Software Reliability Prediction: A Multi-Modelling Approach

Volume I

Sarah Brocklehurst

Thesis Submitted for the Degree of Doctor of Philosophy in Computer Science

City University

Centre for Software Reliability
City University
London EC1V OHB

February 1995

Contents

Volume I

Contents

Volume II

Tables and Illustrations

Tables

Illustrations

Acknowledgements

This work has been supported in part by the UK Science and Engineering Research Council and Alvey directorate under project SE-072 and in part by the CEC ESPRIT programme under projects PDCS1 (3092) and PDCS2 (6362). I would thank Professor B. Littlewood for his supervision, and other colleagues at the Centre for Software Reliability for their advice and help, Dr. P. Y. Chan and Professors A. Sofer and D. R. Miller for use of their software and P. Mellor and A. Tanner for providing raw failure data.

Abstract

Many software reliability models are now available to the user who wishes to assess and make predictions about the future reliability of his/her system by using its past failure behaviour. Experience of applying such models to failure data in the past has shown that, to date, *there is no one model that will give accurate predictions in all circumstances* (i.e., over different data sets). Recent work has thus concentrated on the development of techniques for the assessment of the accuracy of predictions made for the data of interest. One of these techniques also allows the user to improve initially inaccurate predictions via a process of *recalibration*. The demonstrated success of the recalibration technique suggests that it may be possible to apply fairly simple models, for example non-parametric models, and to achieve reliability predictions which are as accurate as those which could be obtained from more sophisticated models.

Here, it is recommended that a "multi-modelling" approach should be taken to the problem of software reliability prediction. That is, a number of parametric and non-parametric models and the recalibration technique should be applied to the failure data from the system of interest and by using the various analysis techniques accurate predictions may be selected for the future failure behaviour of this system, from amongst all the resulting "prediction systems". This thesis gives guidance on how such an approach should be taken and validates the approach by application of these methods to some real software failure data. In order to minimise the effort on the part of the user, the feasibility of automating such selection between prediction systems is also investigated.

The general conclusion of this work is that the "multi-modelling" approach suggested is effective, in terms of obtaining, fairly automatically, reliability predictions which can be trusted for each data source. We make recommendations on how to minimise effort on the part of the user of such techniques, by more intelligent choice of initial software reliability models, application of subsequent techniques for improvement in the accuracy of predictions and automatic selection from amongst the available prediction systems. There is evidence here that we could apply a small number of raw models (some parametric and some non-parametric), the recalibration technique, and finally the meta-predictor for automatically choosing the best predictions, and the resulting predictions will probably be about the same in accuracy as any single predictor that could have been used.

Key to Abbreviations

HPP:	Homogeneous Poisson Process
NHPP:	Non-Homogeneous Poisson Process
JM:	Jelinski Moranda model
GO:	Goel Okumoto model
MO:	Musa Okumoto model
DU:	Duane model
LM:	Littlewood model
LNHPP:	Littlewood *NHPP* model
LV:	Littlewood Verrall model
KL:	Keiller Littlewood model
CM:	Non-parametric model - completely monotone
OTY:	Non-parametric model - optimising trend statistic: *y*-plot
OTL:	Non-parametric model - optimising trend statistic: Laplace and extensions
S:	Spline-recalibrated prediction system
M:	Meta-predictor
cdf	cumulative distribution function
pdf	probability density function
m_j	mean time to next failure
w_j	median time to next failure
K distance:	Kolmogorov distance
K test:	Kolmogorov test
ML	maximum likelihood
PLR	prequential likelihood ratio

1 Introduction

The use of software and computer systems has increased substantially in recent years due to the relative ease with which such systems can be implemented and software is increasingly being used to replace systems previously implemented in hardware, and to develop entirely new systems. Unfortunately evaluation methods which can be used for assessing the future reliability of software systems have not been developed, or have not been taken up by industry, sufficiently quickly to keep up with the pace of the introduction of such systems.

The very nature of software makes the problems involved in assessing performance of systems containing software very different than in the case of more conventional systems implemented in hardware [Littlewood 1989; Musa 1975]. Faults which exist in software are design faults essentially due to human error and making predictions of the failure behaviour which will result from such faults is a non-trivial task. It is clearly not possible to use techniques traditionally used in hardware systems, where it is often possible to be certain that there will be few or no design faults due to the relative simplicity of the design of such systems, or to the fact that new systems often make substantial re-use of previously tried and tested designs, so that the failure behaviour is mainly due to wear-out rather than to design faults. The relative ease with which extra functionality can be added using software often results in a system where the design is substantially more complex than the system which it is intended to replace. The production of a system where part of it is implemented in software, therefore, invariably results in an entirely novel, and often complex, new design, where the failure behaviour can be expected to be largely due to design faults. Further, it is quite possible that a error made in the design of a system, which in terms of design is conceptually fairly trivial, or subtle, may actually result in very serious failure consequences.

The problem of predicting failure behaviour due to design faults is compounded by the relative ease with which changes can be made in the design of the system during its operational life. Such changes are often made either as an attempt to fix a design fault after having observed a failure arising from this fault, or for some other reason, like the desire to add some new functionality for system enhancement or changing the system in some way in response to a change in the needs in the environment in which it is operating. This results in an evaluation problem where we are trying to predict the future failure behaviour of a complex system due to design faults, the design of which is continually changing with the removal (and possibly introduction) of design faults, and which may also be operating in a changing environment.

Methods developed for software reliability prediction such as those to be investigated here are applicable to a wider range of systems than just those containing software, since the theory will relate to any system for which the design is sufficiently complex that failures due to design faults will make a sufficient contribution to the overall unreliability of the system and for which uncertainty about the future failure behaviour due to design faults will inevitably result [Perrow 1984]. Further there is no reason that software reliability modelling cannot be applied to other aspects of dependability (for a definition of dependability attributes see [Laprie 1992]) apart from reliability, for example safety. Recent work [Brocklehurst et al. 1993; Brocklehurst et al. 1994; Littlewood et al. 1994] has investigated the plausibility of applying similar techniques to predict the operational security of a system. All such cases where a system does not perform as it is required as a result of design faults are possible candidates for the application of software reliability modelling techniques.

There are a number of alternatives to software reliability modelling suggested for predicting software failure behaviour. Many of these, rather than being aimed at actual evaluation, consist of trying to assure that the software is developed sufficiently well that the resulting design will be perfect, or good enough, and there will be no, or tolerably few, failures due to design faults in operation.

Some of these methods involve an informal notion of "good design practice" where it is recommended that particular methods and tools, requirements specification and programming languages, and so on, are used in the development process of the software, or that particular software certification standards be used. However, although there is no doubt that good design practice is likely to help in the area of reliability achievement this does not help in the area of reliability evaluation.

