JIRSS (2004)
Vol. 3, No. 1, pp 1-37

When Can Finite Testing Ensure Infinite Trustworthiness?

Nozer D. Singpurwalla, Philip Wilson

The George Washington University, Washington DC 20052, USA.

Abstract. In this paper we contribute to the general philosophical question as to whether empirical testing can ever prove a physical law. Problems that lead to this question arise under several contexts, and the matter has been addressed by the likes of Bayes and Laplace. After pointing out that a Bayesian approach is the proper way to address this problem, we show that the answer depends on what we start with. Namely, under certain prior assumptions, a finite amount of testing can lead to the conclusion of total trustworthiness, though such priors could be unrealistic. However, we do produce a new class of priors under which a finite amount of testing can lead to a high degree of trustworthiness, at a relatively fast pace. We use the scenario of software testing as a way to motivate and discuss our development.

1 Introduction

This paper discusses the general problem of whether a large but finite amount of Bernoulli testing of similar items, all of which result

Received: November 2003
Key words and phrases: Bayes' law, discrete priors, Jeffreys' prior, reliability, sample size, software testing, stockpile stewardship.

in successes, can lead to the judgement of a high trustworthiness of all future items to be tested. The problem dates back to Bayes and to Laplace (1774), and appears under several disguises. To philosophers it is a problem of induction; namely, is it possible to claim that a scientific law is true based on empirical observations alone? To statisticians such as Karl Pearson (1920), this is a "Fundamental Problem of Practical Statistics". To biometricians it is a matter of asserting the effectiveness of a drug for use by the population as a whole, based on the results of a substantive testing protocol. To reliabilists charged with the task of stockpile stewardship and certifying the trustworthiness of mission critical systems, it is a question of how much testing is enough, and how to avoid indiscriminate testing. To provide a point of focus to the general problem described above, we will consider here the scenario of software testing to ensure its credibility. Indeed, this is what has motivated our interest in this problem. In particular, a piece of software code can receive N distinct inputs, where N is large, conceptually infinite. Each input is processed by the code and this results in a success if done correctly, or a failure if done incorrectly. The software is intended to be used in a life-critical environment such as an air traffic control system. Thus the consequences of a failure to correctly process an input are severe: that is, no failures can be tolerated. We are therefore required to ensure that the software is fail-safe, and to do so we are allowed to test n sample inputs.

2 The Problem's Architecture

Let R denote the number of inputs out of N that can be successfully processed. Clearly, $0 \leq R \leq N$, but R is unknown. However, we would like to know if $R = N$. For this, we randomly select n of the N inputs and test these for success or failure. Suppose that n is large and that all the n tests result in successes. Based on the above, can we claim that $R = N$?

The answer to the above question is a no, because the only way we can make such a claim is to test all the N inputs, and since N is prohibitively large, this we cannot do. However, whilst $R = N$ cannot be guaranteed it can be given as a high probability as we desire. Thus we would like to claim that $P(R = N) \approx 1$, and this is what we mean when we say (in the title of this paper) "infinite trustworthiness". Equivalently, we ask if the software's reliability is

close to one against all its future distinct inputs.

In actuality, the software scenario is much more complicated than the bare essentials that we have laid out above. For a deeper appreciation of the caveats and nuances of software testing we refer the reader to Miller et al. (1992). However, to better align our statement of the problem with some notions and verbiage used by software engineers, we introduce the following as assumptions:

i) The software has many lines of code and has so many interconnected modules that for testing purposes it is best treated as a "black-box"[1]. In essence this means that every input is of equal importance and all that matters is whether the input is processed successfully or not. From a statistical point of view, this assumption tantamounts to the judgment that all the binary outputs are *exchangeable* [see de Finetti (1974), p. 215].

ii) The n inputs used to test the software are selected randomly, with equal probability of selection. In the language of software engineering, this means that the software system has an "operational profile" that is uniform.

We emphasise that even though we have used the scenario of software testing to describe the architecture of our problem, the set-up is general enough to embrace many other contexts involving Bernoulli testing. After all, the essence of assumptions *i*) and *ii*) above is that the trials are (conditionally) independent and that sampling is random. The fact that N is conceptually infinite may not be germane to all scenarios (stockpile stewardship being an example) but it certainly does hold in the context of releasing a drug for general use (like aspirin), or for establishing the truth of a physical law (like the sun always rises). In any case, arguments based on limiting operations provide guidance about the general nature of a result; thus the conclusions of this paper would be useful, even if N is finite (but large).

3 Possible Approaches

The problem posed in Section 2 can be addressed via either one of the two paradigms of statistical inference: frequentist (or sample theo-

[1] The term "black-box" originated during World War II. It encapsulates a passive approach to complex systems which cannot be directly observed.

retic) and Bayesian (or Laplacian). Each approach can claim its merits over the other and each approach tends to address the problem in its own way. However, for reasons given below, our focus here will be on the second approach.

3.1 The Frequentist Approach

In principle, given the circumstances of the problem, the frequentist approach is not designed to assess a probability of the type $P(R = N)$. Under the dictates of this approach probabilities can only be estimated for sequences that are embedded in a collective (or ensemble). This boils down to conceptualizing a large collective of software codes, all of which are similar to each other, and each of which can receive N distinct inputs of the type mentioned before. If the software code is one of a kind, as is usually the case, it would be far fetched to think in terms of a collective of software codes. Thus a frequentist approach is unable to answer the question posed in Section 2. However, under this approach it is possible to obtain an estimate of R, say $\widehat{R}$, by first estimating the probability that any randomly selected input is a success and then multiplying this estimated probability by N. Since each trial can be seen as being a member of an ensemble of size N, the aforementioned probability can be estimated as

$$\frac{\text{total number of successes in } n \text{ trials}}{\text{total number of trials } (=n)}.$$

When all the n trials result in a success, the estimated probability is one. Consequently, $\widehat{R} = N$ – a result whose value to a user is suspect. All the same, the frequentist approach is able to provide a lower confidence limit on the quantity R/N and this could serve as a proxy for infallibility. However, confidence limits are not to be interpreted as coverage probabilities on unknown quantities, and thus the extent to which a frequentist approach can take place is limited. For more details on the kind of results that can be obtained under the frequentist approach for the kind of problems considered here, we refer the reader to Launer (2003).

3.2 The Bayesian Approach

The Bayesian approach requires of a user the specification of a prior distribution on the unknown R, and the conclusions of this approach

depend on the prior distribution used. This is strikingly true for the problem considered here. Three noteworthy priors have been previously proposed; these are:

A. The Bayes-Laplace (Uniform) Prior,

B. A Prior by Jefferys,

C. A Prior by Bernardo.

In this paper we shall first interpret these priors vis-à-vis the nature of results that they provide, and then introduce four new priors each of which produces results with a different flavor. Our proposed priors are:

D. A Pessimistic Prior,

E. A Regulated Pessimistic Prior,

F. A Scale Prior, and

G. A Portmanteau Prior.

The pessimistic prior represents a significant departure from the previously proposed three priors, and provides some intriguing results. The regulated pessimistic prior is a variant on the pessimistic prior and provides an alternative to the latter. The scale prior modulates a feature of the first three priors, and the portmanteau prior combines the pessimistic and scale priors to encapsulate prior beliefs that seem meaningful.

The Bayesian approach is able to provide an answer to the question we have posed, namely, what is $P(R = N)$ given n trials, all of which result in success? In what follows we describe the priors mentioned above and the results they provide. We start by first discussing a model for the likelihood, which is the hypergeometric distribution, and the general nature of $P(R = N)$ given all successes in n trials.

4 The Hypergeometric Model and the Posterior Probability

Since testing consists of selecting n inputs without replacement and observing whether or not they result in successes, the distribution to

use in this problem is the hypergeometric distribution. With R fixed but unknown, the probability that out of the n inputs considered t are successful is given by

$$P(T=t|R) = \begin{cases} \frac{\binom{R}{t}\binom{N-R}{n-t}}{\binom{N}{n}}, & t = 0, 1, ..., \min(n, R) \\ \\ 0, & \text{otherwise.} \end{cases}$$

We use this as a likelihood for R. Specifically, we are assuming that $t = n$, all the tests are successful, which leads to the likelihood,

$$P(T=n|R) = \begin{cases} \frac{\binom{R}{n}}{\binom{N}{n}}, & R = n, n+1, ..., N \\ \\ 0, & R = 0, 1, ..., n-1. \end{cases}$$

Our aim is to find $P(R = N|T = n)$. In this approach we need to specify $P(R = r)$, a prior for R. Then by an application of Bayes' Law,

$$\begin{aligned} P\left(R=N|T=n\right) &= \frac{P(T=n|R=N)P(R=N)}{P(T=n)} \\ &= \frac{P(T=n|R=N)P(R=N)}{\sum_{r=0}^{N} P(T=n|R=r)P(R=r)} \\ &= \frac{\frac{\binom{N}{n}}{\binom{N}{n}}P(R=N)}{\sum_{r=0}^{n-1} 0 \cdot P(R=r) + \sum_{r=n}^{N} \frac{\binom{r}{n}}{\binom{N}{n}}P(R=r)}. \end{aligned}$$

Thus we have,

$$P\left(R=N|T=n\right) = \frac{P(R=N)}{\sum_{r=n}^{N} \frac{\binom{r}{n}}{\binom{N}{n}}P(R=r)}, \tag{1}$$

which is a simple method for finding $P(R = N|T = n, N)$ that only requires a prior for R.

