Combining imprecise Bayesian and maximum likelihood estimation for reliability growth models

Lev V. Utkin
Department of Computer Science
Forest Technical Academy
St.Petersburg
lev.utkin@mail.ru

Svetlana I. Zatenko
Department of Mathematics
Forest Technical Academy
St.Petersburg
s_lana2004@mail.ru

Frank P.A. Coolen
Department of Mathematical Sciences
Durham University
Durham
frank.coolen@durham.ac.uk

Abstract

A new framework is explored for combining imprecise Bayesian methods with likelihood inference, and it is presented in the context of reliability growth models. The main idea of the framework is to divide a set of the model parameters of interest into two subsets related to fundamentally different aspects of the overall model, and to combine Walley's idea of imprecise Bayesian models related to one of the subsets of the model parameters with maximum likelihood estimation for the other subset. In accordance with the first subset and statistical data, the imprecise Bayesian model is constructed, which provides lower and upper predictive probability distributions depending on the second subset of parameters. These further parameters are then estimated by a maximum likelihood method, based on a novel proposition for maximum likelihood estimation over sets of distributions following from imprecise Bayesian models for the other subset of parameters. Use of this hybrid method is illustrated for reliability growth models and regression models, and some essential topics that need to be addressed in order to fully justify and further develop this framework are discussed.

Keywords. Bayesian inference, imprecise probabilities, linear regression, lower and upper probability distributions, maximum likelihood estimation, reliability growth models

1 Introduction

One of the main goals of system analysis is to predict its future behaviour on the basis of past experience, for which one typically constructs a statistical model to quantify uncertainties and to enable learning from data. There is a variety of statistical theories and methods for such inference, and researchers often strongly advocate one specific general theory, e.g. the Bayesian approach, whilst rejecting other approaches that also have their merits. In this paper we explore combined use of imprecise Bayesian methods, where sets of prior distributions are used, with maximum likelihood estimation, both on different subsets of all parameters appearing in a statistical model. At first look, these methods may appear to have little in common and one may favour either a complete (imprecise) Bayesian approach or maximum likelihood estimation of all parameters. However, if one considers a Bayesian approach as using a weighted likelihood function, with weights reflecting prior knowledge, the two are less contradictory and exploration of the opportunity to combine both into a hybrid method can be of interest. In this paper we set the first steps in this direction, which include a crucial proposition on maximum likelihood estimation for a subset of parameters following imprecise Bayesian inference on a different subset of parameters. Detailed fundamental analysis and further exploration of this hybrid approach will be important for its full justification, in particular with regard to possible interpretations of the resulting inferences. We present our ideas in the context of reliability growth models.

An important feature of many systems is growth or change of some of their characteristics over time, which has to be taken into account when constructing a statistical model for the system. For example, a common approach for measuring software reliability [18] is by using a statistical model whose parameters are generally estimated from available data on software failures, and the model may be obtained by observing the overall trend of reliability growth during the debugging process. In other words, a software reliability growth model describes how observation of failures, and correcting the underlying faults – such as occurs in software development when the software is being tested and debugged – affect the reliability of software. The word "growth" is rather conventional to describe reliability models with important characteristics changing over time, it does not restrict use of such models to systems whose reliability actually improves. In other words, a growth model can be

regarded to be a mathematical expression which fits experimental data from systems with some important changes over time.

Suppose that $X_1, ..., X_n$ is a series of random variables, for instance, numbers of successful software runs between the $(i-1)$-th and i-th software failures. We suppose that variable X_i is governed by a probability distribution function $p_i(x \mid \mathbf{b}, \mathbf{d})$ depending on two vectors of parameters $\mathbf{b}$ and $\mathbf{d}$. The vector $\mathbf{b}$ contains parameters of the probability distribution under consideration. The vector $\mathbf{d}$ of parameters characterizes the growth, i.e., the growth is modelled by a function $f(i, \mathbf{d})$ which characterizes the change of the system behavior ('growth'). For example in software reliability analysis, the function f mainly shows how parameters $\mathbf{b}$ of the probability distribution p_i change with the number of corrected errors or faults i. Generally, the vector $\mathbf{b}$ depends on $\mathbf{d}$ and the number i of the random variable X_i under consideration.

It should be noted that the growth function in some models is explicitly stated. For instance, Littlewood and Verrall [8] suggest software reliability models with linear and quadratic forms for the function f with two parameters $\mathbf{d} = (d_0, d_1)$: $f(i, \mathbf{d}) = d_0 + d_1 i$ or $f(i, \mathbf{d}) = d_0 + d_1 i^2$. In these models, the growth function is included as parameter of a gamma distribution, which changes with the number of corrected errors in the software.

A similar feature occurs in regression models [9], which in their simplest form provide a relation between predictor variables X_i, $i = 1, ..., n$, and a response variable Y. A typical regression model can be written as

$$Y = f(\mathbf{X}, \mathbf{d}) + \epsilon.$$

Here $\mathbf{X} = (1, X_1, ..., X_n)$; $\mathbf{d}$ is the vector of parameters; ϵ are uncorrelated random errors or noise, usually assumed to have expected value 0 and unknown variance σ^2. In such a model, $\mathbf{d}$ can be a set of growth parameters, for instance, coefficients in a linear regression model, while setting $\mathbf{b} = (\sigma^2)$ fits with the generic notation suggested above.

Clearly, the growth function f may model different characteristics. In software reliability models, it typically enables possible changes of the parameters $\mathbf{b}$ of the probability distribution of random variables X_i to reflect actual changes to software systems, mostly due to error corrections. In regression models, the parameter $\mathbf{b} = (\sigma^2)$ is assumed to be constant, but the growth function characterizes the system behaviour. Nevertheless, both types of models are equivalent from mathematical point of view[1]. In both the cases, we assume a form of f and wish to learn about the parameters $\mathbf{d}$ of f from data.