A more formal notion which is aimed at guaranteeing that the system, or part of the system, is correct involves the use of "formal methods" [Fenton and Hill 1993; Gehani and McGettrick 1986; Shaw 1984]. Here mathematical proofs may be used to assure that the implemented software meets exactly some formal specification of the requirements. There are several problems with this approach [Fetzer 1988]. The first is that these methods may be carried out manually by experts and in such cases it is not possible to guarantee that mistakes have not been made in the proofs, particularly in the case of large complex systems. Also, in such a translation from one level of formality to another it seems likely that there will be a tendency for mistakes to occur in difficult parts of the problem for which the system is the proposed solution, and thus to coincide with design faults in the implemented code. Further, it is often too impractical or costly to use such methods manually on anything but small systems, or subsystems, where the failures due to design faults may, in any case, be expected to occur fairly infrequently. Automatic

tools are available for applying such formal methods, but again, this area is not sufficiently mature that it can be guaranteed that mistakes have not be made either in the tools themselves, or in the use of these tools. A further problem, which is unlikely to be solved by advances in research in formal methods, is that it is not possible to assure that a formal specification truly meets the informal requirements of the users. So, although the use of such formal methods is likely to increase the reliability of a system, it cannot be used to guarantee that a system is correct, particularly in the case of large complex systems, and we need to evaluate the reliability of each new system which is produced.

Similarly, other methods, such as the implementation of software fault tolerant schemes and redundancy in order to mask failures in individual software versions [Lee and Anderson 1990], do help in reliability achievement but it is not possible to guarantee perfect or high enough reliability of such systems in a particular case and methods of evaluation of the achieved reliability in such systems are still required.

In theory, it should be possible to measure the operational reliability of systems in particular application areas and compare this with features of their design processes, in order to derive relationships between the two, and thus use this information to predict the operational reliability, or at least get reasonable confidence that some required level of reliability has been achieved, for a new system based on information about its design process. However, a proper scientific approach to this problem is fraught with practical difficulties. There are many factors effecting the operational reliability of a system and getting accurate predictions would involve insuring that the new system was sufficiently "similar" to the previous systems such that all explanatory variables that significantly effect the reliability have been captured in the measured features of the design process.

Unfortunately, there is little hard empirical evidence to support the fairly modest claims that are often made, such as the use of particular tools, formal methods, software certification standards, and so on, in the design process, will bring benefits in terms of quality or cost-effectiveness, and even less on measuring the actual benefits achieved from the use of a particular method [Fenton et al. 1994]. This is hardly surprising since the nature of software development is such that it is very difficult and costly to make such empirical investigations on the development of real systems, or to contrive properly controlled experiments on development of artificial systems.

Much research is needed in this area before we can ever hope to assess the actual impact of features of the design process on the achieved operational reliability, and to classify "similarity" of systems such that we could hope to infer the operational reliability of a new system in a particular class, from previous systems in this class. Further, since each new system is clearly unique, it is quite possible that there will not be enough

previous "similar" systems in existence so that such an inference could be made about the system of interest.

There has been some research (see, for example, [Khoshgoftaar and Munson 1990; Khoshgoftaar et al. 1992]) which addresses the problem of relating *static product measures*, such as program size, or complexity, to software "quality" on classes of products, with the aim of predicting "quality" of a new product via its static product measures. The motivation for taking such an approach arises from obvious truths such as that a system which is more complex is likely to have more design faults. However, in these investigations the aspect of quality which is used is the number of defects discovered in test or in operation, and so, although these methods are useful for identifying potentially problematic modules on which developers should concentrate their testing effort, for example, they do not help with evaluating the achieved operational reliability. What is really required are empirical investigations of the relationship between these static product measures and the actual operational reliability. However, such an approach to reliability evaluation would also be subject to the practical difficulties discussed above relating to being able to classify systems as sufficiently similar such that these static product measures will capture all significant differences in the operational reliability within a particular class.

We are thus not currently in the position to be able to guarantee that a system of reasonable size is completely reliable, nor to use information about the design process, or static measures of the product, in order to predict the operational reliability of a system.

Software reliability modelling, however, is sufficiently mature that it can be used during testing, or operation, of a system, for evaluating the reliability of a system due to design faults. This is done by using past failure behaviour of a system in order to make predictions about its future failure behaviour. It should be noted however, that there are various limitations to when such techniques can be used and restrictions on how they should be applied.

The work presented here is specifically for evaluating the reliability in situations where the system is either in operation, or it is being tested in such a way that the real operational profile (i.e., the environment in which the system is going to operate) is being adequately simulated. There are extensions to this [Cheung 1980; Littlewood 1979b] where, for example, a structural reliability model can be used together with the techniques presented here, and knowledge of the use in different environments with respect to the components together with reliability of the components may be used to evaluate the reliability in each environment. But these techniques still require, of course, that the operational profile of each environment be known.

Setting up a situation where such techniques may be used in test is something which system developers are often unwilling to do since it is frequently difficult or costly to simulate the real operational profile. However, since the failure behaviour of any system depends on the environment(s) in which it will operate with respect not only to the kinds of demands placed on the system but also with respect to the distribution of these demands, it is not feasible to believe that reliability evaluation can ever be achieved without using this knowledge of the operational environment(s). It also seems that there is often a rather unreasonable expectation that these very complex systems can be built adequately (i.e., reliably enough and with assurance of that reliability) at a cost which may be disproportionately low compared with the benefits that the system is actually intended to provide.

A further requirement for using these techniques is that a time metric which represents the amount of use, or *stress*, to which the system is being subjected [Mellor 1986; Musa et al. 1987] (e.g. C.P.U. execution time) needs to be found. These issues relating to the operational profile and to choice of an appropriate time metric are discussed further in Chapter 2.

One major limitation to the area of software reliability modelling is that, in order to get accurate predictions of the future failure behaviour of a system it is required that enough past failure data is seen in test or in operation. These means that these techniques cannot be used in the case of systems which are intended to have ultra-high dependability [Littlewood and Strigini 1992; Littlewood and Strigini 1993] (except of course, to reject a system on the basis that its reliability is too low), unless testing can be accelerated in some way. However, they are good techniques for use in the case of systems where the reliability requirements are relatively modest and thus enough failures could be expected to be observed.

There are several reasons why software failure prediction methods have not been readily taken up in industry even in cases where the reliability requirements are relatively modest. One is due to a cultural response to the nature of software faults and failures. Software failures are *deterministic* in nature; given that a system remains unchanged a failure will always result given the same triggering conditions of a design fault, or input, to the system. Due to this there is a tendency to believe that the failure behaviour is similarly deterministic and so a probabilistic approach to evaluation is inappropriate. This is, of course, a fallacy. A probabilistic approach is clearly appropriate for any system where we cannot be certain about the future failure behaviour arising from possible design faults due to the impossibility of exhaustively trying every triggering event which can occur in the expected life-time of the system, the uncertainty about what design faults which have not already been revealed exist in a complex system at any one time and the

uncertainty about what inputs will next arise, even if we can be certain about bounds within which these inputs lie.