5 Possible Priors for the Number of Failures

5.1 Bayes-Laplace Prior

This strategy involves expressing prior indifference among all the values that R can take, namely, $R = 0, 1, ..., N$, and so assigns a uniform prior on R, see Figure 1. This is in accordance with the Bayes-Laplace principle of insufficient reason. Thus:

$$P(R = r) = \frac{1}{N+1}, \qquad r = 0, 1, ..., N.$$

This is also known as an "objective" or "ignorance" or "public" prior.

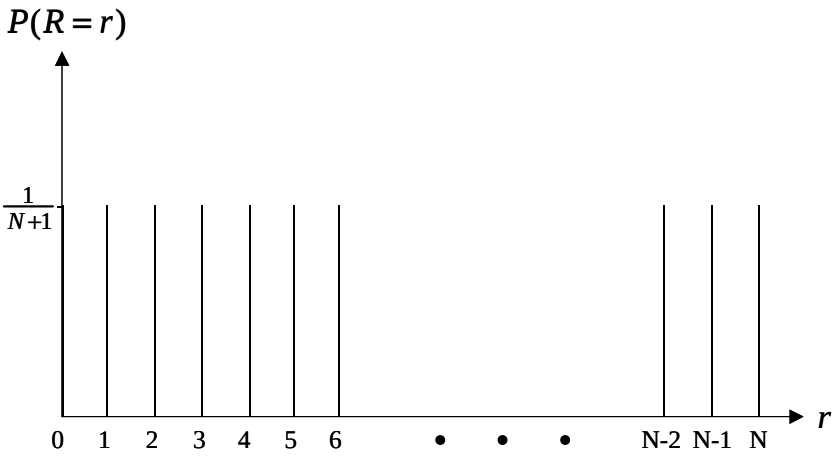

Figure 1: Uniform Prior

$R = 0$ corresponds to the case that every trial is a failure. This can happen when there is a flaw in a path taken by all N inputs. $R = 1$ can happen when practically all inputs choose a path with a flaw, and so on for $R = 2, 3, ..., N$. However the prior probability of $(N + 1)^{-1}$ assigned to each R is small since N is large.

We can go on to calculate the probability that $R = N$, namely

$$\begin{aligned} P\left(R = N | T = n\right) &= \frac{P(R=N)}{\sum_{r=n}^{N} \frac{\binom{r}{n}}{\binom{N}{n}} P(R=r)} \\ &= \frac{\frac{1}{N+1}}{\sum_{r=n}^{N} \frac{\binom{r}{n}}{\binom{N}{n}} \frac{1}{N+1}} \\ &= \frac{n+1}{N+1}. \end{aligned}$$

If we are interested in whether the next input will result in success, this is equivalent to putting $N = n + 1$, so that

$$P\left((n+1)^{\text{th}} \text{ input success} | T = n\right) = \frac{n+1}{n+2}.$$

Also $\frac{n+1}{n+2} \to 1$ as $n \to \infty$, so that the probability that the next input results in success tends to 1 as the number of successful tests also increases. Indeed the probability that the next k inputs result in successes also approaches 1.

However, for any fixed but large n, $\lim_{N\to\infty} \frac{n+1}{N+1} = 0$. Thus, irrespective of how favorable our experience has been over a large number of trials, if the number of future trials is big compared to the number of tests, then the probability that they will all be successful is close to zero. In particular, even if we observe **all** successes during testing, the probability that all future tests will be successful is zero. This result goes against the grain of scientists. This state of affairs provides a motivation for finding a suitable prior that is in accord with the disposition of scientists.

5.2 Jeffrey's Prior

The astronomer-mathematician-philosopher, Harold Jeffreys (1961) proposed the following as his prior on R:

$$P(R=r) = \begin{cases} \frac{1-2k}{N+1}, & r = 1, ..., N-1 \\ \frac{1-2k}{N+1} + k, & r = 0, N, \end{cases}$$

$0 < k \leq \frac{1}{2}$. For $k = 0$, Jeffrey's prior reduces to the Bayes-Laplace prior.

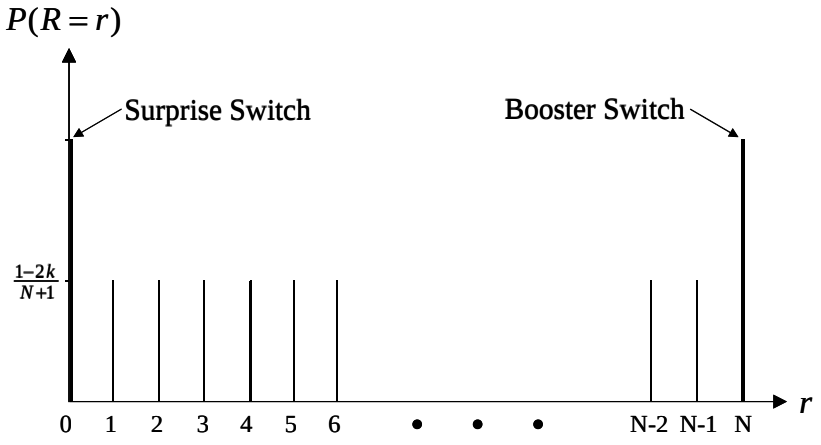

Figure 2: Jeffrey's Prior

Jeffrey's prior puts a probability mass of k on $R = 0$ and $R = N$, and then spreads the balance of $(1 - 2k)$ over all the $N + 1$ points, $R = 0, 1, ..., N$, giving the points $R = 0$ and N a mass of $k + \frac{1-2k}{N+1}$. We call the probability mass at $R = 0$ a ***surprise switch***, and the probability mass at $R = N$ a ***booster switch***. The reasons for this are discussed in more detail in Section 5.4. While for the Bayes-Laplace prior, the probabilities for $R = 0$ and N tend to zero as N increases, along with the probabilities for the other values of R, this is not the case with Jeffrey's prior. This prior gives a significant amount of mass to the two extreme possibilities, namely the software always works and the software always fails, and spreads the rest out uniformly. The surprise switch at $R = 0$ encapsulates the possibility of a fundamental flaw in the software that always leads to a failure, while the booster switch at $R = N$ encapsulates the possibility that no flaw exists at all.

Using Equation 1 we can calculate the posterior probability of all

N inputs resulting in success as

$$P\left(R=N|T=n\right)$$

$$= \begin{cases} \dfrac{\frac{1-2k}{N+1}+k}{\sum_{r=n}^{N-1}\left(\frac{\binom{r}{n}}{\binom{N}{n}}\frac{1-2k}{N+1}\right)+\frac{1-2k}{N+1}+k}, & n>0 \\ \dfrac{\dfrac{1-2k}{N+1}+k}{\frac{1-2k}{N+1}+k+\sum_{r=1}^{N-1}\frac{1-2k}{N+1}+\frac{1-2k}{N+1}+k}, & n=0 \end{cases}$$

$$= \begin{cases} \dfrac{(n+1)\left[(N-1)k+1\right]}{(n+1)\left[(N-1)k+1\right]+(N-n)(1-2k)}, & n>0 \\ \dfrac{1-2k}{N+1}+k, & n=0. \end{cases}$$

We also note that, for any $n>0$,

$$\lim_{N\to\infty} P\left(R=N|T=n\right)=\frac{(n+1)k}{(n+1)k+1-2k}\neq 0, \tag{2}$$

unlike the situation with the Bayes-Laplace prior. For $n=1$,

$$\lim_{N\to\infty} P\left(R=N|T=n\right)=2k,$$

thus the probability that the software is infallible doubles when the very first test is sucessful.

For $k=\dfrac{1}{4}$,

$$\lim_{N\to\infty} P\left(R=N|T=n\right)=\frac{n+1}{n+3},$$

which increases in n, see Figure 3.