There are several approaches for inference about growth models on the basis of statistical data. Nowadays, the most popular inferential methods tend to use the likelihood function as main mechanism to link model parameters and statistical data. For models such as reliability growth models, estimation is required both for parameters of the basic probability model and parameters explicitly modelling the growth behaviour. This may involve a substantial number of parameters, with possibly relatively few data available. In this paper, we explore a possible way for dealing with this, by considering imprecise Bayesian inference for one subset of parameters, and a maximum likelihood approach to estimate the other subset of parameters. Such imprecise Bayesian inference has been presented, without a link to maximum likelihood for further parameters, by Walter, Augustin and Peters [17] with application to linear regression models. Typically, a precise parametric model is assumed, with imprecision following through the use of sets of conjugated priors [1, 11, 16]. It is theoretically feasible to use sets of priors for all parameters combined, but this may well lead to very wide posterior intervals for inferences of interest, and if one can estimate some of the parameters by means of maximum likelihood methods, it could be also be attractive with regard to not needing to attempt to assign informative (sets of) prior distributions, in particular if they are on a feature about which no clear expert judgement is available or which one strongly wishes to infer from the data.

The approach we propose in this paper is as follows. By using imprecise Bayesian inference, we can exclude all the parameters of the vector $\mathbf{b}$ from the model, and derive a set of predictive *cumulative distribution functions* (CDFs) such that their lower and upper bounds are conditional on all the parameters of the vector $\mathbf{d}$. This is followed by estimation of the parameters of the vector $\mathbf{d}$, for which we use a modified maximum likelihood estimation method described and justified in Section 3. This approach allows us to reduce the number of parameters in the model and to maximize the likelihood function only over parameters of the vector $\mathbf{d}$ without considering the parameters of vector $\mathbf{b}$. Even further, it can be applied if one explicitly wishes to take expert judgement into account on the part of the model corresponding to parameters $\mathbf{b}$, and this expert judgement is best reflected by imprecise probabilities, while no such prior information is available for the model aspects related to parameters $\mathbf{d}$, for which, however, one can use process data.

[1] The software reliability growth models in the literature are often called regression models due to some common features of the models.

To simplify the presentation of the proposed approach, we study discrete random variables X_i corresponding to the number of successful software runs between the $(i-1)$-th and i-th software failures (for the first software reliability growth model, Section 6) or to the random number of failures between t_{i-1} and t_i (for the second software reliability growth model, Section 7), $i = 1, ..., n$. A general scheme for such combined inference for regression models will be briefly considered in Section 8, to demonstrate that the proposed framework can be applied to various problems.

2 The likelihood principle for constructing standard models

Let $\mathbf{K} = (k_1, ..., k_n)$ be a realization of $X_1, ..., X_n$, with k_i non-negative integers. If probability distributions $p_i(k_i \mid \mathbf{b}, \mathbf{d})$ of the random variables X_i, $i = 1, ..., n$, are known or assumed, then the standard way for obtaining the parameters $\mathbf{b}$ and $\mathbf{d}$ of a growth model is to maximize the likelihood function

$$L(\mathbf{K} \mid \mathbf{b}, \mathbf{d}) = \prod_{i=1}^{n} p_i(k_i \mid \mathbf{b}, \mathbf{d})$$

over a set of parameters $\mathbf{b}$ and $\mathbf{d}$. Values of the parameters $\mathbf{b}$ and $\mathbf{d}$ should be chosen in such a way that makes $L(\mathbf{K} \mid \mathbf{b}, \mathbf{d})$ achieve its maximum.

Many well-known software reliability growth models presented in the literature have been implemented with such standard maximum likelihood estimation. Such models differ only by assumptions about the probability distributions p_i and the growth function f. For example, p_i in the Jelinski-Moranda model [6] is exponential, the Rayleigh distribution is used in the Schick-Wolverton model [13], and the Littlewood-Verrall model [8] uses a Beta distribution.

3 Maximization of the likelihood function over a set of distributions

Suppose that the random variable X_i is governed by an unknown CDF $F_i(k)$ which is only known to belong to the set $\mathcal{M}_i(\mathbf{d})$ defined by the lower and upper CDFs

$$\underline{F}_i(k \mid \mathbf{d}) = \inf_{\mathcal{M}_i(\mathbf{d})} F(k), \tag{1}$$

$$\overline{F}_i(k \mid \mathbf{d}) = \sup_{\mathcal{M}_i(\mathbf{d})} F(k). \tag{2}$$

It should be noted that the set $\mathcal{M}_i(\mathbf{d})$ is the set of *all* CDFs bounded by $\underline{F}_i(k \mid \mathbf{d})$ and $\overline{F}_i(k \mid \mathbf{d})$, so it is *not* the set of parametric distributions having the same parametric form as the bounding distributions. This is an important feature of the proposed approach for combined imprecise Bayesian and likelihood inference in this paper. Moreover, the bounds $\underline{F}_i(k \mid \mathbf{d})$ and $\overline{F}_i(k \mid \mathbf{d})$ are assumed not to depend on the parameters $\mathbf{b}$, which is achieved by taking the predictive CDFs resulting from the imprecise Bayesian approach applied with regard to the parameters $\mathbf{b}$.

The likelihood function can be written in the following form:

$$L(\mathbf{K} \mid \mathbf{d}) = \Pr\{X_1 = k_1, ..., X_n = k_n\}.$$

Proposition 1 explains how the above likelihood function is maximized over all distributions belonging to $\mathcal{M}_1(\mathbf{d}), ..., \mathcal{M}_n(\mathbf{d})$.