Another reason for the poor take-up of such methods is that software engineering is a relatively new area, and that, again due to the ease of implementation using software, there is a tendency to believe that the effort required to develop such systems is disproportionately small compared to what might reasonably be expected for each novel complex new design. Software developers are often reluctant to put in the effort which is required to randomly test software (and investigate the operational profile, as discussed above) and/or to collect the required failure data in test or in operation. However, there are numerous instances in industry (see, for example, [Neumann; Perrow 1984]) where we see failures arising from design faults in operation with embarrassing, and sometimes catastrophic consequences, or where managers grossly underestimate the time and effort required to produce a system which is sufficiently dependable for the application for which it is intended.

Another factor which may have discouraged the use of such techniques is the overselling of particular software reliability models in the past. Users of a particular model may find that it is not robust (does not give accurate reliability predictions) for all sets of failure data, but only on some and on this basis be discouraged from using these techniques further. However, recent work [Abdel-Ghaly et al. 1986; Brocklehurst and Littlewood 1992], and the work presented here, supports the idea that individual models should not be expected to be robust and provides methods for overcoming these problems. Finally, although there have been techniques available for software reliability prediction for some considerable period of time, a certain amount of expertise is involved in using such techniques. The approach suggested here, therefore, is fairly pragmatic, with an emphasis on developing techniques which can be applied fairly automatically minimising the effort and expertise needed by the user of such techniques, with reasonable assurance that the resulting reliability predictions will be accurate for the system under investigation.

There are now many software reliability models available for the evaluation and prediction of the failure behaviour of a program (or system) undergoing debugging in test or in operation. A good theoretical description of most of these models is given in [Xie 1991]. The main purpose of software reliability modelling is to make *predictions* of the *future* reliability of the system using past failure data. For example they can be used during testing to make predictions of the current reliability and hence to decide whether a system is reliable enough to be put into operation, or to estimate how much testing will be required before a pre-specified target reliability for the system will be reached, or to estimate the future reliability of the system in operation and hence likely maintenance

costs. Reliability models can also be used to examine failure behaviour in retrospect (i.e., use failure data to make estimations about the past) in order to make observations about the development and testing process, but most questions of interest are really about prediction, and so emphasis here will be placed on the ability of the models to give accurate reliability predictions.

Investigations of the ability of all these various software reliability models to give accurate reliability predictions on real data (see, for example [Brocklehurst et al. 1991; Brocklehurst and Littlewood 1992; Chan 1986]) have shown that some are universally bad, while the performance of others varies from data set to data set but *it is not possible to select a globally good model* which will perform well over even a particular class of systems. This has led to the development of techniques [Abdel-Ghaly et al. 1986; Brocklehurst and Littlewood 1992] which assess the past predictive performance and compare the relative merits of these models in a particular context (i.e., when applied to the data set under investigation). These techniques allow us to apply a number of models to the data of interest and select a "best" model to use for future predictions on this data based on their past predictive performance. The approach taken here, similarly, does not concentrate on the development of intricate new software reliability models, but instead on techniques for making decisions about which predictions to use and on making improvements in the initial predictions.

One of the techniques used allows us to assess one aspect of the error in the model predictions, a kind of "bias", and to *recalibrate* the model predictions with respect to this error. The resulting recalibrated prediction system is still truly predictive and we can therefore use the analysis techniques described above to assess the improvement gained via recalibration every time it is applied. It has proved to be beneficial (see [Brocklehurst et al. 1990]) particularly when all *raw* (i.e., not recalibrated) models applied are in error, and the computational effort required for this technique is negligible compared with the effort required to obtain many of the initial raw model prediction systems. It is recommended, therefore, that it be applied as a matter of course to all the raw model prediction systems.

With the availability of robust techniques, such as recalibration, to improve on initially inaccurate raw reliability predictions, the possibility that the resulting predictive accuracy obtained by applying fairly simple-minded reliability models together with these methods for improvement may be as good as that obtained from more sophisticated models is worth investigating. If this turned out to be the case, then the effort involved in applying more sophisticated models (both with respect to computational effort or, more importantly, human expertise) could be substantially reduced. To this aim the performance of some simple non-parametric models are investigated here.

The general approach suggested is that, for each new failure data set, a number of models (preferably as many as is practical) should be applied and then the recalibration technique should be applied to each of these raw models. The analysis techniques may then be used to compare the predictive accuracy of all the resulting prediction systems. If the total number of prediction systems we have to compare is large, it may be desirable, particularly for the naive user, to automate this process of selection via one of the analysis techniques. A simple "meta-predictor" for automatic selection of predictions from amongst the available predictors will be investigated here.

One of the major barriers facing research in the area of software reliability modelling [Mellor 1986] is the lack of the data required in order to make predictions about the reliability of a program or system. Collecting the appropriate data can be a costly process and, as previously discussed, testing has to be conducted under a suitable operational profile (i.e., a user profile), or data must be collected in operation, in order to get accurate reliability predictions. With a view to aiding this area objectives of the Alvey SRM project [Potter 1988; Potter 1989; Simmonds 1988a; Simmonds 1988b] included the setting up of a database in order to collect some reliability data. Here some of the data collected in this project (inter-failure time data from in-field use over *4* years of a single user work station) together with other available data [Gaudoin 1988] consisting of continuous inter-failure times of systems undergoing debugging, is analysed and used to illustrate and validate the approach suggested here.

In the following text we shall briefly describe the parametric models which we are going to apply to the data sets in Chapter 2. In Chapter 3 we shall describe the analysis techniques used in order to compare their predictive performance and in Chapter 4 the recalibration procedure will be described and investigated. The non-parametric models to be applied will be described in Chapter 5, and in Chapter 6 a simple technique which utilises one of the criteria for analysis of predictive accuracy in order to allow the user to automatically choose between the resulting prediction systems for the data set of interest will be described. We shall then discuss details of the failure data to which we are going to apply these techniques in Chapter 7 and follow with a detailed analysis in Chapter 8 of the performance of the various prediction systems and techniques on the data sets. The raw reliability data, together with all associated tables and plots for analysis, are contained in Appendix B in a separate volume, Volume II. A summary of the general conclusions together with suggestions for future work will be given in Chapter 9.

The general conclusion of this work is that the "multi-modelling" approach suggested is a good one. We make recommendations on how to minimise effort on the part of the user of such techniques, by more intelligent choice of initial software reliability models, application of subsequent techniques for improvement in the accuracy of

predictions and automatic selection from amongst the available prediction systems. There is evidence here that we could apply a small number of raw models (some parametric and some non-parametric), the recalibration technique, and finally the meta-predictor for automatically choosing the best predictions, and the resulting predictions will probably be about the same in accuracy as any single predictor that could have been used.

2 Raw Reliability Models

The overall objective of reliability modelling is to use past failure behaviour of a system in order to make predictions about its future failure behaviour. The reliability predictions are expressed as probabilities; it is reasonable to expect the failure behaviour of large software systems to be non-deterministic due to uncertainty about which inputs will next arise in a particular operating environment and to uncertainty about which of these inputs will result in failure. More detailed arguments as to why it is reasonable to expect random failure behaviour from such systems may be found in [Jelinski and Moranda 1972], [Laprie 1984] and [Xie 1991].

