation verifies that the mathematical formalization of the model, as well as its underlying assumptions, is fundamentally correct. Publication of the theory of the model in peer-reviewed scientific journals allows an opportunity for the underlying theory to be independently assessed, articulates the underlying assumptions, and, most importantly, documents the scientific support for the model structure. For this step to be successful, the model theory must be thoroughly explained. Publication, while necessary, is not sufficient; an editorial decision to publish a paper does not constitute fundamental proof of the scientific validity or usefulness of the contents.

The advent of electronic publication removes space restrictions and allows for the possibility of publishing online supplementary material, and gives modellers the opportunity to expand on their methods. The underlying data on which the conclusions are based can and should be published as supplementary material so that independent

researchers can inspect it and use it to independently verify the results obtained. For open-source software, the computer code can also be published as supplementary material, or as a link provided to the location of the code [12]. The code can then be studied by independent researchers, facilitating an understanding of the model, an important component of conceptual validation. The implementation of the model can also then be independently assessed by interested parties.

The most straightforward way to demonstrate conceptual validity is for the model developer to embrace a transparent approach, which allows for true independent review and verification. A transparent approach requires all of the model assumptions to be described, and accessible to anyone who wishes to independently re-implement the model. This approach is demonstrated by [7,8]. This is diametrically opposed to a black-box approach in which only partial explanations are provided, denying an independent researcher the ability to scrutinize the details and re-implement the model if desired [3,13].

2.1.2. Operational validation

We follow [14] and define operational validation as the procedure that determines whether “the model’s output behavior has the accuracy required for the model’s intended purpose over the domain of the model’s intended applicability”. Operational validation is usually verified using a “computerized model”. In other words, unless a computer implementation of the model is available that can run a profile and yield an output, the operational validity of the model cannot be tested (Fig. 1). Operational validity is tested via user-defined criteria that can be either accepted or rejected. These can be determined for LR-based models. For example, the following properties can readily be tested:

- Comparison to a standard basic model that operates with minimal assumptions so that the effectiveness of models that take into account additional parameters may be measured objectively. Gill and Haned [15] defined the requirements for such model, which allows the evaluation of complex DNA profiles without using all available information.
- The LR of a set of propositions for any profile is lower or equal to the inverse match probability of the profile questioned under the numerator hypothesis [9].
- The LR obtained for a given profile decreases with increasing

Download English Version:

<https://daneshyari.com/en/article/106885>

Download Persian Version:

<https://daneshyari.com/article/106885>

[Daneshyari.com](https://daneshyari.com)