Proposition 1 *Suppose that discrete random variables $X_1, ..., X_n$ are governed by a probability distribution $F(k)$ from sets $\mathcal{M}_i$ defined by bounds (1)-(2), respectively. If $X_1, ..., X_n$ are independent, then there holds*

$$\max_{\mathcal{M}_1, ..., \mathcal{M}_n} \Pr\{X_1 = k_1, ..., X_n = k_n\} = \prod_{i=1}^{n} \left\{\overline{F}_i(k_i) - \underline{F}_i(k_i - 1)\right\}. \tag{3}$$

Proof. Denote $N = \{1, 2, ..., n\}$, $\mathbf{M} = (m_1, ..., m_n)$. Let $I_{\{1,...,k_i\}}(m)$ be the indicator function taking the value 1 if $m \le k_i$. The indicator functions are used in the proof to represent all probabilities as expectations of indicator functions, and to write the natural extension in its standard form. The upper bound for the joint probability $\Pr\{X_1 = k_1, ..., X_n = k_n\}$ can be found by solving the following optimization problem:

$$\max \sum_{m_1=1}^{\infty} \cdots \sum_{m_n=1}^{\infty} I_{\{k_1,...,k_n\}}(\mathbf{M}) \prod_{i=1}^{n} p_i(m_i),$$

subject to

$$\sum_{m=1}^{\infty} p_i(m) = 1,$$

$$\underline{F}_i(j) \le \sum_{m=1}^{\infty} I_{\{1,...,j\}}(m) p_i(m) \le \overline{F}_i(j),$$

$$i = 1, ..., n,\ j = 1, 2...$$

The objective function can be rewritten as follows:

$$\prod_{i=1}^{n} \sum_{m_i=1}^{\infty} \left(I_{\{1,..,k_i\}}(m_i) - I_{\{1,..,k_i-1\}}(m_i) p_i(m_i)\right).$$

Introduce new variables

$$F_i(j) = \sum_{m_i=1}^{\infty} I_{\{1,...,j\}}(m_i) p_i(m_i).$$

Then we can rewrite the optimization problem as

$$\max \prod_{i=1}^{n} \{F_i(j) - F_i(j-1)\},$$

subject to

$$\underline{F}_i(j) \leq F_i(j) \leq \overline{F}_i(j),$$
$$\underline{F}_i(j-1) \leq F_i(j-1) \leq \overline{F}_i(j-1),\ i = 1, ..., n.$$

By using the known rules of interval analysis, we obtain (3), which completes the proof. ■

Proposition 1 generalizes the standard likelihood estimation for precise probability models.

4 Imprecise Bayesian models as a way for obtaining the set $\mathcal{M}$

We now consider how to derive the set $\mathcal{M}(\mathbf{d})$. A straightforward way is to use ideas similar to Walley's imprecise Bayesian approach [16].

4.1 Standard Bayesian analysis

One of the efficient approaches to estimation of the model parameters is Bayesian analysis [2, 4, 12]. It treats parameters of concern as random variables which are assigned a prior probability distribution before observations become available. If we assume that the random variable has a probability distribution with vector of unknown parameters $\mathbf{b}$, then these parameters would be regarded as random variables with a prior probability density $\pi(\mathbf{b} \mid \mathbf{c})$, characterized by (hyper-)parameters $\mathbf{c}$. In this case, the Bayesian approach can be applied for computing the CDF for the random variable of interest, with the parameter $\mathbf{b}$ integrated out:

$$F(k \mid \mathbf{c}) = \int_{\Omega} F(k \mid \mathbf{b}) \cdot \pi(\mathbf{b} \mid \mathbf{c}) \mathrm{d}\mathbf{b}.$$

Here Ω is the set of values of $\mathbf{b}$.

Central to the Bayesian approach is the derivation of the posterior distribution of the unknown parameters, given both the data and the assumed prior density for these parameters, and achieved by application of Bayes' theorem. Suppose that the prior distribution $\pi(\mathbf{b} \mid \mathbf{c})$ represents our uncertainty with regard to $\mathbf{b}$ prior to collecting information in the form of a set $\mathbf{K} = (k_1, ..., k_n)$ of observed values of independent random variables $X_1, ..., X_n$. Let $p(k)$ be the probability mass function for the observed data k given $\mathbf{b}$. Then the posterior distribution $\pi(\mathbf{b} \mid \mathbf{K}, \mathbf{c})$ as the conditional distribution of $\mathbf{b}$ given the observed data $\mathbf{K}$ and prior parameters $\mathbf{c}$ is computed as

$$\pi(\mathbf{b} \mid \mathbf{K}, \mathbf{c}) \propto p(k_1) \cdots p(k_n) \cdot \pi(\mathbf{b} \mid \mathbf{c}).$$

Here $\pi(\mathbf{b} \mid \mathbf{K}, \mathbf{c})$ represents updated beliefs about $\mathbf{b}$, with information $\mathbf{K}$ taken into account.

The prior distribution is often chosen to facilitate calculation of the prior, especially through the use of *conjugate priors* [2]. If the posterior distribution $\pi(\mathbf{b} \mid \mathbf{K}, \mathbf{c})$ and the prior distribution $\pi(\mathbf{b} \mid \mathbf{c})$ both belong to the same family of distributions, the π and p are called conjugate distributions and π is called a conjugate prior for p.

4.2 Imprecise prior models

A critical feature of any Bayesian analysis is the choice of a prior distribution, which is often done by considering the choice of (hyper-)parameters of an assumed parametric prior probability distribution. This is both important if one aims at modelling prior information and if one aims to choose a prior distribution in order to reflect the absence of prior information about the parameters. In this paper we focus on the latter case, where a so-called non-informative prior has to be constructed. Many criteria for non-informativeness, and methods to determine non-informative priors, have been proposed in the literature [2, 12], with many methods applying the Bayes-Laplace postulate or the principle of insufficient reason. However, this choice meets some difficulties or problems. In particular, Walley [16] provides examples illustrating possible problems and shortcomings of the principle of insufficient reason. Syversveen [14] presents a detailed review of methods for constructing non-informative priors.

An alternative way for using the Bayesian approach if one wishes not to take prior knowledge into account is through the use of a class $\mathcal{P}$ of (non-informative) prior distributions π [15], which can overcome most problems that can occur when single non-informative priors are used. Such a class of priors can be considered through the lower $\underline{P}$ and upper $\overline{P}$ probabilities of an event A as

$$\underline{P}(A) = \sup\{P_\pi(A) : \pi \in \mathcal{P}\},$$
$$\overline{P}(A) = \inf\{P_\pi(A) : \pi \in \mathcal{P}\}.$$

As pointed out by Syversveen [14] and Walley [16], the class $\mathcal{P}$ under some conditions is "not a class of reasonable priors, but a reasonable class of priors". This means that each single member of the class is not a reasonable model for prior ignorance, because no single distribution can model ignorance satisfactorily, but the whole class is a reasonable model for prior ignorance. When we have little prior information, the upper probability of a non-trivial event should be close to one and the lower probability should be

close to zero. This means that the prior probability of the event may be arbitrary from 0 to 1.