All the reliability models considered here are *black-box* reliability models; they are applied to failure data of the complete system/program with no attention given to the internal structure of the system. There is no possibility, unless applied to sub-components of the system, for example, together with an appropriate white-box model (see, for example, [Littlewood 1979b] and [Cheung 1980]), for extrapolating from failure data in one environment, to reliability predictions in another, different, environment. This means that if they are applied to failure data observed when the system is in a particular environment the resulting reliability predictions may only be expected to be accurate for continued use of the system in the *same* environment.

Formally, such an environment (or *operational profile*) may be characterised by random selection of inputs from the *input space* (which may include the system states and other environmental factors which may affect the failure behaviour as well as the totality of physical inputs to the system) subject to a probability distribution which represents usage of the system under operational conditions [Musa et al. 1987]. Such failure data may thus arise from either random testing under the appropriate operational profile (i.e., that profile which truly represents expected use of the system for which the reliability predictions are required) or from real operation of the system. Here the issue of how to characterise the operational profile in the case of random testing is not addressed since it is largely application dependant, although it is recognised that for many systems this problem is likely to be non-trivial. The failure data from the single user work station presented later in Chapter 7, arises from real operational use, and so the problem of characterising the operational profile is not an issue.

A further point to consider in the application of reliability models is that of what metric on which to base the failure data. In order to get accurate reliability predictions by application of the reliability models presented here it is necessary to have a metric which represents the amount of use, or *stress*, to which the system is being subjected [Mellor

1986; Musa et al. 1987]. Real (or calendar) time is rarely appropriate since this does not usually represent the times at which the system is actually operating and thus includes times at which failures cannot occur. C.P.U. execution time is clearly a better measure of the amount of use [Musa 1975] although it is often difficult, in practice, to measure actual execution time. In the case of the work station failure data it was decided that "hands-on" time of the user at the work station was a suitable metric. This is discussed in more detail in Chapter 7.

Most reliability models assume that the system being subjected to test, or in operation, is undergoing perfect debugging (i.e., each fault is removed immediately after the first failure associated with this fault is observed). Thus, in the case of perfect debugging repeated failures from the same fault will not occur and the data collected will consist of times between successive failures of the system, for example. For the single user work station data presented in Chapter 7, corrections were (generally) not carried out and the data collected consisted of all failures observed; only the first failure arising from each fault was later extracted in order to obtain the required data. This effectively simulates what would happen under perfect debugging and the resulting reliability predictions are thus predictions about future first failure occurrences of new faults (and not repeated failures from previously identified faults).

Once having decided on an appropriate time metric there is still the form in which the failure data is collected to be considered. Commonly *discrete failure data* may be available, that is, the number of failures within successive (possibly non-equal) intervals of time are recorded. In fact, in real applications, this is often the easiest form in which to collect the failure data, since failure occurrences may be reported during test or from operation, on a daily, or weekly basis, for example, without the actual times of these failures being recorded. Models for discrete data are considered in [Abdel-Ghaly 1986], [Wright, 1989 #540] and [Knafl 1992]. In this work we limit our analysis to more stringent continuous inter-failure time data, that is, data for which all of the successive times at which each failure occurs are known.

The data we are considering are thus times, t_1, t_2, t_3, ... (with an appropriately chosen time-metric, as discussed above) between successive failures resulting from first occurrence of unique faults of a system.

For simplicity we shall be limiting our analysis to *one-step-ahead* predictions although many of the models have the ability to predict further into the future. Using the previous inter-failure times, t_1, t_2, ..., t_{j-1}, the raw reliability models provide a probability prediction of the current (and as yet unobserved) inter-failure time, T_j, in the form of a *predictive cumulative distribution function* (*cdf*),

$$\hat{F}_j(t) = \hat{Pr}(T_j \leq t) \qquad \text{.....(2.1)}$$

From *(2.1)* we also have a *predictive probability density function* (*pdf*) for T_j,

$$\hat{f}_j(t) = \hat{F}'_j(t) \qquad \text{.....(2.2)}$$

These can be thought of as estimates of the true underlying *cdf* and *pdf*, $F_j(t)$ and $f_j(t)$, respectively, of the current inter-failure time, T_j.

From these estimates a number of different types of reliability measures may be obtained, and the choice as to which types of predictions are of interest will depend on the particular context in which they are applied, or on the individual preferences of the user of such models. Examples of commonly used predictions are the *reliability function*, which is the probability that the system will operate without failure beyond a specified time, t,

$$\hat{R}_j(t) = 1 - \hat{F}_j(t) \qquad \text{.....(2.3)}$$

and the *hazard function* of T_j,

$$\hat{h}_j(t) = \frac{\hat{f}_j(t)}{1 - \hat{F}_j(t)} \qquad \text{.....(2.4)}$$

and summary statistics such as the *mean time to next failure*,

$$\hat{m}_j = \int_{t \geq 0} t \, \hat{f}_j(t) \, dt \qquad \text{.....(2.5)}$$

In the subsequent analysis in Chapter 8 the *median time to next failure* ,

$$\hat{w}_j = \hat{F}_j^{-1}(0.5) \qquad \text{.....(2.6)}$$

is examined. For illustrative purposes we limit ourselves to consideration of the median time to next failure since this reliability measure can be easily obtained for all the raw reliability models and other techniques for obtaining predictions which will be investigated here. For some models the mean time to next failure does not exist [Littlewood 1979a].

As the data evolves we can repeatedly make one-step-ahead predictions from each model. Thus for a data set which consists of q inter-failure times altogether each model may be applied to the data $t_1, ..., t_{j-1}$, to obtain our one-step-ahead prediction for T_j, for $j = s, ..., q$, say, where s is a number suitably large for making the first prediction. We then have what we shall refer to as our "raw prediction system" for each of the models,

$$\{\hat{F}_j(t), \hat{f}_j(t); j = s, ..., q\} \qquad \text{.....(2.7)}$$

As mentioned above, most conventional reliability models assume that once a fault manifests itself as a failure it is fixed instantaneously and operation (or testing) continues: the expected failure behaviour is therefore *reliability growth*, at least in the long term, although there may be short term reversals. This is not an implausible assumption if the data is collected sensibly. (Note that if we did not expect, on average, reliability growth then we would not be debugging our system). All the raw models considered here can model stable reliability or reliability growth (i.e., constant failure rate or monotonically decreasing failure rate) while some models can also model reliability decay (i.e., monotonically increasing failure rate). None of the models applied here, however, are able to model trend *changes* (for example the transition from stable reliability to reliability growth), since they all assume a smooth trend. However, even though each fitting of the model will be unable to represent a change in the trend of the data, a sequence of *predictions* from the model may respond to such a change. That is, when dynamically applying a model over a succession of one-step-ahead predictions upon a data set containing changes in trend, it may be that the series of predictions *themselves* (i.e., the raw prediction system) do not exhibit smooth trend.