5.3 Bernardo's Prior

Using information-theoretic arguments, Bernardo (1985) proposes a class of priors called the *reference priors*. His arguments lead to the prior

$$P(R=r)=\begin{cases} \dfrac{1-k}{N}, & r=0,1,2,...,N-1 \\ k, & r=N. \end{cases}$$

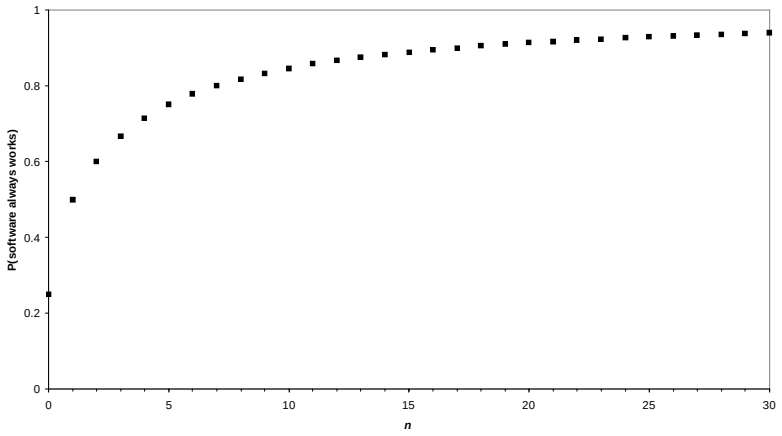

Figure 3: $P(\text{Software Never Fails} \mid T = n)$ for Jeffrey's Prior With $k = \frac{1}{4}$

Figure 3: $P(\text{Software Never Fails}|T = n)$ for Jeffrey's Prior With $k = {}^{1}$

Using this prior, the posterior probability that the software successfully processes all inputs as

$$P\left(R = N|T = n\right) = \frac{(n+1)Nk}{(n+1)Nk + (N-n)(1-k)},$$

which leads to

$$\lim_{N\to\infty} P\left(R = N|T = n\right) = \frac{(n+1)k}{(n+1)k + (1-k)}. \tag{3}$$

Bernardo's prior does not put a large point mass at $R = 0$, which is what Jeffrey does. In putting the k of Jeffrey's prior equal to $1/4$, we have considered the case where the prior distributes a probability mass of $1/2$ between the extreme points and the rest uniformly over the remaining points. Bernardo's prior only has one large point mass, and so if we choose to put probability mass of $1/2$ at $R = N$, then we get

$$P\left(R = N|T = n\right) = \frac{(n+1)N}{(n+1)N + (N-n)}.$$

Consequently $\lim_{N\to\infty} P\left(R = N|T = n\right) = \frac{n+1}{n+2}$, which increases in n, but faster than the equivalent result of $\frac{n+1}{n+3}$ for Jeffrey's prior with $k = \frac{1}{4}$. This is because with $k = \frac{1}{2}$, Bernardo puts more prior weight on $R = N$ than Jeffreys, who puts a weight of approximately $\frac{1}{4}$.

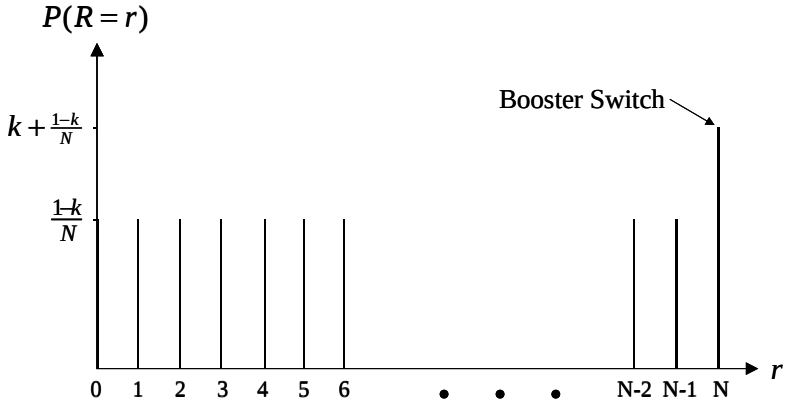

Figure 4: Bernardo's Prior

However if Bernardo were to pick $k = \frac{1}{4}$, so that $P(R = N)$ is almost the same as that for Jeffreys' prior, then the priors would be more compatible, and $\lim_{N\to\infty} P(R = N|T = n) = \frac{n+1}{n+4}$. This would tend to 1 slower than that under Jeffreys' prior.

The reason for this is that although $P(R = N)$ is almost the same for both priors, Jeffrey's prior has a *surprise switch* at $R = 0$, which Bernardo's prior does not.

Having now considered the priors that have been suggested, we move on to develop some new priors. These will allow us to express a wider range of prior opinions, and will also lead to interesting and useful results.

5.4 A Pessimistic Prior

"Pessimists" are those who expect that things will go wrong. "Wise Pessimists" are those who make provision for surprises. Thus they build into their models two switches, at $R = 0$ a *surprise switch* and at $R = N$ a *booster switch*. These switches facilitate mood swing between pessimism and optimism. The more powerful the switch, the quicker the moodswing. "pure optimists" provide for a weak surprise switch and a strong booster switch. They expect good things to happen, and are thus not surprised when they encounter a slew of successes. Thus their mood swings are slow.

Putting a probability mass at $R = 0$ tantamounts to installing a "surprise switch", while putting a probability mass at $R = N$ boils

down to installing a "booster switch". The priors of Jeffreys and Bernardo both have strong booster switches, and with $k = \frac{1}{4}$ for both priors the success switches are compatible. However Jeffreys' surprise switch is stronger than Bernardo's, which tends to zero as N increases. Thus the success of all n trials is no surprise to Bernardo (the pure optimist), but for Jeffrey's (the wise pessimist), success at $n = 1$ causes him to annihilate all the probability mass at $R = 0$, and redistribute this over $R = 1, 2, ..., N$, proportionately. Consequently his success switch (at $R = N$) gains more probability mass than Bernardo's booster switch, resulting in a faster convergence of $P(R = N|T = n)$ to 1 of Jeffrey's prior than Bernardo's prior.

On the other hand, the Bayes-Laplace prior is devoid of personality. It's surprise switch, and booster switch are both weak, so that even an abundance of good news cannot fundamentally change the mood of the prior to optimism, and as a consequence,

$$\lim_{N\to\infty} P\left(R = N|T = n\right) = 0.$$

We now go on to use the above concepts of the booster switch and the surprise switch to motivate a prior that is wisely pessimistic, but can react quickly to even a very small number of successes. This could save on the amount of testing software would have to go through, which in turn would save money.

Instead of having a single surprise switch at $R = 0$, the prior proposed here has several suprise switches in the vicinity of $R = 0$, and a small success switch at $R = N$. The surprise switches are decreasing exponentially. The switch at $R = 0$ encapsulates the possibility that the software has a flaw in a path encountered by all inputs. The switch at $R = 1$ encapsulates the possibility of a flaw in a path that is rarely avoided. Similarly the success switch at N is associated with there being no flaw and the mass at $N-1$ relates to there being a flaw that is rarely encountered, etc. One way to construct such a prior is via a geometric distribution with support on $R = 0, 1, ..., N - 1$, i.e. starting at $R = 0$, and truncated at $N - 1$, with the addition of a point mass of $N^{-\lambda}$, $\lambda > 0$, at $R = N$ (see Figure 5). Even though this point mass tends to 0 as N increases, we shall still call it a booster switch, because it functions in a similar way to the surprise switch in the priors of Jeffreys and Bernardo. Specifically, we define the pessimistic prior by

$$P(R=r)=\begin{cases} C_N\,(1-q)\,q^r, & r=0,1,...,N-1 \\ N^{-\lambda}, & r=N, \end{cases} \tag{4}$$

with $q \in (0,1)$, $\lambda > 0$, and $C_N = \dfrac{1-N^{-\lambda}}{1-q^N}$. Note that $C_N \to 1$ as $N \to \infty$. Thus $\lim\limits_{N\to\infty} P\,(R=N) = (1-q)\,q^r$.

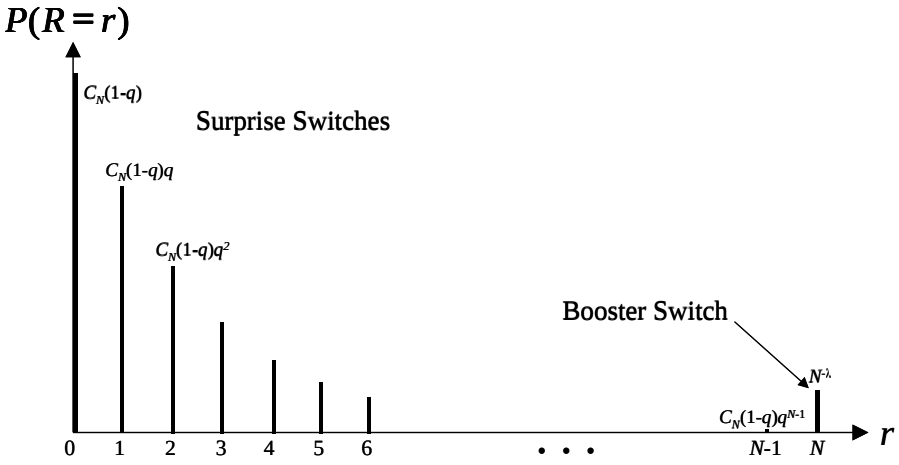

Figure 5: Pessimistic Prior

For finite N the predictive distribution is cumbersome. However for finite n and $N \to \infty$,

$$\lim_{N\to\infty} P\,(R=N|T=n)=\begin{cases} 1, & n>\lambda \\ \left[1+n!\left(\frac{q}{1-q}\right)^n\right]^{-1}, & n=\lambda \\ 0, & n<\lambda. \end{cases} \tag{5}$$

Thus for example, for $\lambda = 1$, with only one input leading to a success, the software's trustworthiness increases from 0 to $1-q$, and with 2 inputs both being successful, the trustworthiness of the software is assertively ensured.