Quaeghebeur and de Cooman [11] proposed a class of imprecise probability models in the framework of the so-called exponential families of probability distributions [2]. These models significantly extend a set of Bayesian imprecise models and give a possibility to develop a framework for imprecise growth models. In our approach, the set $\mathcal{P}$ is used in the imprecise Bayesian framework to take data into account with regard to parameters $\mathbf{b}$, and thus to generate the set $\mathcal{M}$ of predictive distributions with lower and upper bounds which allow us to apply Proposition 1 for maximum likelihood estimation of the parameters $\mathbf{d}$.

5 A general scheme of the model construction

We now present a general scheme for our proposed method that combines imprecise Bayesian inference and maximum likelihood estimation. We present it using the setting of reliability growth models discussed earlier in this paper, but the general idea is more widely applicable. The first task is to define the sets $\mathcal{M}_1(\mathbf{d}), ..., \mathcal{M}_n(\mathbf{d})$ or their bounds by using an appropriate imprecise Bayesian model. It consists of four steps.

1. We divide the set of parameters into two subsets. The first subset contains the parameters $\mathbf{b}$ of the assumed probability distribution p of the random variables $X_1, ..., X_n$. The second subset consists of the growth parameters $\mathbf{d}$.
2. For the assumed probability distribution p of the random variables, we choose an appropriate type of the conjugate prior $\pi(\mathbf{b} \mid \mathbf{c})$ with parameters $\mathbf{c}$.
3. We construct the corresponding Bayesian imprecise model on the basis of results of Walley [16] or Quaeghebeur and de Cooman [11]. At that point we replace the parameters $\mathbf{c}$ by new parameters including the hyperparameter s (see [11, 16] and examples below). The produced set $\mathcal{P}$ depends on the hyperparameter s.
4. By using n observations $k_1, ..., k_n$, we write the lower $\underline{F}_i(k \mid \mathbf{d}, s)$ and upper $\overline{F}_i(k \mid \mathbf{d}, s)$ predictive CDFs as functions of the parameters $\mathbf{d}$ and the hyperparameter s for every debugging period. These functions form the sets[2] $\mathcal{M}_1(\mathbf{d}), ..., \mathcal{M}_n(\mathbf{d})$.

[2]It should be noted that the set $\mathcal{M}_i(\mathbf{d})$ also depends on the hyperparameter s. However, we omit this parameter for shorter notation.

After completing the four steps of the first task, the sets $\mathcal{M}_1(\mathbf{d}), ..., \mathcal{M}_n(\mathbf{d})$ have been derived and these sets do not depend on the parameters $\mathbf{b}$ or $\mathbf{c}$. They depend only on the growth parameters $\mathbf{d}$, the hyperparameter s for the imprecise prior class, and the number of debugging periods i. The second task is to estimate the parameters $\mathbf{d}$, it consists of two steps.

1. The likelihood function $L(\mathbf{K} \mid \mathbf{d}, s)$ is derived by applying Proposition 1.
2. Values of the parameters $\mathbf{d}$ for a fixed s should be chosen in such a way that makes $L(\mathbf{K} \mid \mathbf{d}, s)$ achieve its maximum.

Note that the parameters $\mathbf{b}$ do not appear in the process, as they have been integrated out with the use of a class of priors to derive predictive distributions, and this process also implicitly replaced the parameters $\mathbf{c}$ by s. Clearly, the step to get $\mathbf{b}$ out of the model, without explicitly estimating their values, is imprecise and leads to predictive imprecise probabilities for the random variables of interest. For example, if we construct a software reliability model, then we are looking for the predictive behavior of the analyzed software after n corrections of errors. In other words, we have to compute the probability measures of time to the $(n+1)$-th failure, in particular, the lower and upper probability distributions of time to the $(n+1)$-th failure. These bounds are totally determined by the parameters $\mathbf{d}$ and s in our approach, with s chosen to specify the class of priors, and $\mathbf{d}$ to be estimated by our proposed maximum likelihood approach in the second stage of our method.

In the following sections, we illustrate our method by considering some special cases which apply known imprecise Bayesian models and consider well-known software reliability growth models.

6 A software run reliability growth model

The detailed description of software run reliability models is given in [3]. A run is a minimum execution unit of software. Any software execution process can be divided into a series of runs. When a run is executed, the software either passes or fails. Usually it is assumed that after observing a software failure, the software is corrected and it is usually assumed that this action actually removes the software error that caused the failure, hence the software improves due to this action and therefore the term reliability growth tends to be used. There are many variations to this basic scenario in the software reliability literature, we do not address these here.

Let X be a run lifetime of software, that is, X is a discrete random variable taking the value k if the software fails during the k-th run after $k-1$ successful runs. The run lifetime distribution (probability mass function) is defined as $p(k) = \Pr\{X = k\}$.

6.1 The imprecise beta-geometric model

If we assume that the random variable X is governed by the geometric distribution with parameter r and the probability mass function

$$p(k \mid r) = (1-r)^{k-1} r,\ k = 1, 2, ...,$$

then the set $\mathcal{M}$ can be constructed by using an imprecise model that is very similar to the beta-binomial model proposed by Walley [16]. The prior Beta distribution of the random variable r, denoted $\mathrm{Beta}(\alpha, \beta)$ with parameters $\alpha > 0$ and $\beta > 0$, has probability density function

$$\pi(r) = \frac{1}{\mathrm{B}(\alpha, \beta)} r^{\alpha-1}(1-r)^{\beta-1},\ 0 \le r \le 1.$$

Here $\mathrm{B}(\alpha, \beta)$ is the standard beta function.