In some texts [Kanoun and Sabourin 1987; Kanoun et al. 1988; Martini et al. 1990] the data is examined for such trend changes and the models are then applied accordingly over different intervals of data separated by such changes. For example, if the data exhibits stable reliability followed by reliability growth (as is commonly the case for such failure data) then the models may be applied over the region of growth by omitting the early data. Here this approach is not taken, partly because the emphasis is on prediction and it is difficult to identify such "change-points" in the data until some time after they have occurred and also because there are no formal tests to identify anything but very simple changes in non-stationary data. So here the models are applied "blindly" as would a naive user over the whole range of the data, and application of the models begins at a stage at which it is decided, fairly arbitrarily, that there are enough initial data points for making the first prediction. There is evidence [Brocklehurst et al. 1991] which suggests that any inaccuracy in the predictions resulting from such an approach may be adjusted for by the process of recalibration described later in Chapter 4.

As we shall see later, in Chapter 7, there is also a tendency for *outliers* to occur in the inter-failure time data. These are points which are unreasonably large or small when compared to immediately preceding data points. Such data points may be omitted before application of reliability models (see, for example, [Kanoun 1989] and [Kanoun et al. 1988]). This approach will not be adopted here, partly because it is difficult to tell whether these points correspond to genuine outliers or whether they are actually the beginning of a change in the trend in the data, until much later when more data has

accumulated, and also because there are currently no formal tests for detecting the presence of such outliers in non-stationary data.

The following section outlines the conventional *parametric* models to be applied to the failure data and some details associated with them. Additional *non-parametric* models are described later in Chapter 5.

2.1 Parametric Models

The *parametric models* assume a form for the sequence of *pdfs* (*cdfs*) which depends on some unknown parameter(s). Estimates for these parameters are made at each stage, j, by using the previous failure data, $t_1, t_2, ..., t_{j-1}$, and the method of maximum likelihood (*ML*). These parameters are then substituted into the *pdf* (*cdf*) in order to make predictions about T_j. The resulting predictive performance of these models will depend not only on their precise mathematical structure, but on the *ML* inference technique and the substitution rule for prediction. It should be noted that these two approaches to statistical inference and prediction are chosen here for convenience: other techniques, such as Bayesian inference, tend to be computationally intensive.

The parametric models which are applied here are the Jelinski-Moranda *(JM)* [Jelinski and Moranda 1972], Goel Okumoto *(GO)* [Goel and Okumoto 1979], Musa Okumoto *(MO)* [Musa and Okumoto 1984], Duane *(D)* [Crow 1977; Duane 1964], Littlewood *(LM)* [Littlewood 1981], Littlewood non-homogeneous Poisson process *(LNHPP)* [Miller 1986], Littlewood Verrall *(LV)* [Littlewood and Verrall 1973] and Keiller Littlewood *(KL)* [Keiller et al. 1983a] models. Since most of these models are well known the details are omitted.

As mentioned previously all of the models listed above assume a smooth trend in the data and can model stable reliability and reliability growth. The *DU*, *LV* and *KL* models can also model reliability decay whereas the other *5* models are restricted to growth or stable reliability.

The models described here vary considerably in the effort required to achieve the *ML* solutions at each stage. The *ML* solution for the *DU* model is analytical and so very easy to achieve, while the *JM*, *GO* and *MO* models require a one-dimensional optimisation for their *ML* solutions. The remaining *4* models require two-dimensional optimisations and are thus much more computationally intensive. Bearing in mind that we dynamically repeat our *ML* estimation for these models in order to achieve successive one-step-ahead predictions as the data evolves, quite a lot of effort is involved in application of these latter, more sophisticated models. The software used to apply all these models was the Software Reliability Modelling Package (*SRMP*) [RSCL 1988]

running on a Sun 3-80. A certain amount of knowledge is required in order to apply these models to each failure data set in deciding what bounds to use for the optimisation routines and other control parameters required. Details of how these control parameters can be chosen, and of the actual search algorithms used for the *ML* solutions can be found in [Chan 1986] and [RSCL 1988].

Many of the parametric models described in this chapter have the property that they tend towards simpler models as their parameters (or functions of their parameters) tend to extreme values (see [Littlewood and Verrall 1981], [Chan 1986] and [Miller 1986]). They will do this as a result of the behaviour of the data over which the models are being applied. Typically (see [Chan 1986]) the models may switch from one limiting case to another as they are applied over increasingly large intervals of data. For example, if the data is exhibiting no growth (the definition of no growth depends on the model we are concerned with), the *JM, GO, MO, LM* and *LNHPP* models tend to the stable reliability homogeneous Poisson process (*HPP*). There are a variety of ways, listed below[1], in which these models can tend to simpler case solutions.

$$\lim_{N\to\infty\ \phi\to 0} JM(N;\phi) = HPP\left(\frac{j\text{-}1}{\sum_{k=1}^{j-1} t_k}\right)$$

$$\lim_{\mu\to\infty\ \phi\to 0} GO(\mu;\phi) = HPP\left(\frac{j\text{-}1}{\sum_{k=1}^{j-1} t_k}\right)$$

$$\lim_{\xi\to\infty\ \beta\to\infty} MO(\xi;\beta) = HPP\left(\frac{\xi}{\beta}\right)$$

$$\lim_{N\to\infty\ \alpha\to 0} LM(N;\alpha;\beta) = MO(N\alpha;\beta)$$

$$\lim_{\alpha\to\infty\ \beta\to\infty} LM(N;\alpha;\beta) = JM\left(N;\frac{\alpha}{\beta}\right)$$

$$\lim_{N\to\infty\ \beta\to\infty} LM(N;\alpha;\beta) = HPP\left(\frac{N\alpha}{\beta}\right)$$

$$\lim_{\mu\to\infty\ \alpha\to 0} LNHPP(\mu;\alpha;\beta) = MO(\mu\alpha;\beta)$$

$$\lim_{\alpha\to\infty\ \beta\to\infty} LNHPP(\mu;\alpha;\beta) = GO\left(\mu;\frac{\alpha}{\beta}\right)$$

---

1 $HPP(\lambda)$ is used to denote a homogeneous Poisson process with rate λ.

$$\lim_{\mu\to\infty\ \beta\to\infty} LNHPP(\mu;\alpha;\beta) = HPP\left(\frac{\mu\alpha}{\beta}\right)$$

Limiting cases for the *DU*, *LV* and *KL* models are not known.

For the *JM* and *GO* models the conditions on the data, $t_1,\ldots,t_{j-1}$, for the limiting case of an *HPP*, are known [Chan 1986] and thus can be tested before the model optimisation is applied. If these conditions are satisfied no optimisation is performed and the solution is simply exponential with the mean equal to the average of the data. For the remaining models similar conditions (for example conditions for which the *LM* model will tend to the simpler *JM* model) are unknown, although it can be observed in retrospect that this has occurred since the optimum solution will terminate with applicable parameters on the bounds. Thus, in these circumstances, where there are intervals of data for which there is nothing to be gained by fitting a sophisticated model, no saving in computational effort may be made.

In Chapter 8 we shall examine the accuracy of the various predictions resulting from application of these 8 reliability models to the data presented in Chapter 7.