The cumbersome predictive distribution of R, for $n = 2$ and N

finite is,

$$P(R = N|T = n = 2) =$$
$$\left[1 + \frac{N^{\lambda-1}}{N-1}\frac{2q^2 - q^N\left(q^2(N-1)(N-2) - 2qN(N-2) + N(N-1)\right)}{(1-q^N)(1-q)^2}\right]^{-1}.$$

It can easily be verified that taking the limit as $N \to \infty$, for different values of λ, will produce the results given in Equation 5, and illustrated in Figure 6.

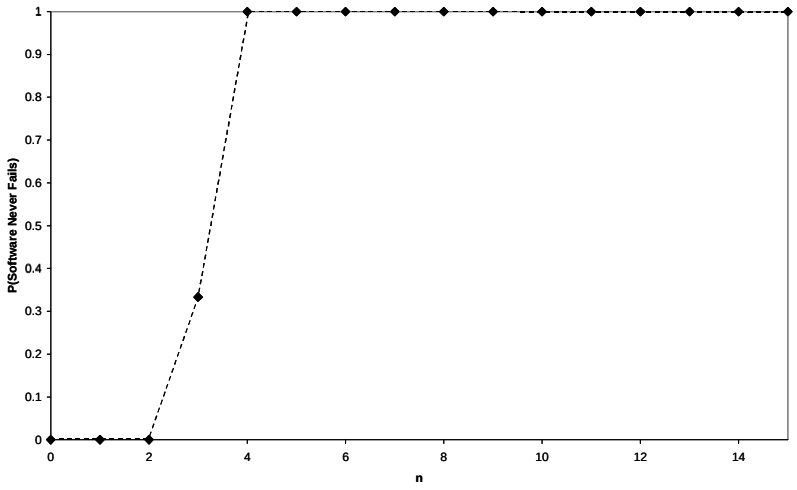

Figure 6: $P(\text{Software Never Fails}|\ T = n)$ for Pessimistic Prior with $\lambda = 3$ and $q = 0.5$

Thus with the pessimistic prior, we have an answer to the question posed in the title of this paper. Finite testing can assure infinite trustworthiness, when the prior used is a pessimistic prior of the type given by Equation 4, and illustrated in Figure 5. A result like this need not be too surprising. Its analogue is the tossing of a coin which has either heads on both sides or tails on both sides, but one does not know which. All it takes is a single toss to resolve the uncertainty and claim that all future tosses of the coin will yield a head (or a tail). The prior we have described uses the geometric distribution, and adds to this a booster switch. However, any discrete probability distribution with support on $R = 0, 1, ...,$ and a booster switch at $R = N$ can be used, and the behavior will be essentially the same.

An objection to this prior is that it is too extreme in that after only a few successful tests, the software is said to be guaranteed to work. To put it another way, to some people finite testing should not result in infinite trustworthiness. We next consider a modification of the pessimistic prior that avoids this objection and at the same time provides an improvement over the Jeffreys-Bernardo results.

5.5 Regulated Pessimistic Prior

As it stands, this pessimistic prior suffers from the problem that if there are two succeses, the conclusion is that software will always work with probability 1. The problem occurs because our prior essentially asserts that the software either almost always fails, or it never fails. It is necessary to add a component to the prior to represent the middle ground regarding the software occasionally failing. For this purpose we add a switch at $N-1$. This switch competes with the success switch, and consequently it reduces $\lim_{N\to\infty} P\left(R=N|T=n\right)$. Thus this switch is called a *regulatory switch*, as it regulates the probability of the software always working. It ensures that no finite amount of testing can ever lead to the conclusion that the software is guaranteed to work. The regulatory switch must decrease at the same rate as the success switch, so that both switches will function. Thus the regulatory switch is chosen to be a multiple A times the size of the booster switch. Typically A will be small, but in general the only requirement on A is that it is positive. Thus we have the prior,

$$P(R=r)=\begin{cases} C_N\left(1-q\right)q^r, & r=0,1,...,N-1 \\ AN^{-\lambda}, & r=N-1 \\ N^{-\lambda}, & r=N. \end{cases} \tag{6}$$

with $q\in(0,1)$, $\lambda>0$, $0<A<1$, and $C_N=\dfrac{1-(1+A)N^{-\lambda}}{1-q^N}$, see Figure 7.

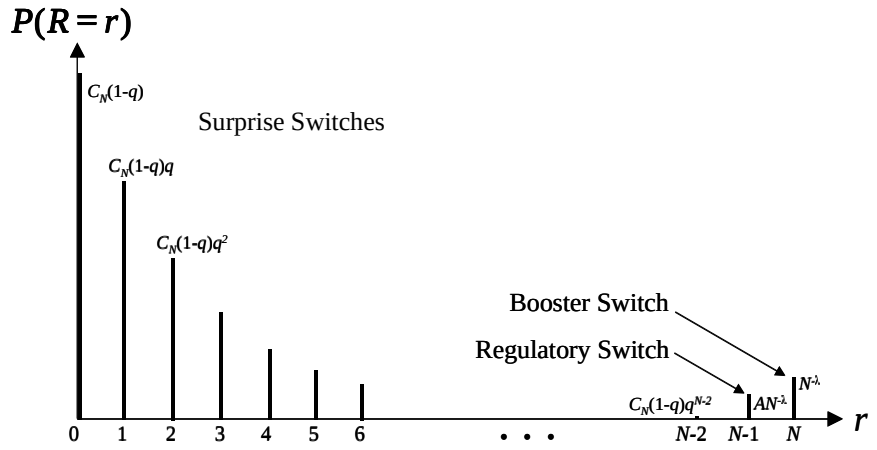

Figure 7: Regulated Pessimistic Prior

This now leads to,

$$\lim_{N\to\infty} P\left(R = N|T = n\right) = \begin{cases} \left[1+A\right]^{-1}, & n > \lambda \\ \left[1 + A + n!\left(\frac{q}{1-q}\right)^n\right]^{-1}, & n = \lambda \\ 0, & n < \lambda. \end{cases} \quad (7)$$

This is illustrated in Figure 8 for $A = \frac{1}{10}$, $q = \frac{1}{2}$, and $\lambda = 2$.

Equation 7 shows that the regulated pessimistic prior leads to a satisfactory result. After sufficient successful trials (the necessary number is determined by λ), $\lim_{N\to\infty} P\left(R = N|T = n\right)$ goes from 0 to $\left[1 + A + n!\left(\frac{q}{1-q}\right)^n\right]^{-1}$ (if λ is an integer) and then after a further success it reaches $\left[1+A\right]^{-1}$, which is where it remains as long as future tests are sucessful. However this probability does not increase for additional successful testing, even though this would be expected to lead to greater confidence in the reliability of the software, as occurs with Bernardo's and Jeffreys' priors. Thus we now turn to a generalized version of these priors that will useful in solving this problem.

5.6 Scale Priors

Jeffreys' and Bernardo's priors are based on the uniform prior, with the addition of switches corresponding to prior opinions concerning

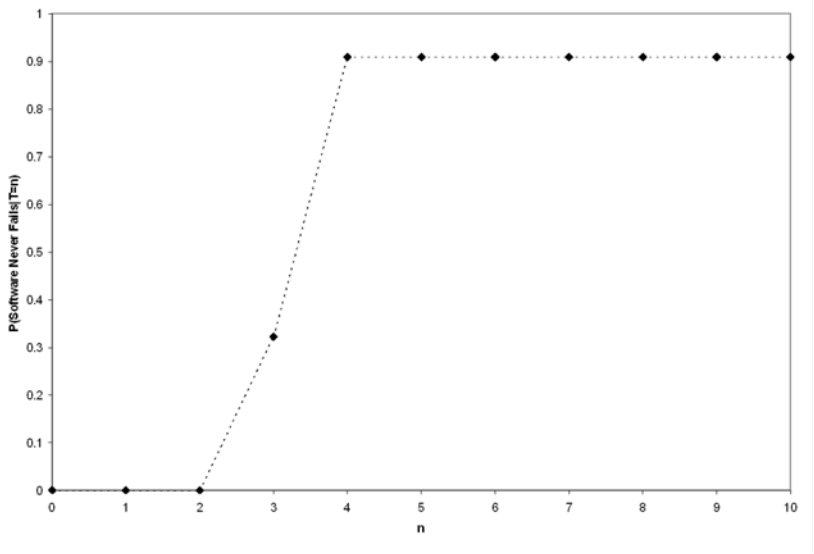

Figure 8: $P(\text{Software Never Fails}|\ T = n)$ for Pessimistic Prior with $A = 0.1$, $\lambda = 2$ and $q = 0.5$

the probabilities of the software always working and always failing. However in the situation where software has been designed to be extremely reliable, it is hard to justify using the uniform distribution, especially where testing has already been performed on components or previous versions of the software. In particular, the software might be expected to be much more likely to work most of the time than rarely, since serious programming errors would be carefully avoided, and any that do occur should be easily identified and eliminated. One way that such opinion may be expressed is with statements such as, 1/3 of the inputs resulting in failure is twice as likely as 2/3 of the inputs resulting in failure. Such belief is consistent with a scale prior Suppose that $f(x)$ is a non-negative function bounded above on $[0, 1]$. Then f can be used to define the prior

$$P(R = r) = \begin{cases} S_N, & r = 0 \\ C_N\, f(\frac{r}{N}), & r = 1, ..., N-1 \\ B_N, & r = N, \end{cases} \tag{8}$$

where $C_N = \dfrac{1 - S_N - B_N}{\sum_{r=1}^{N-1} f(\frac{r}{N})}$, $S_N \geq 0$, $B_N \geq 0$, $S_N + B_N \leq 1$. S_N and B_N are the surprise switch and the booster switch respectively. In the interest of generality we allow S_N and B_N to be different.