Using the general notation introduced before in this paper for our new method, we write $\mathbf{b} = (r)$, $\mathbf{c} = (\alpha, \beta)$. If we observe k runs of software between the $(i-1)$-th and i-th software failures, and we assume that the number of such runs is geometrically distributed with parameter r, then the posterior distribution $\pi(r \mid k, \mathbf{c})$ is again a beta distribution, namely

$$\pi(r \mid k, \mathbf{c}) = \mathrm{Beta}(\alpha + 1, \beta + k).$$

Here Bayesian analysis leads to the probability distribution of the number of events with parameters α and β. We can call this a beta-geometric model. In the beta-binomial model, Walley proposed to replace these parameters by introducing s and γ, with $\alpha = s\gamma$ and $\beta = s - s\gamma$, and then the parameter γ is allowed to take on any value in the interval from 0 to 1, hence a set of prior distributions is created which only depends on the choice of $s > 0$, and which trivially leads to a corresponding set of posterior distributions. The hyperparameter s determines the influence of the prior distribution on posterior probabilities [16]. The beta-geometric model proposed here can be given exactly the same imprecise Bayesian treatment, resulting in what we call the imprecise beta-geometric model. The lower and upper bounds can be obtained by minimizing and maximizing the probabilities of events over all values γ in $[0, 1]$.

6.2 The imprecise beta-geometric growth model

Suppose that the probability $r = r_i$ is a random variable having a beta distribution with prior parameters α and $\beta + f(i, \varphi)$. Here $f(i, \varphi)$ is a function characterizing the software reliability growth, in particular, assume for simplicity that $f(i, \varphi) = (i-1) \cdot \varphi$. In this case, we get a model with three parameters, including two prior parameters α and β of the probability distribution and one parameter φ which characterises the reliability growth. The notation introduced above can be used by defining $\mathbf{c} = (\alpha, \beta)$ and $\mathbf{d} = (\varphi)$.

The construction of the model is based on the idea of dividing the set of parameters α, β, φ into two subsets and to consider the imprecise Bayesian model on the set $\mathcal{M}_i(\varphi)$ of CDFs bounded by some lower $\underline{F}_i(k \mid \varphi, \alpha, \beta)$ and upper $\overline{F}_i(k \mid \varphi, \alpha, \beta)$ CDFs which are defined by the set of parameters $\mathbf{c} = (\alpha, \beta)$ for a fixed parameter φ, for $i = 1, ..., n$. In other words, we fix φ and construct the sets of CDFs $F_i(k)$ with bounds depending on $f(i, \varphi)$ by using the imprecise beta-geometric model.

After constructing the set $\mathcal{M}_i(\varphi)$ of CDFs $F_i(k \mid \varphi)$ having the lower $\underline{F}_i(k \mid \varphi, \alpha, \beta)$ and upper $\overline{F}_i(k \mid \varphi, \alpha, \beta)$ CDFs for every $i = 1, ..., n$, and by assuming that the random variables $X_1, ..., X_n$ are independent, the likelihood function can be written and maximized by application of Proposition 1, leading to the value φ_0 that maximises this likelihood, so which we consider an appropriate estimate of φ.

Denote the parameters of the i-th posterior beta distribution after n observations

$$\alpha^* = \alpha + n - 1,\quad \beta_i^* = \beta + D_i(\varphi),$$

where

$$D_i(\varphi) = K_n + f(i, \varphi),\ K_n = \sum_{j=1}^{n-1} (k_j - 1).$$

We have to draw attention that the prior parameter β for the i-th posterior beta distribution is $\beta_i^* = \beta + f(i, \varphi)$. In addition, we get K_n runs of the software during n periods of observations. This implies that the posterior parameter β_i^* for i-th period of debugging is defined by n periods of observations. This is a very important feature and that is why we use index i for the posterior parameter β^*.

It can be also seen from the above that the posterior parameters depend on $\mathbf{d}$. In the considered special case, β^* depends on $f(i, \varphi)$.

Now we can write the predictive CDF for the i-th step of the software debugging after n observations as

follows:

$$F_i(k \mid \varphi, \alpha, \beta) = \int_0^1 (1-(1-p)^k) \cdot \mathrm{Beta}(\alpha^*, \beta_i^*)\mathrm{d}p$$
$$= 1 - \frac{\mathrm{B}(\alpha^* + \beta_i^*, k)}{\mathrm{B}(\beta_i^*, k)}.$$

By using the introduced notation $\alpha = s\gamma$, $\beta = s - s\gamma$, we write

$$F_i(k \mid \varphi, \gamma, s) = 1 - \frac{\mathrm{B}(s+n-1+D_i(\varphi), k)}{\mathrm{B}(s - s\gamma + D_i(\varphi), k)}.$$

The function $F_i(k \mid \varphi, \gamma, s)$ increases as γ increases in the interval $[0,1]$, because the beta function $\mathrm{B}(x,y)$ is decreasing in x for $x > 0$. This implies that the lower bound for $\mathcal{M}_i(\varphi)$ is determined as

$$\underline{F}_i(k \mid \varphi, s) = \sup_{\gamma \in (0,1)} F_i(k \mid \varphi, \gamma, s)$$
$$= 1 - \frac{\mathrm{B}(s+n-1+D_i(\varphi), k)}{\mathrm{B}(s + D_i(\varphi), k)}.$$

The upper bound is determined as

$$\overline{F}_i(k \mid \varphi, s) = \inf_{\gamma \in (0,1)} F_i(k \mid \varphi, \gamma, s)$$
$$= 1 - \frac{\mathrm{B}(s+n-1+D_i(\varphi), k)}{\mathrm{B}(D_i(\varphi), k)}.$$

By having the lower and upper CDFs, it follows from Proposition 1 that the likelihood function maximized over $\mathcal{M}_i(\varphi)$ by given s and φ is of the form:

$$\max_{\mathcal{M}(\varphi)} L(\mathbf{K} \mid \varphi, s)$$
$$= \prod_{i=1}^{n} \left(\overline{F}_i(k_i \mid \varphi, s) - \underline{F}_i(k_i - 1 \mid \varphi, s)\right)$$
$$= \prod_{i=1}^{n} \left(\frac{\mathrm{B}(C_i,\ k_i - 1)}{\mathrm{B}(s + D_i(\varphi),\ k_i - 1)} - \frac{\mathrm{B}(C_i,\ k_i)}{\mathrm{B}(D_i(\varphi),\ k_i)} \right).$$

Here $C_i = s + n - 1 + D_i(\varphi)$.