3 Analysis of Predictive Quality Techniques

In this chapter three methods for assessing and comparing the performance of the various predictions are briefly described. These techniques for analysing predictive quality depend on a comparison between the estimated *cdf* or *pdf* (see *(2.1)* and *(2.2)*) at stage j, with the (later observed) realisation, t_j, of the next inter-failure time, T_j. This comparison is made over a series of one-step-ahead predictions, i.e., the prediction system as defined in *(2.7)*.

3.1 The u-plot

Our first technique, the *u-plot,* aims to detect consistent differences between predicted and observed failure behaviour in a single prediction system. It involves substituting the (later observed) t_j into the (earlier computed) predictive *cdf* over the range of one-step-ahead predictions.

$$u_j = \hat{F}_j(t_j) \qquad j = s, \ldots, q \qquad \ldots\ldots (3.1.1)$$

If each random variable, T_j, truly had *cdf* $\hat{F}_j(t)$ (i.e., $F_j(t) \equiv \hat{F}_j(t)$ for all $j = s, \ldots, q$) then the random variables $U_j = \hat{F}_j(T_j)$ will be uniformly distributed on *(0,1)* (i.e., $U_j \sim U(0,1)$); the sequence $\{u_j; j = s, \ldots, q\}$ will thus be a random sample from a uniform distribution.

The *u*-plot is the sample *cdf* of the *u's* defined in *(3.1.1)*. This is constructed as follows. First the *u*s in *(3.1.1)* are re-ordered in ascending order of magnitude to obtain the ordered *u*s: $u_{(s)}, u_{(s+1)}, \ldots, u_{(q)}$. A step-function is then drawn, with step-size $\frac{1}{q\text{-}s+2}$ (the number of *u*s plus one) as shown in figure 3.1-1. If the prediction sequence is accurate then the *u*-plot should lie close to the *45°* line, which is the *U(0,1) cdf.*

Significant departures of the *u*-plot from the *45°* line indicate that the prediction system is inaccurate in some way. If the inaccuracies in the predictions are *stationary* (i.e., consistent) then the *u*-plot can tell us something about the *nature* of the prediction errors. For example, figure 3.1-2 shows a *u*-plot resulting when the predictions are consistently optimistic. It has been shown in the past [Keiller et al. 1983b; Abdel-Ghaly et al. 1986] that application of the *JM* model to real data sets frequently results in optimistic predictions. If a model is consistently optimistic then this means the model is making predictions that are indicating that the system is more reliable than it in fact is. Thus, an optimistic predictor would result in too many small *u* values, since the actual observations would tend to lie to the left of the predicted *pdf* and the resulting *u*-plot

would tend to lie above the *45°* line. Conversely, if the predictions are consistently pessimistic then the plot would be expected to be everywhere below the *45°* line.

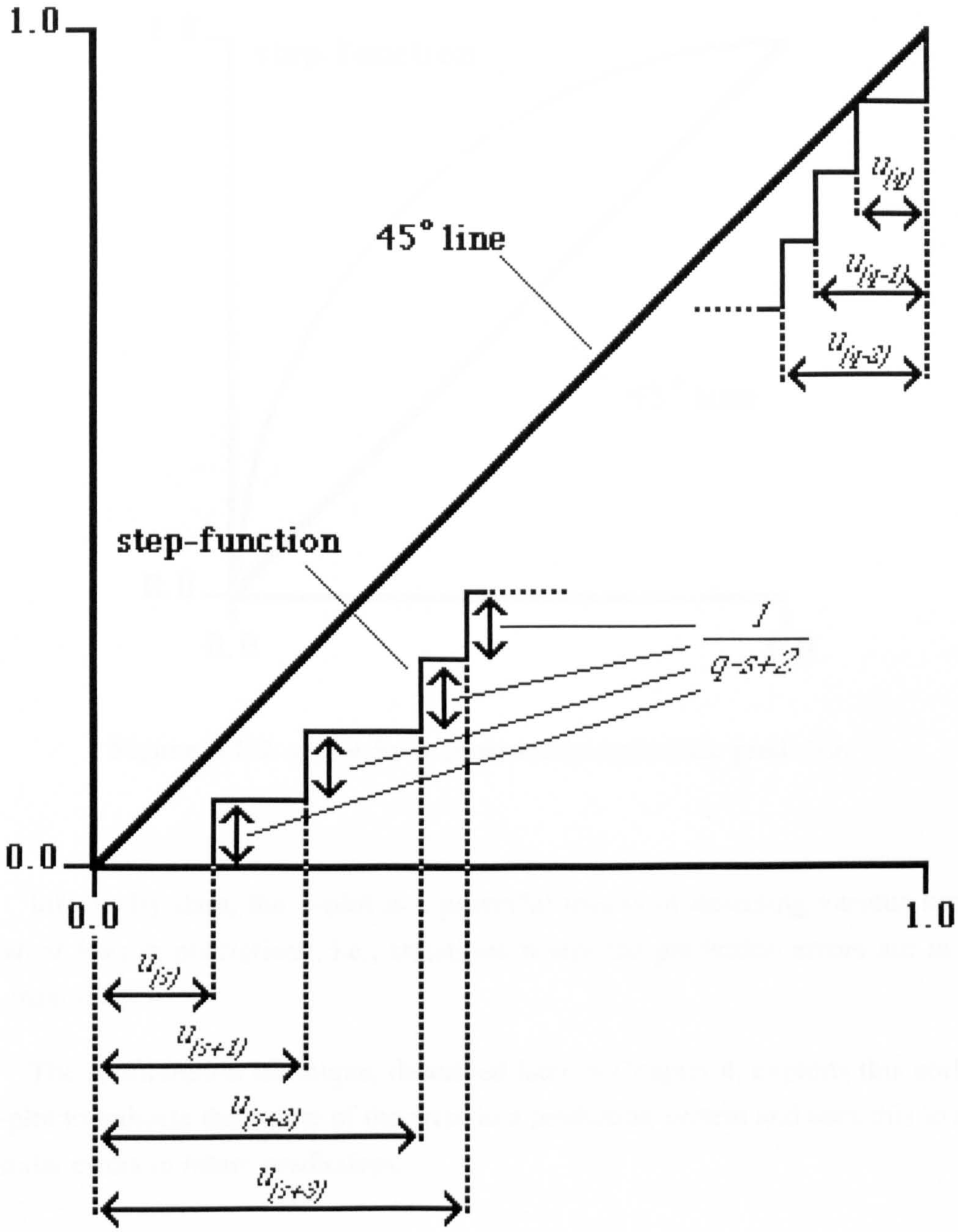


Figure 3.1-1. Constructing the *u*-plot.

Theoretically the *u*-plot may detect more complicated departures of the predictions from the truth than simple optimism and pessimism. For example, if a predictor is optimistic for small times and pessimistic for large times, an *S*-shaped *u*-plot would result

where the plot crosses the *45°* line. In practice though, such plots are frequently the result of non-stationary prediction errors as opposed to stationary errors.