The priors of Jeffreys and Bernardo, as well as the uniform prior are all examples of scaling priors with $f(x) = 1$. They all express indifference about $r = 1, 2, ..., N-1$. Instead of this we may wish to use a prior that incorporates our knowledge about the numbers of failures, for example $f(x) = x^2$, see Figure 9.

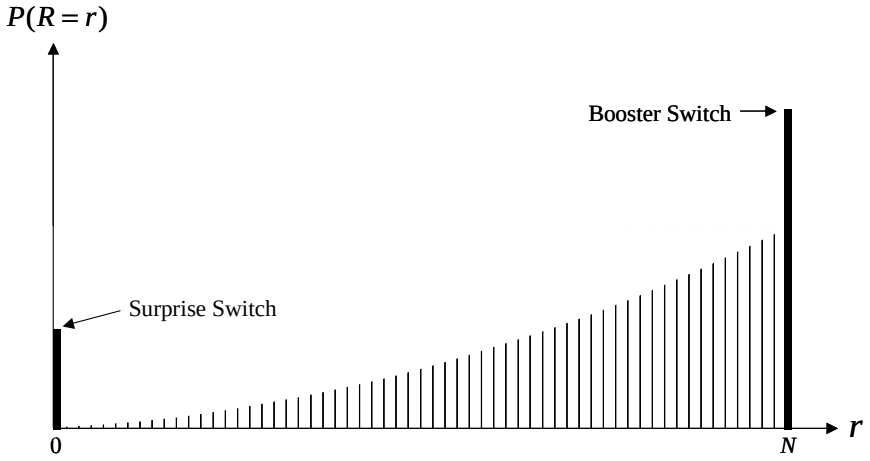

Figure 9: Scale Prior with $f(x) = x^2$

Use of Equation 1 for the scalar prior leads to

$$\begin{aligned}
&\lim_{N\to\infty} P\left(R = N | T = n\right) \\
&\quad = \lim_{N\to\infty} \left[\frac{1 - S_N - B_N}{B_N} \frac{\sum_{r=1}^{N-1} \frac{r(r-1)...(r-n+1)}{N(N-1)...(N-n+1)} f(\frac{r}{N})}{\sum_{r=0}^{N-1} f(\frac{r}{N})} + 1 \right]^{-1} \\
&\quad = \lim_{N\to\infty} \left[\frac{1 - S_N - B_N}{B_N} \frac{\int_0^1 x^n f(x) dx}{\int_0^1 f(x) dx} + 1 \right]^{-1}.
\end{aligned}$$

If we let $\lim_{N\to\infty} S_N = S$ and $\lim_{N\to\infty} B_N = B$, then,

$$\lim_{N\to\infty} P(R = N|T = n) = \begin{cases} \left[\dfrac{1-S-B}{B}\dfrac{\int_0^1 x^n f(x)dx}{\int_0^1 f(x)dx} + 1\right]^{-1}, & S > 0 \\ 0, & S = 0. \end{cases} \tag{9}$$

It can be easily verified that this leads to Equations 2 and 3. Since Equation 9 is given in terms of integrals, it provides an easy method for calculating trustworthiness, even for scale priors based on complicated $f(x)$. Also, since $\int_0^1 x^n f(x)dx \to 0$, as $n \to \infty$, it shows that $\lim_{N\to\infty} P(R = N|T = n) \to 1$ as $n \to \infty$, for any scale prior with $L_s > 0$.

5.7 Portmanteau Prior

The scale prior has the benefit that $\lim_{N\to\infty} P(R = N|T = n)$ tends to 1 as n increases, while the regulated pessimistic prior jumps very quickly to a value that can arbitrarily close to 1 after only a few sucessful trials, however it does not increase further. The final prior that we propose is a hybrid of these two priors, incorporating features from both and leading to the desirable properties of both, namely $\lim_{N\to\infty} P(R = N|T = n)$ exhibits a sudden increase to close to 1 after only a few trials, and then convergence to 1 as n increases further.

This prior is formed from a pessimistic prior by adding a scale section from $R = \frac{9N}{10} - 1$ to $N-1$. This acts like a regulatory switch in that it prevents $\lim_{N\to\infty} P(R = N|T = n)$ from jumping straight up to 1, but it retains the feature of the scale priors that $\lim_{N\to\infty} P(R = N|T = n)$ also increases to 1 as n increases.

We shall use $f(x) = kx^2$, $k > 0$, as the function for the scale component. The function is evaluated for values of x, $0 < x \leq 1$. We avoid evaluating the function at zero in order to ensure that none of the resulting probabilities are 0. In principle any bounded function could be used, although it makes sense to use a function that increases

from 0.

$$P(R=r) = \begin{cases} C_N\,(1-q)\,q^r, & r = 0, 1, ..., \lceil \frac{9N}{10} \rceil - 1 \\ k\left(\frac{10(r+1)}{N} - 9\right)^2 N^{-\lambda-1}, & r = \lceil \frac{9N}{10} \rceil, ..., N-1 \\ N^{-\lambda}, & r = N, \end{cases} \tag{10}$$

with $q \in (0,1)$, $\lambda > 0$, and

$$C_N = \frac{1 - kN^{-\lambda} - \sum_{r=\lceil 9N/10 \rceil}^{N-1} \left(\frac{10(r+1)}{N} - 9\right)^2 N^{-\lambda-1}}{1-q^N}.$$

(see Figure 10).

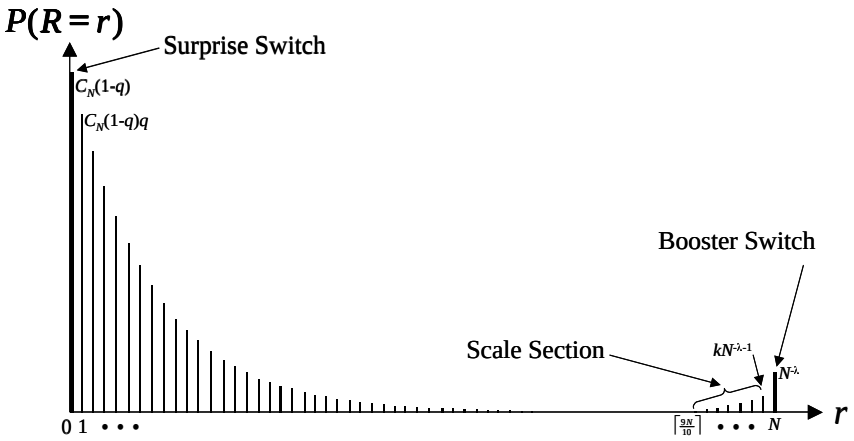

Figure 10: Portmanteau Prior

The factor k, $k > 0$, is equivalent to T, the size of the regulatory switch in the regulated pessimistic prior, and its effect is the same in shrinking $\lim_{N\to\infty} P(R = N|T = n)$ and so slowing its convergence to 1 as n increases. This convergence would also be affected by choosing a different $f(x)$, as in the section on scale priors. It is worth emphasizing that there are two variables for which convergence occurs. In the first place $N \to \infty$ leads to statements about the software never failing, while we are also considering $n \to \infty$, which relates to further successful testing.

The final result for the probability that the software will never fail if n tests have proven successful, using the combined prior is

$$\lim_{N\to\infty} P\,(R = N|T = n) =$$

$$\begin{cases} \left[1 + \int_0^1 f(x)\left(\frac{x+9}{10}\right)^n dx\right]^{-1}, & n > \lambda \\ \left[1 + n!\left(\frac{q}{1-q}\right)^n + \int_0^1 f(x)\left(\frac{x+9}{10}\right)^n dx\right]^{-1}, & n = \lambda \\ 0, & n < \lambda. \end{cases}$$

Note that since $\int_0^1 f(x)\left(\frac{x+9}{10}\right)^n dx \to 0$ as $n \to \infty$,

$$\lim_{N\to\infty} P\,(R = N|T = n) \to 1 \;\text{ as }\; n \to \infty.$$

Figure 11 illustrates $\lim_{N\to\infty} P\,(R = N|T = n)$ for $f(x) = 5x^2$.