The parameter φ should be chosen in such a way that makes $\ln L(\mathbf{K} \mid \varphi, s)$ achieve its maximum. The optimal value φ_0 of φ can be found by numerically solving the equation $\partial \ln L(\mathbf{K} \mid \varphi, s)/\partial\varphi = 0$. Once we have calculated the estimate of the parameter φ, we can derive the lower and upper software run failure functions after the n-th software failure, i.e., we can compute the lower and upper CDFs of the $(n+1)$-th failure

$$\underline{F}_{n+1}(k, s) = 1 - \frac{\mathrm{B}(s+n+D_{n+1}(\varphi_0),\ k)}{\mathrm{B}(s + D_{n+1}(\varphi_0),\ k)},$$

$$\overline{F}_{n+1}(k, s) = 1 - \frac{\mathrm{B}(s+n+D_{n+1}(\varphi_0),\ k)}{\mathrm{B}(D_{n+1}(\varphi_0),\ k)}.$$

7 NHPP software reliability models

One of the important frameworks for developing software reliability models dealing with numbers $N(t)$ of software failures occurring up to a certain time period t is the non-homogeneous Poisson process (NHPP). Let $X_i = N(t_i) - N(t_{i-1})$ be the random number of failures between t_{i-1} and t_i. For any time points $0 < t_1 < t_2 < ...$ (for ease of notation, let $t_0 = 0$), the probability that the number of failures between t_{i-1} and t_i is k, $k = 0, 1, 2, ...$, can be written as

$$\Pr\{N(t_i) - N(t_{i-1}) = k\}$$
$$= \frac{\{m(t_i) - m(t_{i-1})\}^k}{k!} e^{\{-(m(t_i) - m(t_{i-1}))\}}. \quad (4)$$

Here $m(t)$ is the mean number of failures occurring up to time t. The NHPP models differ through the function $m(t)$, popular examples of which for software reliability models are $m(t) = a(1 - \exp(-bt))$ (Goel-Okumoto model [5]) and $m(t) = a\ln(1+bt)$ (Musa-Okumoto model [10]). Our goal is to estimate the parameters a and b for such a model, based on statistical data consisting of numbers of failures k_i per subintervals $(t_{i-1}, t_i]$, $i = 1, ..., n$. As before, we denote these data by the vector $\mathbf{K} = (k_1, ..., k_n)$.

7.1 The imprecise negative binomial model

When the number of failures has a Poisson distribution with the parameter λ, gamma distributions are conjugate priors, denoted by $\mathrm{Gamma}(\alpha, \beta)$. If we observed K failures during a period of time T, then the posterior distribution is $\mathrm{Gamma}(\alpha^*, \beta^*)$, where $\alpha^* = \alpha + K$ and $\beta^* = \beta + T$. Hence, the predictive probability of k failures during time t under condition that K failures were observed during time T is [2]

$$P(k) = \int_0^\infty \frac{(\lambda t)^k e^{-\lambda t}}{k!} \mathrm{Gamma}(\alpha^*, \beta^*)\mathrm{d}\lambda$$
$$= \frac{\Gamma(\alpha^* + k)}{\Gamma(\alpha^*)k!} \left(\frac{\beta^*}{\beta^* + t}\right)^{\alpha^*} \left(\frac{t}{\beta^* + t}\right)^k. \quad (5)$$

Here $\Gamma(\alpha)$ is the standard gamma function.

7.2 The imprecise negative binomial growth model

A wide range of suitable mean value functions can be represented in the form $m(t; a, b) = a \cdot \tau(t, b)$. The parameter λ of the Poisson distribution in (5) and the argument t can be replaced by the parameter a and the discrete time $\tau(t_i, b) - \tau(t_{i-1}, b)$, respectively. In fact, by replacing λ by a, we get the Poisson process with a scaled time of the software testing, i.e., every time interval $[t_{i-1}, t_i]$ is replaced by the interval

$[\tau(t_{i-1},b),\tau(t_i,b)]$. Then we can write the predictive CDF of the number of failures in the time interval between t_i and t ($t\in[t_i,t_{i+1}]$) after n observation periods through the regularized incomplete Beta-function [7] as follows:

$$F_i(k,t|\mathbf{c},b)=1-\frac{\mathrm{B}_{q(i,t)}(k+1,\alpha+K_n)}{\mathrm{B}(k+1,\alpha+K_n)}$$
$$=1-I\left(q(i,t),k+1,\alpha+K_n\right).$$

Here $t_0=0$, $k_0=0$,

$$q(i,t)=\frac{T_i(t,b)}{\beta+\tau(t_n,b)+T_i(t,b)},$$
$$T_i(t,b)=\tau(t,b)-\tau(t_i,b),\ K_n=\sum_{j=1}^{n}k_j,$$

$\mathrm{B}_q(k+1,r)$ is the incomplete Beta-function with $I\left(q,k,r\right)$ the regularized incomplete Beta-function.

We must select a bounded set for the vector (α,β), in order to avoid ending up with vacuous posterior predictive distributions. In analogy with imprecise prior classes described above, we want this set to be described by a single hyper-parameter s, and we choose all vectors (α,β) within the triangle $(0,0)$, $(s,0)$, $(0,s)$. This implies that all possible prior 'rates of occurrence of failures' are represented, as the prior allows interpretation of $\alpha/\beta=\gamma$ as this rate, hence this would include all such rates in $(0,\infty)$. This prior set, and related inferences, is of course similar in nature to the work by Quaeghebeur and de Cooman [11], yet it is slightly different. This prior set leads to the lower and upper bounds for $\mathcal{M}_i(b)$ by $t\in[t_i,t_{i+1}]$

$$\underline{F}_i(k,t\mid s,b)=1$$
$$-I\left(\frac{T_i(t,b)}{\tau(t_n,b)+T_i(t,b)},k+1,s+K_n\right),$$

$$\overline{F}_i(k,t\mid s,b)=1$$
$$-I\left(\frac{T_i(t,b)}{s+\tau(t_n,b)+T_i(t,b)},k+1,K_n\right).$$

The next step is to use Proposition 1 and to maximize the likelihood function over the set of b

$$L(\mathbf{K}|b,s)=\prod_{i=1}^{n}\left(\overline{F}_i(k_i,t_i\mid s,b)-\underline{F}_i(k_i-1,t_i\mid s,b)\right).$$

Once we have the maximum likelihood estimator, following Proposition 1, of the parameter b, we can construct the lower and upper bounds for the CDF of the number of failures in time interval $[t_n,t]$ after n periods of debugging.