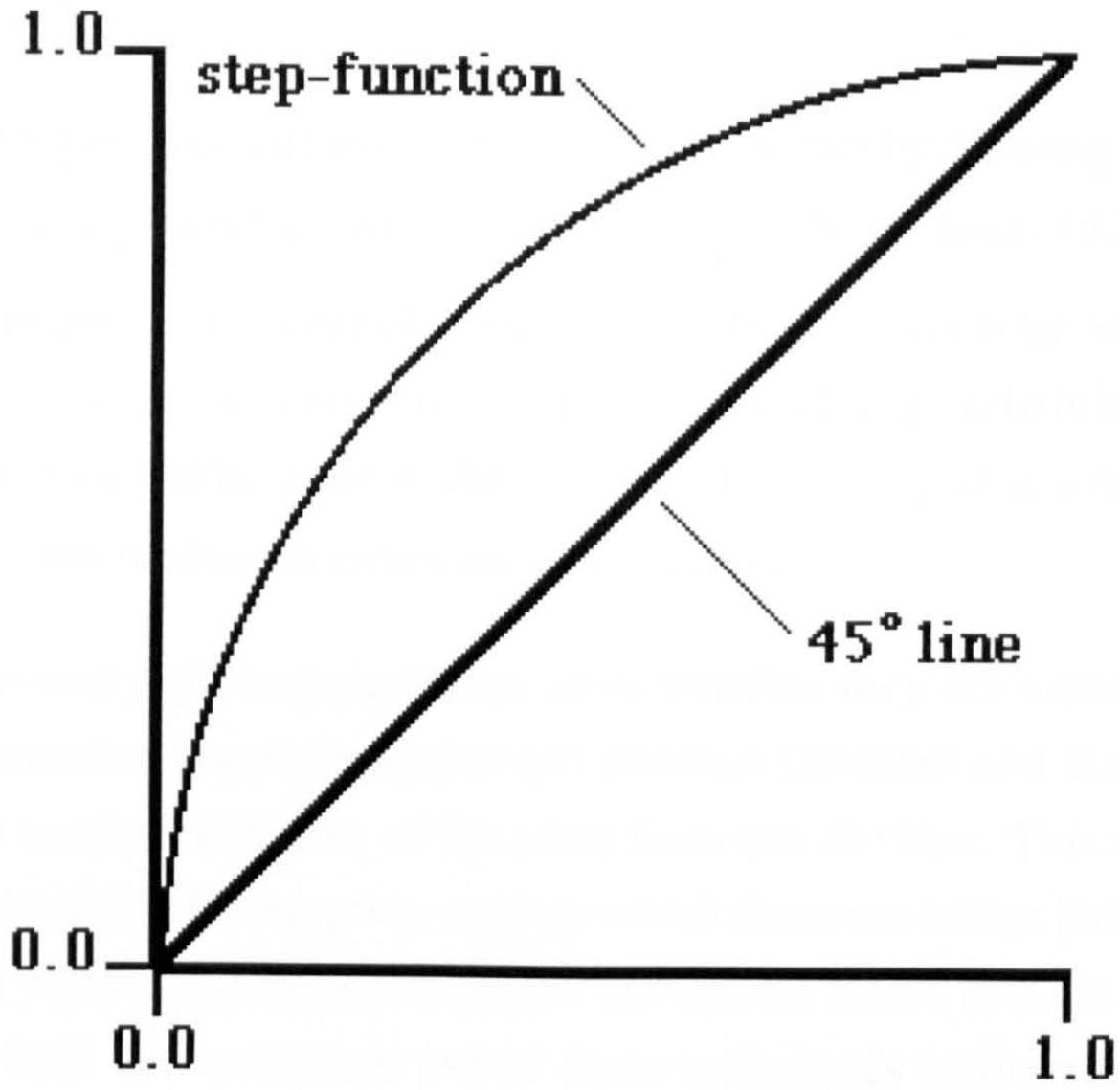


Figure 3.1-2. *u*-plot for a consistently optimistic predictor.

Informally then, the *u*-plot is a powerful means of detecting various kinds of *consistent bias* in predictions, i.e., situations where the prediction errors are in some sense *stationary*.

The recalibration technique, described later in Chapter 4, exploits this ability of the *u*-plot to indicate the nature of the error in a prediction system and uses this to adjust for similar errors in future predictions.

3.2 The y-plot

The second technique, the *y*-plot, is intended to detect non-stationarity in the inaccuracy that can occur for a single prediction system. As with the *u*-plot this technique depends on comparison of the predicted *cdf* and the observations for a range of one-step-ahead predictions. Let

$$x_j = -\log(1-u_j) \qquad j = s, \ldots, q \qquad \ldots\ldots (3.2.1)$$

where u_j is as defined in *(3.1.1)* and

$$y_r = \frac{\sum_{j=s}^{r} x_j}{\sum_{j=s}^{q} x_j} \qquad r = s, \ldots, q \qquad \ldots\ldots (3.2.2)$$

Then the *y*-plot is constructed similarly to the *u*-plot by drawing the sample *cdf* of the *y*'s (this time a step-function with step-size $\frac{1}{q\text{-}s+1}$). Note, from *(3.2.1)* and *(3.2.2)*, that the *y*-plot preserves the order of occurrence of the *u*'s over time whereas the *u*-plot does not. If our prediction system has captured the trend (e.g., reliability growth) in the data then the *y*-plot should be close to the *45°* line. Thus the *y*-plot is a means of detecting those cases where the prediction errors are *non-stationary*.

For the *u*- and *y*-plots, judgements as to whether they are significantly far from the *45°* line are based on the *Kolmogorov* (*K*) distance [Kendall and Stuart 1979] (which is the maximum vertical distance) of the plots from the *45°* line. This test for statistical significance is called the *Kolmogorov* test for which there are tables [Miller 1956]. More details on *u*- and *y*-plots and the Kolmogorov test can be found in [Cox and Lewis 1966] and [DeGroot 1986] and extensive use of these techniques in the context of software reliability prediction is shown in [Keiller et al. 1983b], [Keiller et al. 1983a], [Abdel-Ghaly et al. 1986], [Chan 1986] and [Brocklehurst and Littlewood 1992] .

3.3 The Prequential Likelihood Ratio

It should be noted that it is possible for a prediction system to give good *u*- and *y*-plots and yet still be inaccurate; for example it could be very *noisy*, so that individual predictions emanating from it are very inaccurate even though on average there is no bias, and there is no evidence of non-stationarity in the errors in the predictions. For this reason it is necessary to use a further measure called the *prequential likelihood ratio (PLR)* [Dawid 1984] which is intended as a global comparison of goodness of prediction for one prediction system versus another. Again this measure may be applied over a range of one-step-ahead predictions.

Suppose we have two prediction systems, *A* and *B*, say. Then the *PLR* is defined to be

$$PLR_{s\,i}^{AB} = \frac{\prod_{j=s}^{i} \hat{f}_j^A(t_j)}{\prod_{j=s}^{i} \hat{f}_j^B(t_j)} \qquad \ldots\ldots (3.3.1)$$

Notice that, unlike the *u*- and *y*-plots, this measure depends upon the *pdfs* rather than the *cdfs*.