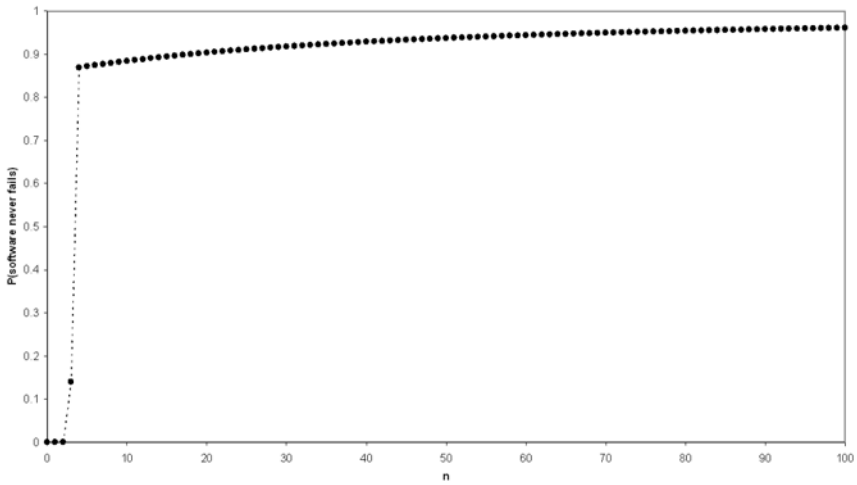

Figure 11: P(Software Never Fails$|\ T = n$) for combined prior with $f(x) = 5x^2$, $\lambda = 3$, $q = 0.5$

Thus the combined prior leads to the conclusion of a high probability of infallibility, after only a few successful tests, while retaining the desirable property that this probability of infallibility increases further towards one with additional successful testing. This means

that the software may be accepted as sufficiently reliable with only a small amount of testing.

6 Conclusion

Ensuring an item's trustworthiness (or reliability) by testing is very much dependent on your prior opinions. Realistic results can be obtained by using a prior where the probabilities of the extreme points do not tend to zero as N increases, as with the priors of Jeffreys and Bernardo. However, short amounts of testing can only be justified under certain prior beliefs. The pessimistic prior does ensure infinite trustworthiness with only finite testing, however for some this is unrealistic. The portmanteau prior on the other hand provides high trustworthiness after limited testing, and is more realistic because as the amount of successful testing increases the probability of infinite trustworthiness also increases.

Acknowledgments

Sir David Cox drew our attention to the paper of Pearson. We thank him for alerting us to this important reference. The work was supported by The U.S. Army Research Office, Grants DAAD19-01-1-0502 under a MURI and DAAD19-02-1-0195.

References

Bernardo, J. M. (1985), On a famous problem of induction. Trabajos de Estadística, **36**, 1:24-30.

de Finetti, B. (1974), Theory of Probability; A Critical Introductory Treatment. vol. 2, London, New York: Wiley.

Jeffreys, H. (1961), Theory of Probability. 3rd ed., Oxford, England: Clarendon Press.

Laplace, P. S. (1774), Mémoire sur la probabilité des causes par les évenemens. Mémoires de mathématique et de physique presentés à l'Acad émie royale des sciences, par divers savans, & lûs dans ses assembl ées, 6: 621-656.

Launer, R. (2003), Choosing the Prior for Zero Defects. (in progress).

Miller, K. W., Morell, L. J., Noonan, L. J., Park, L. J., Nicol, D. M., Murrill, B. W., and Voas, M. (1992), Estimating the Probability of Failure When Testing Reveals No Failures. IEEE Transactions on Software Engineering, **18**, 1: 33-43.

Pearson, K. (1920), The Fundamantal Problem of Practical Statistics. Biometrika, **13**, 1: 1-16.

Comment

José M. Bernardo

Universitat de València, Spain.

I have very much enjoyed this thought provoking paper on a very important problem. The comments below are intended to expand the discussion on the interpretation of the alternative priors discussed.

The authors make very clear that determination of

$$\Pr[R = N \mid r = n]$$

is indeed a fundamental problem of practical statistics, that this problem cannot be properly addressed from a frequentist viewpoint, and that the Bayesian solution, immediate once a prior distribution for $R \in \{0, 1, \ldots, N\}$ has been assumed, is *very* sensitive to the choice of the prior. Indeed, an alternative title for this paper could be "a sensitivity analysis of the choice of the prior in the problem of induction". To perform this analysis, the authors describe the consequences of alternative priors proposed in the literature and, building on alternative proposals which they actually discard for their unappealing properties, they suggest the use of the *portmanteau* prior arguing that this induces the posterior properties which software engineers might like.

I believe it is important to specify the role which the prior distribution is intended to play. If a subjective viewpoint is taken, then an appropriate (presumably large) family of priors with appropriate properties should be envisaged, and the scientist instructed to select that closest to his o her personal views. The family of portmanteau-like priors suggested by the authors may well be a reasonable choice for this purpose. This is of course a perfectly valid procedure, but the conclusions reached will have limited interest to other scientists.

As a baseline for the range of conclusions which may be reached from different priors, one needs an *objective* prior which, in a well defined mathematical sense, may be claimed maximize the missing information about the question of interest, here whether or not $R = N$. This is precisely the *definition* of the reference prior, and the prior in Bernardo (1985) is the reference prior for this problem. Thus, the question is *not* whether that prior is "optimistic" (a subjective evaluation) or otherwise but whether or not it provides a sensible baseline

for possible conclusions on $\Pr[R = N \mid r = n]$, under the *sole* assumption of the accepted hypergeometric model. If the scientist is prepared to spend the necessary effort to specify a prior which describes his or her opinions about the problem, he or she should go ahead, compute the corresponding posterior probability, and compare this with the *reference* value provided by by the reference posterior probability; any discrepancy is a consequence of his or her subjective prior information, and this is relevant knowledge, for both the scientist involved in specifying the subjective prior, and his or her colleagues. Reference priors are derived from a *systematic*, information-theory based procedure, which has proved to provide appropriate answers in all known cases: see Bernardo (1977), for a general discussion, and Bernardo (1994, Ch. 5) for technical details. Thus the Bernardo prior for this problem is *not* an *ad hoc* proposal, but an application of a successful general theory and should therefore be judged not only on the merits of the specific solution found (whose behaviour I actually like), but also on the strength borrowed from the general theory.

I would like to conclude with a comment on the "optimistic" nature of the reference prior, drawn from actual experience. In January 1983 I was invited to give some talks and do some consultancy on Bayesian statistics at the Charles Darwin Research Station, on the Galapagos Islands. About the first question posed to me was a particular case of the problem discussed in this paper: a scientist had observed that all the $n = 67$ galapagos seen in a visit to one of the smaller islands presented a modification on their shells, presumably a genetic evolution to facilitate access to the prevailing food in the island, and she naturally asked me what was the *probability* (her words) that *all* galapagos in the island did have such modification. This motivated my interest in the problem, which I analyzed there, and the scientist was very happy with my result, $(n + 1)/(n + 2) \approx 0.986$ (since the total number N of galapagos in the island was assumed to be much larger than n). I then asked her what would be her reaction if, in a future visit to the island, she founded one galapago without the modified shell; she said that she would assume that this was a genetic leftover from the original population, and that this subspecies would eventually die out in the island in competition with the better equipped, modified galapagos. Technically, she was claiming that the unmodified galapago would not really be a member of the population under study. Thus, the reference prior, optimistic or not, seems to be in agreement of actual expectations from scientists.

Additional References

Bernardo, J. M. (1997), Noninformative priors do not exist. J. Statis. Planning and Inference, **65**, 159–189 (with discussion).

Bernardo, J. M. and Smith, A. F. M. (1994), Bayesian Theory. Chichester: Wiley.

Comment

Philip J. Boland

National University of Ireland, Dublin, Ireland.

The question of infinite trustworthiness is of course an interesting one, with wide potential applications. We can all think of situations where one desires infinite trustworthiness, quasi certainty or near perfection from successful information at hand. In many situations, one desires such a degree of belief or confidence with a relatively small amount of sample information.

The problem is principally motivated through concerns about software testing and reliability - particularly when the software being tested is for example life critical (software used in air traffic control, space shuttle technology, or surgical instrumentation and diagnosis). In such situations, one must have near perfect reliability before release. Although software testing does provide a focus for this research - which is important, even for life critical software the model implied for testing is a severe oversimplification of what actually happens (see Boland et al., 2003). For example, to imply that all inputs are or can be considered of equal importance is rarely (if ever) the case. Software testing is to a large extent targeted (and usually random only in a restricted and limited sense), where for example some primary targets of the testing might be certain (perhaps critical) software specifications and requirements of the software. Furthermore, software faults are usually multivariate in nature, and in particular may affect different functional areas of the software. They may also be of different severities and have different priorities (if the software is being developed by a phased iterative procedure this is often the case - see Faundez Sekirkin, 2004). Even strategic and safety critical software is rarely fault free (see for example Schneidewind, 1997) - but

this may be acceptable if at the time of release the only remaining faults are noncritical.

The essence of the discussion centers around the question - what is $P(R = N)$ given that n tests give n successes? In particular, what kind of prior distributions for this quantity give reasonable conclusions? Surely the Laplace prior is seldom appropriate. For example, in a setting like that of life critical software, if you are so uncertain and uninformed about the software reliability, then the software itself can really be nowhere near release stage. To say that the result ($\lim_{N\to\infty} \frac{n+1}{N+1} = 0$) goes against the grain of most scientists is perhaps a bit extreme - for to hope that it might be 1 or close to it is expecting a lot from (under the assumptions) so little! Jeffrey's prior in the setting of life critical software is also a bit extreme (is it not slightly abnormal to put a large mass at both of the extremes $R = 0$ and $R = N$?) Bernardo's prior is getting closer to a reasonable prior.