8 Regression model (general scheme)

We briefly explain how the combined imprecise Bayes and likelihood approach, proposed in this paper, can be applied to basic regression problems. Suppose that we have $n+1$ variables Y and X_j, $j=1,...,n$, with Y being a dependent variable and $\{X_1,...,X_n\}$ being n independent predictor variables, related to Y according to the relation $Y=f(X_1,...,X_n)$. The standard linear regression model[3] is a special case and can be written as

$$Y=\mathbf{X}\mathbf{d}+\epsilon.$$

Here $\mathbf{X}=(1,X_1,...,X_n)$; $\mathbf{d}=(d_0,...,d_n)^{\mathrm{T}}$ is the vector of parameters; ϵ are random errors or noise having zero mean and the unknown variance σ^2.

To fit with the presentation in this paper, we assume that ϵ is a discrete variable[4]. Let us construct the imprecise Bayesian model for ϵ. If ϵ is governed by some probability distribution $p(z\mid\sigma)$ and there is the corresponding conjugate distribution $\pi(\sigma\mid\mathbf{c})$, then we can find the predictive CDF $F_n(z\mid s,\gamma)$ after having n observations $(y_1,\mathbf{x}_1),...,(y_n,\mathbf{x}_n)$ depending on new parameters s,γ [11] and its bounds $\underline{F}(z\mid s)$, $\overline{F}(z\mid s)$.

Denote $z_i=y_i-\mathbf{x}_i\mathbf{d}$ and $\mathbf{Z}=(z_1,...,z_n)$. Having derived the lower and upper CDFs, it follows from Proposition 1 that the likelihood function that is to be maximized over $\mathcal{M}$, with given s, is of the form:

$$\max_{\mathcal{M}}L(\mathbf{Z}\mid s)=\prod_{i=1}^{n}\left(\overline{F}(z_i\mid s)-\underline{F}(z_i-1\mid s)\right).$$

Denote $z_i=y_i-\mathbf{x}_i\mathbf{d}$. Hence

$$\max_{\mathcal{M}}L(\mathbf{Z}\mid s)$$
$$=\prod_{i=1}^{n}\left(\overline{F}(y_i-\mathbf{x}_i\mathbf{d}\mid s)-\underline{F}(y_i-\mathbf{x}_i\mathbf{d}-1\mid s)\right).$$

Now we can find parameters $\mathbf{d}$ by maximizing the obtained likelihood function.

In the regression model, we again separate the parameters of the probability distribution of ϵ and the parameters $\mathbf{d}$. However, in contrast to the software reliability models, the parameters $\mathbf{c}$ directly do not change with the growth parameters $\mathbf{d}$ (see the parameter β^* and the function $f(i)$ in Subsection 6.2 for comparison). Moreover, the set $\mathcal{M}$ and its bounds do not depend on the parameters $\mathbf{d}$. This allows us to avoid the index i and to consider identical sets $\mathcal{M}$. Nevertheless, the general approach for modelling and inference is the same as described in this paper.

[3] The more general model $Y=f(\mathbf{X},\mathbf{d})+\epsilon$ which can be analyzed in the same way.

[4] See Section 9 for comments relevant to the more usual case with continuous ϵ

9 Concluding remarks

In this paper we have proposed a way towards development of statistical methods that combine imprecise Bayesian inference for one subset of all parameters with maximum likelihood estimation for the other parameters. The key to this approach is Proposition 1, which provides a generalization of maximum likelihood estimation for discrete variables with sets of distributions. There are many important research questions that need answering, in particular with regard to the interpretations of these inferences and their application to large scale problems. We particularly see a benefit in models with differing features related to different parameters, for example the reliability growth models discussed in some detail and used to present and illustrate the novel approach in this paper, where some parameters are specifically used to model the growth aspect. It should also be studied in which situations this approach is most valuable. For example, it may well be most suitable in situations where one has significant prior knowledge on some parameters, yet does not feel confident enough to assign precise prior distributions to them, whereas on another aspect of the model one has no prior knowledge and explicitly wishes only to estimate those parameters using the data. Some statisticians might object if the same data set is used for related inference in two different stages, feeling that the same data might be used twice. This would be wrong, as the parameters estimated at the different stages play different roles, and hence estimates are based on different aspects of the information within the total data set available.

We presented the main idea of the new framework in this paper as an extension of the known imprecise Bayesian models [11, 16] to situations where the process considered has some changeable behaviour, which we also wish to estimate using the data. In line with most reported developments in such imprecise Bayesian models, we presented it from the perspective of a non-informative prior set of distributions, but it may indeed well be more useful to apply this combined method with an informative prior set of distributions. When such sets are also defined using conjugate priors in the same way as for these non-informative prior sets, that is done in a straightforward manner which we will discuss and explore further elsewhere. We chose to focus our presentation on software reliability growth models, as these typically have clear divisions of the parameters according to the different roles, which we consider very suitable for the method proposed. As indicated, the general approach might also provide a promising method for imprecise regression models.

We have stated in Section 3 that the set $\mathcal{M}_i(\mathbf{d})$ is the set of *all* CDFs bounded by $\underline{F}_i$ and $\overline{F}_i$. One could also consider the use of only a set of parametric distributions, all with the same parametric form as the bounding distributions. However, following this approach, maximization of the likelihood function over a set of distributions with parameters $\mathbf{c}$ derived in Section 3 is reduced to its maximization over a set of parameters $\mathbf{c}$. In this case, we get the standard statistical model completely based on the maximum likelihood estimation, which does not differ from many well-known models of software reliability and regression models.

Due to limited size of this paper, we did not illustrate the proposed models by data examples, such examples will be included in specific topic oriented presentations elsewhere, where we also compare these inferences to other inferences including full Bayesian and full likelihood approaches. Nevertheless, we wish to point out that initial indications from computational examples suggest that this new combined method performs well, also so if there are relatively few data, but further study is required in order to draw general conclusions.

We did not consider continuous random variables, but of course this case is very important. However, Proposition 1 can be extended on the continuous case, so it looks like the method can also be applied for continuous random variables $X_1, ..., X_n$. In this case, the likelihood function can be written as

$$L(\mathbf{X}) = \lim_{\triangle_1 \to 0, \ldots, \triangle_n \to 0} \frac{\Pr\{x_1 \le X_1 \le x_1 + \triangle_1, .., x_n \le X_n \le x_n + \triangle_n\}}{\triangle_1 \cdots \triangle_n},$$

and this suggests that maximum likelihood estimates for the parameters can be derived by maximizing

$$\max_{\mathcal{M}_1, \ldots, \mathcal{M}_n} L(\mathbf{X}) = \prod_{i=1}^{n} \left(\overline{F}_i(x_i) - \underline{F}_i(x_i)\right) \delta(x_i). \quad (6)$$

Here $\delta(x_i)$ is Dirac function which has unit area concentrated in the immediate vicinity of points x_i. The likelihood function achieves its maximum by taking the probability density functions such that $\rho_i(x_i) = \left(\overline{F}_i(x_i) - \underline{F}_i(x_i)\right) \delta(x_i)$. However, whether or not condition (6) is fully correct is yet to be established, which is an important topic for further research.

The continuous case would enable many application models. For example, it would enable our combined method to be applied to regression models with the common assumption that the random variable ϵ is normally distributed, $\mathcal{N}(0, \sigma^2)$, where a gamma distribution Gamma(α, β) can be used as conjugate prior

for $1/\sigma^2$. Hence, the predictive probability density function after having n observations is of the form:

$$p(z|s,\gamma) = \frac{1}{\sqrt{\pi}} \frac{\Gamma\left(\frac{s+k+3}{2}\right)}{\Gamma\left(\frac{s+k+2}{2}\right)} \frac{(s\gamma+\tau_k)^{\frac{s+k+2}{2}}}{(s\gamma+\tau_k+z^2)^{\frac{s+k+3}{2}}},$$

where

$$\tau_k = \sum_{j=1}^{k} z_j^2 = \sum_{j=1}^{k} (y_i - f(\mathbf{X}_i, \mathbf{d}))^2 .$$

By using the imprecise Bayesian normal model [11], we can then construct the imprecise regression model combining imprecise Bayesian inference with maximum likelihood estimation as briefly discussed in Section 8 where only discrete random variables were used in line with the general presentation in this paper. So, establishing the detailed and fully justified generalization of the approach in this paper to continuous random variables is very important, and we are hopeful to report on this in the near future.

Acknowledgement

We thank referees for useful and detailed suggestions that improved the paper.

References

[1] J.-M. Bernard. An introduction to the imprecise Dirichlet model for multinomial data. *International Journal of Approximate Reasoning*, 39(2-5):123–150, 2005.

[2] J.M. Bernardo and A.F.M. Smith. *Bayesian Theory*. Wiley, Chichester, 1994.

[3] K.Y. Cai. Towards a conceptual framework of software run reliability modeling. *Information Sciences*, 126:137–163, 2000.

[4] S. Ferson. Bayesian methods in risk assessment. Technical report, RAMAS, 2006.

[5] A.L. Goel and K. Okomoto. Time dependent error detection rate model for software reliability and other performance measures. *IEEE Trans. Reliab.*, R-28:206–211, 1979.

[6] Z. Jelinski and P.B. Moranda. Software reliability research. In W. Greiberger, editor, *Statistical Computer Performance Evaluation*, pages 464–484. Academic Press, New York, 1972.

[7] N.L. Jonson, S. Kotz, and A.W. Kemp. *Univariate discrete distributions*. Wiley, New York, 1992.

[8] B. Littlewood and J. Verall. A Bayesian reliability growth model for computer software. *Applied Statistics*, 22:332–346, 1973.

[9] C. Loader. *Local regression and likelihood*. Springer-Verlag, New York, 1999.

[10] J.D. Musa, A. Iannino, and K. Okumoto. *Software Reliability: Mesurement, Prediction, Application*. McGraw-Hill, 1987.

[11] E. Quaeghebeur and G. de Cooman. Imprecise probability models for inference in exponential families. In J.-M. Bernard, T. Seidenfeld, and M. Zaffalon, editors, *Proc. of the 4rd Int. Symposium on Imprecise Probabilities and Their Applications, ISIPTA'05*, Pittsburgh, Pennsylvania, July 2005. Carnegie Mellon University.

[12] C.P. Robert. *The Bayesian Choice*. Springer, New York, 1994.

[13] G.J. Schick and R.W. Wolverton. An analysis of competing software reliability models. *IEEE Trans. on Software Engineering*, SE-4:104–120, 1978.

[14] A.R. Syversveen. Noninformative Bayesian priors. Interpretation and problems with construction and applications. Preprint Statistics 3, Department of Mathematical Sciences, NTNU, Trondheim, 1998.

[15] P. Walley. *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall, London, 1991.

[16] P. Walley. Inferences from multinomial data: Learning about a bag of marbles. *Journal of the Royal Statistical Society, Series B*, 58:3–57, 1996. with discussion.

[17] G. Walter, Th. Augustin, and A. Peters. Linear regression analysis under sets of conjugate priors. In G. de Cooman, J. Vejnarova, and M. Zaffalon, editors, *Proceedings of the Fifth International Symposium on Imprecise Probabilities and Their Applications*, pages 445–455, Prague, Czech Republic, 2007.

[18] M. Xie. *Software Reliability Modeling*. World Scientific, 1991.