Most of us are risk averse, so it is to be expected that we might cling to conservative or pessimistic type priors. The idea of several small surprise switches is somewhat appealing (like those in the pessimistic and regulated pessimistic priors), but one would naturally worry about a prior which so quickly (that is for small n) gives such perfection in trustworthiness. Furthermore one would like to see some intuition in the parameters λ and q, and why certain values lead to such dramatic results for small samples. The idea of scaling is interesting, and leads to a wide variety of possibly more intuitive priors. The portmanteau concept is also of value, although a serious challenge to constructing a good one must be in deciding on a good change point (in this treatment it is $R = \frac{9N}{10} - 1$).

In spite of these reservations, the topic is treated in both an interesting and intriguing manner, and the authors are to be complimented in this regard.

References

Boland, P. J., Faundez Sekirkin, S., and Singh, H. (2003), Theoretical and practical challenges in software reliability and testing. Mathematical and Statistical Methods in Reliability, Edited by Bo H. Lindqvist and Kjell Doksum, **7**, Series in Quality, Reliability and Engineering Sciences, World Scientific Publishing (Singapore), 505-520.

Schneidewind, N. F. (1997), Reliability modeling for safety-critical

software. IEEE Transactions on Reliability, **46**, 1, 88-98.

Faundez Sekirkin, S. (2004), Statistical methods in the software quality assurance process. PhD Thesis, National University of Ireland.

Comment

D. R. Cox

Nuffield College, Oxford, UK.

This very interesting paper addresses issues arising when major extrapolation is needed in passing from the data to the aspect of interest. Whatever approach is used sensitivity to assumptions is virtually inevitable and this is nicely llustrated in terms of sensitivity to choice of prior distribution. Flood forecasting and reservoir safety are other applications where somewhat similar difficulties arise.

A rather similar formal problem with a simple solution is as follows. Suppose that a catastrophe occurs in a Poisson process of unknown rate λ, it is required to predict whether the catastrophe will not occur within a specified time horizon which we take to be the time interval (0, 1), The required probability is $\pi = e^{-\lambda}$. It is observed that the catastrophe has not occurred in $(0, a)$,where $a < 1$. Then the required predictive (posterior) probability is

$$E(e^{-\lambda})/E(e^{-\lambda a}) = E(\pi)/E(\pi^a).$$

Here the expectation is over the prior distribution of λ or equivalently π. When a is small, strong sensitivity to the choice of prior is easily studied, for example by taking a gamma prior for λ,

The authors have entirely reasonably studied what forms of prior are needed to achieve answers with particular broad properties. This is quite counter to the usual Bayesian personalistic formulation in which the prior assesses *your* opinion separated from the data, whatever it may be, and the posterior then indicates what, subject to coherency, your opinion after obtaining the data should be. This approach inevitably raises the question: why should anyone except *you* be interested? One answer may sometimes be that the prior is based on evidence, in which case we should ask in principle at least

what that evidence is and how has it been analysed. A second answer might be that it is an expert opinion. While there are occasions when this has to be accepted, it is perilously close to settling issues by appeal to authority and this is in principle unacceptable in scientific contexts and in other contexts potentially very dangerous.

The paper throws light on important issues and I take the implication to be that there is really no substitute for the careful assembly of as much evidence as possible directly or indirectly bearing on the issue in question. Hydrologists have developed quite elaborate methods essentially for downweighting observations distant from the site under study and these methods could be regarded as a form of *empirical* Bayes analysis although typically not formulated that way.

Comment

Dave Higdon, Charles W. Nakhleh

Los Alamos National Laboratory.

We congratulate authors on an interesting paper. The problem of determining when trustworthiness is high with very limited testing is particularly important to applications we are involved with at Los Alamos National Laboratory. In addition to sofware testing, the question is clearly relevant to a wide variety of reliability, engineering and industrial applications.

The authors rightly point out that whether or not it is possible to attain infinite – or even just high – trustworthiness depends on the nature of the prior distribution $P(R = r)$ over the subset of the support given by $\lceil \rho N \rceil, \ldots, N$ (note $\rho < 1$ is taken to be $9/10$ in the portmanteau prior of Section 5.7). This is because if there is appreciable prior mass in the region $\rho N < r \leq N - 1$, then there is some chance a failure will occur for a small proportion of the possible input states. This is exactly the case when only a large amount of testing can dispel this possibility.

We find it useful to consider how the complexity of the computer code can affect the prior mass over this critical support region of the prior. To this end, Figure 1 shows a simplistic binary tree-based model of a computer code. We assume the tree has K levels, so that there are $N = 2^K$ possible input states. To process state j, the code must execute a sequence of subroutines denoted by the black nodes

in Figure 1. This sequence is uniquely determined for each state. We further specify that the chance a node/subroutine works correctly is p_k for each node in level k of the tree. Specifying prior probabilities $p_0, \ldots, p_K$ then induces the prior on the number of states the code correctly processes.

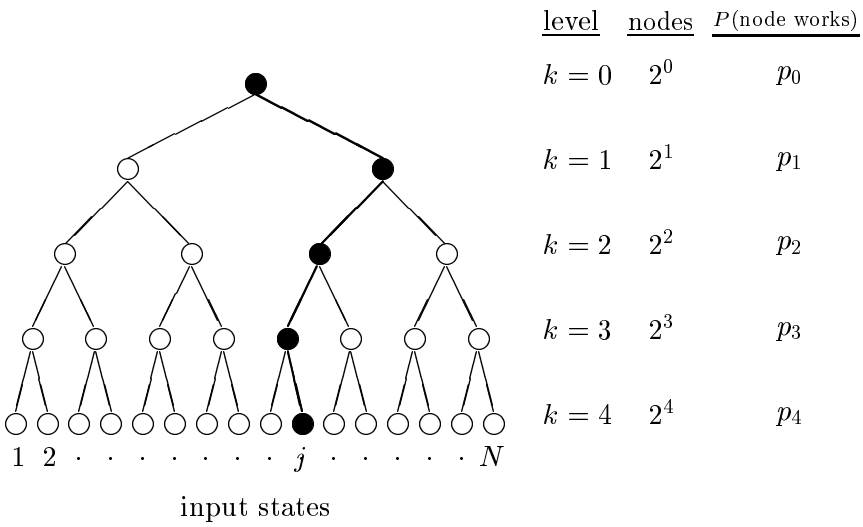

Figure 1: Binary tree-based model for a computer code.

For simplicity, we only consider priors for which a single $p_k < 1$, while the remaining probabilities are exactly one. For any such prior we fix the booster switch to have probability 0.25; hence $p_k = 0.25^{1/2^k}$. In this case, the number of nodes X in level k that work correctly follows a Binomial$(2^k, p_k)$ distribution. The total number of states that the code correctly processses is then $R = X\frac{2^K}{2^k}$ giving the induced prior shown in the top row of Figure 3 for $k = 2, 4$, and 6. The choice of k essentially gives the prior complexity of the code under investigation.

We can gain some insight into complexity by considering the entropy of R (MacKay, 2003) for a given level k:

$$H_k(R) = -\sum_{j=0}^{2^k} p(j) \log_2 p(j). \tag{11}$$

The entropy of a distribution over a finite set of points provides a logarithmic measure of the likely diversity of that set. By "likely diversity" we mean that if all the points are equally likely, then the

entropy of the set is the logarithm of the number of points; moreover, any point that has zero probability does not contribute to the entropy. (We take $0 \log 0 = 0$.) As the number of points 2^k in a uniform distribution increases, the entropy increases like $\log 2^k = k$. Note for R and X as defined above that $H_k(R) = H_k(X)$.

By fixing the booster switch at 0.25 for each level, we are effectively increasing the reliability of the nodes at a given level as k increases. This prevents the entropy of a level from increasing without bound as the number of levels increases. If the booster switch were not fixed at each level, and instead the reliability of each node were simply kept constant, then the entropy of the levels would increase linearly with k as the number of nodes in each level increased exponentially. A plot of H_k by k (Figure 2) illustrates these two cases. This figure illustrates the fact that a constant booster switch is essentially equivalent to a constant entropy. Clearly, as the entropy for the prior code model increases, it takes more code tests n to ensure adequate trustworthiness. The entropy measure also captures quantitatively how increased component reliability can offset increased complexity.

However, although $H_k(R) = H_k(X)$, the factor 2^{K-k} that appears in the definition of R means that the support of the distribution for R is pushed to higher and higher values of r, as illustrated in the upper row of Figure 3. This is due to the fact that the constant booster switch constrains the entropy to be nearly constant, and therefore the level k is also needed to specify the complexity of the code. If we had not imposed the constant booster switch constraint, then the entropy would be an equally good measure of complexity as k.

This mechanism for obtaining the prior $P(R = r)$ controls the amount of probability assigned to the range $\rho N < r \leq N$, as does the portmanteau prior. The resulting posterior probability the software never fails given $T = n$ is shown in the bottom row of Figure 3. Here high trustworthiness can be obtained with small n when the code has small k. Note that this probability does not depend on N in the limit – only k.

We suspect there are cases where it is more more natural to base the prior on the assumed complexity of the computer code. For example, we would consider the complexity of a routine to compute the cholesky decomposition of a matrix to be relatively small so that a small number of tests should lead to high – or even complete – trustworthiness. On the other hand, a very complex code – for example, a

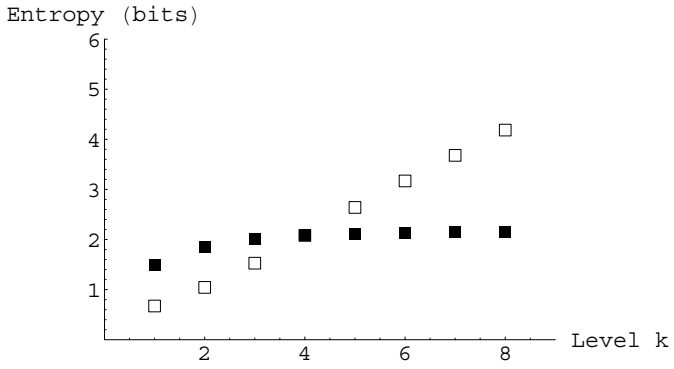

Figure 2: Entropy of the binary tree-based prior as a function of k. Filled squares have $p_k = 0.25^{1/2^k}$, open squares have $p_k = 0.25^{1/2^4}$.

telephone switching code (Eick *et al.*, 2001) – still has an appreciable chance of failing, even after extensive testing. This simplistic prior we propose here gives an example of how a complexity-based prior could be constructed. Again, we thank the authors for a thoughtful paper which lays out a framework for dealing with an important and difficult problem.

References

Eick, S. G., Graves, T. L., Karr, A. F., Marron, J. S. and Mockus, A. (2001), Does code decay? assessing the evidence from change management data. IEEE Transactions in Software Engineering, **27**, 1–12.

MacKay, D. J. C. (2003), Information Theory, Inference, and Learning Algorithms. Cambridge: Cambridge University Press.

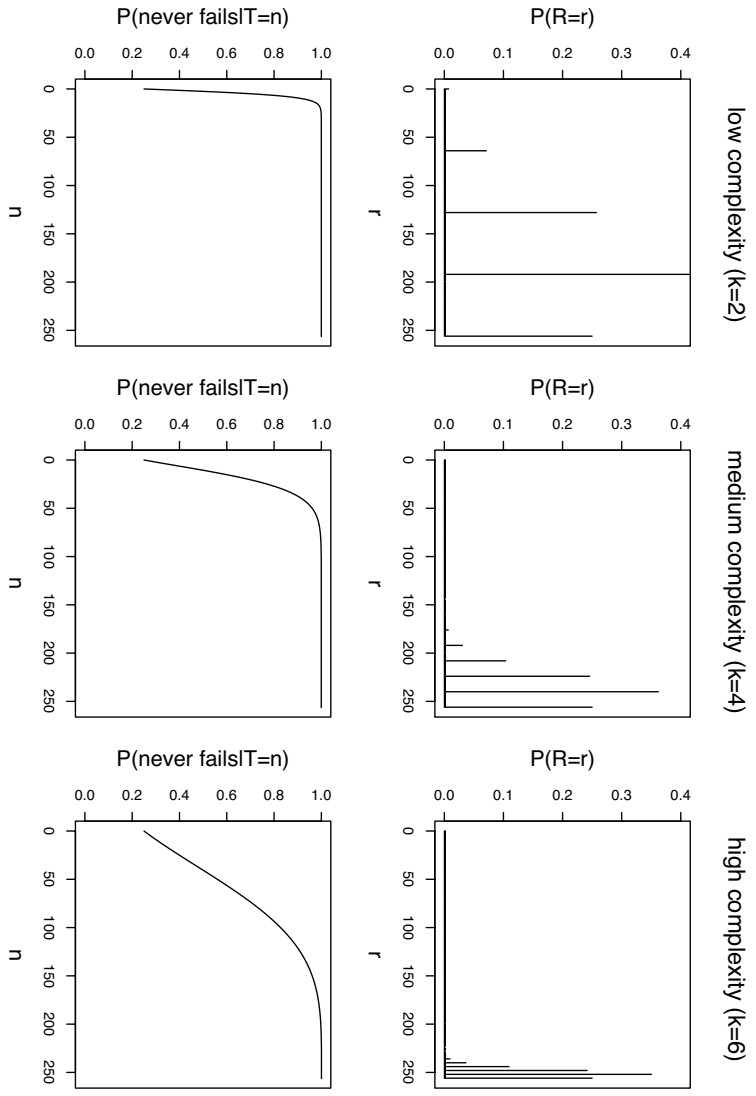

Figure 3: Tree based priors at various levels of complexity along with their corresponding posterior $P(\text{software never fails}|T = n)$. Low complexity priors quickly give high trustworthiness while the high complexity prior requires many trials before $P(\text{software never fails}|T = n)$ approaches 1. Each prior has a booster switch probability of .25.

Rejoinder

Nozer D. Singpurwalla, Philip Wilson

We thank the discussants for taking the time to comment on this paper and for the many insightful suggestions they make. We are grateful to Professor José Bernardo for orchestrating the discussion. Due to a severe time limitation imposed upon us by the Editor, Professor Ahmad Parsian, we are unable to fully capitalize upon some excellent points and ideas that the discussion has spawned. What we give below are a few cursory reactions.

Professor Bernardo suggests an alternative title to the paper, a title that better encapsulates the essence of the work. Whereas José's suggested title is more coherent than ours, we have succumbed to the Pygmallion effect and have retained ours; we think that ours whets a user's appetite. José is entirely correct in stating that in a purely subjectivistic context a large family of priors should be put on the table and the assessor instructed to select the prior that comes closest to the assessor's views. More important, the conclusions reached from such an exercise will be of limited interest to other scientists – a point also made by Sir David Cox. The point of our paper was not to advocate one particular prior over another. Rather it was to dissect each prior and look at its anatomy from the point of view of offering an explanation for the kind of result that each can produce. José's reference prior which inspired this work, and for which we applaud him, can indeed be used as a benchmark. What we may have ended up doing is to point out that José's information-theory based criteria may have, hidden behind it, elements of optimism. An issue which now needs to be explored is whether a general theory that is based on the premise of optimism is the one that is germane to all problems of scientific induction.

Professor Philip Boland questions the suitability of the set-up we consider for software testing. We share his concern because, what we have as a point of focus is "black-box" testing. This is a form of portmanteau testing. Such "black-box" testing is often done after all other forms of testing for specifics have been completed. But in a more truthful vein, our answer to Philip is that the software scenario was chosen in the spirit of a motivating illustration. We were attracted to the problem more from a foundational perspective, in the spirit of Pearson and José, than any particular application.

Philip also points out to some misprints in the paper; this of course is the fault of (the other) Philip who should never be trusted to write software code. And if he ever does write code, then let it be known to all that the most pessimistic prior be used!

We thank Sir David Cox for his kind words. There are two striking issues in David's discussion. The first is that the scope of the "fundamental problem of practical statistics" has been expanded from the discrete scenario of binomial trials considered before, to the continuous scenario of observations over time encountered in flood forecasting and reservoir safety. This expansion is insightful and is particularly appealing to us because of our interest in reliability and survival analysis wherein items are test observed for some time and then released for general use. David's second – more philosophical – point pertains to his question "why should anyone except you be interested?" A question like this could generate the view that the only place for subjective probability is personal (or group) decision making. David's answers to his question suggest that a decision maker, say $\mathcal{D}$, elicits its priors from someone else – say an expert $\mathcal{E}$ – and then either questions these priors or takes them as a word of authority before invoking Bayes' Law for fusing the prior with the data. By itself, such a process would not be coherent. Rather, what $\mathcal{D}$ needs to do is use Bayes' Law to fuse $\mathcal{E}$'s testimony with $\mathcal{D}$'s own prior using a likelihood whose nature encapsulates $\mathcal{D}$'s questioning of $\mathcal{E}$'s sources of evidence and also the extent to which $\mathcal{D}$ acknowledges $\mathcal{E}$'s authority. Once $\mathcal{D}$ does this, a prior that can be fused with the data evolves.

The discussion of Drs. Dave Higdon and Charles Nakhleh brings into the arena two features. The first is a binary tree model for conceptualizing a software's success/failure process. The second is constructing a prior based on the tree model. An ingenious aspect of Dave and Charlie's contribution is their construction of a binomial probability model with $p_k = 0.25^{1/2^k}$; this ensures that for all k, the size of (our) booster switch is always 0.25. The value of k encapsulates the complexity of the code. Consequently, codes having a low complexity provide a quick assessment of high trustworthiness. All this makes good sense. Where we encounter a difficulty with these discussants is their connecting of complexity with entropy. We are inclined to believe that entropy may not be the right metric of complexity. In our particular scenario the entropy would be the same whether the switch is a booster switch or a surprise switch. Yet the rate at which trustworthiness is achieved would be different under

the two scenarios. The difficulty with using the entropy measure alone, is that entropy is invariant with respect to the placement of probabilities. What is therefore needed is a measure of complexity that addresses both the nature of the probability masses and their location.

Once again, we thank all the discussants for their contribution to our learning and to Professor Ahmad Parsian for inviting us to write a paper for JIRSS which we wish much success.