If $PLR_{s\,i}^{AB} \to \infty$ as $i \to \infty$ then we would choose model A as being the better of the two models. Conversely if $PLR_{s\,i}^{AB} \to 0$ as $i \to \infty$, we would favour model B over model A. As we clearly never have $i \to \infty$ in practice, the best we can do is to look for steady increases and decreases in our *PLR* plots of one model versus another over the whole data set. In the *PLR* analysis that follows in Chapter 8, for convenience $log(PLR_{s\,i}^{AB})$ against i is plotted for the sequence of one-step-ahead predictions, $i = s, ..., q$.

For an intuitive explanation of the *PLR* observe figure 3.3-1. Here the true underlying predictive distribution for the next time to failure, T_j, is shown, together with predictions of this *pdf* from two models, A and B. Clearly the prediction from model A is more accurate than that from model B. The next observation, t_j, is obviously likely to lie in the main body of the true distribution (i.e., where $f_j(t)$ is larger). Since A is closer to the truth than B this means that t_j is likely to lie in regions for which the A *pdf* has a value greater than that of the B *pdf*. Thus the ratio of the A *pdf* to the B *pdf* is likely to be greater than *1*. So, if A is generally closer to the truth than B over a sequence of predictions the $PLR_{s\,i}^{AB}$ defined in *(3.3.1)* will tend to increase with i.

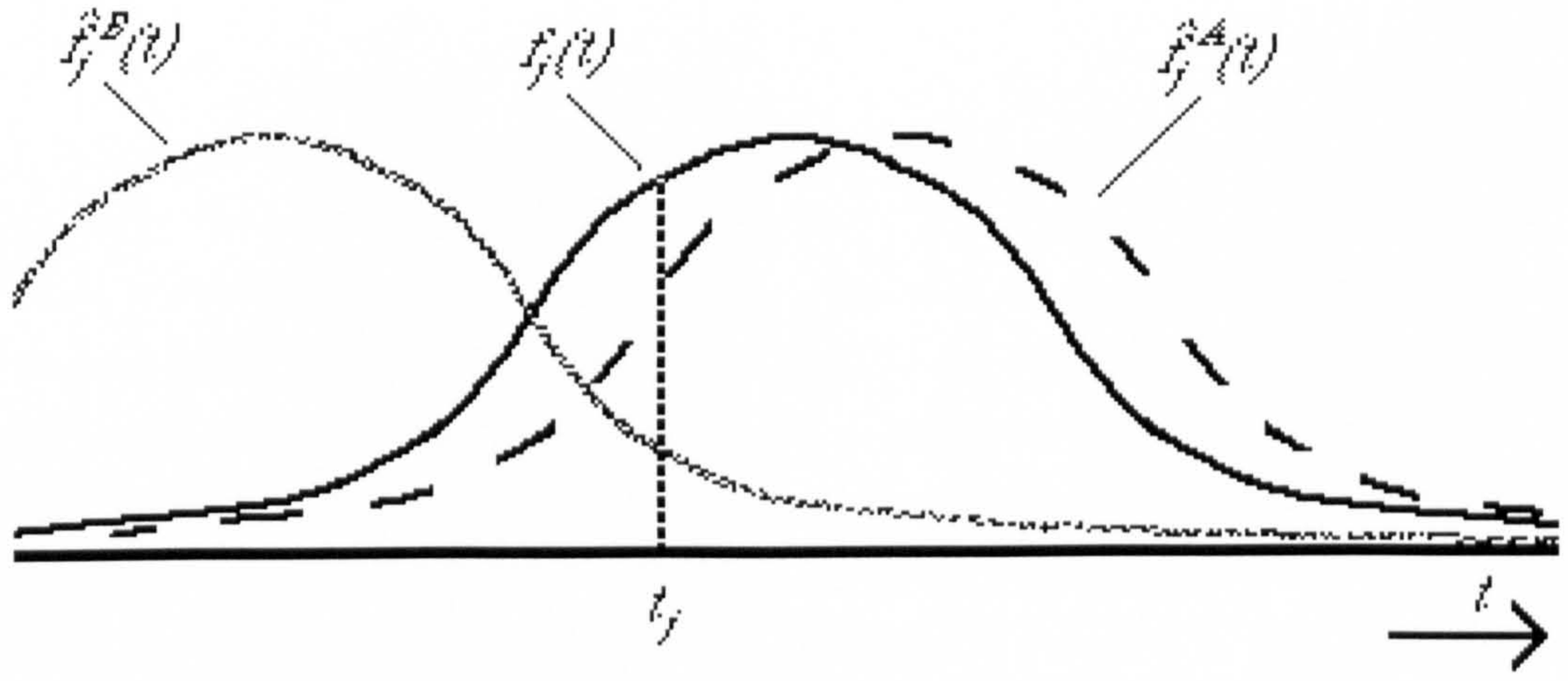


Figure 3.3-1. True predictive *pdf* together with estimates of the *pdf* from two models, A and B.

There is a more formal asymptotic theory behind the *PLR* approach which can be found in [Dawid 1984] and examples of the use of the *PLR* in the context of software reliability prediction can be found in [Abdel-Ghaly et al. 1986], [Chan 1986] and [Brocklehurst and Littlewood 1992].

In practice [Brocklehurst et al. 1991] it is often the case that, according to such a *PLR* analysis, the relative fortunes of the different models tend to switch as the data evolves. So, if this method is used to select a "best" model for future predictions of the data, then it is likely that such a choice will dynamically switch between the various prediction systems as the data evolves. This is discussed further in Chapter 6, where a simple method for automating such dynamic selection is suggested.

4 The Recalibration Technique

It is frequently the case that for some data sets, according to the analysis techniques described in the previous chapter, all of a group of reliability models applied are in error. In order to achieve accurate predictions for such data sets, where there is no single model for which the predictions can be trusted to be accurate, a *recalibration* technique [Littlewood and Keiller 1984; Brocklehurst et al. 1990] has been developed. Recalibration is a *learning* technique, where the nature in the error of *past* predictions, for a single raw prediction system, is assessed and used in order to adjust *future* predictions.

This technique is intended for circumstances where the prediction system has captured the trend in the data, i.e., the prediction errors are stationary, but the predictions are biased (this should result in a good *y*-plot and a bad *u*-plot). In such circumstances, as discussed in section 3.1, the (approximately) consistent departure of the prediction system from the truth is represented by the departure of the *u*-plot from the *45°* line. This is where the *recalibration technique* can use the *u*-plot to eliminate this bias from the raw prediction system in the hope that a new improved prediction system will result. This technique consists of using the joined up *u*-plot, G_i, (see figure 4-1) based on *previous raw predictions* of $t_s,...,t_{i-1}$, in order to adjust the current raw prediction of T_i,

$$\hat{F}_i^*(t) = G_i(\hat{F}_i(t)) \qquad .....(4.1)$$

For example, suppose that the raw predictions are consistently optimistic. As previously stated in section 3.1 this will result in a *u*-plot which is everywhere above the *45°* line. Then application of the recalibration technique may be expected to result in adjustment of the next raw predictive *cdf* in the right direction (from figure 4-2 we can see it will make the predictive *cdf* everywhere larger) and our new recalibrated *cdf* will be closer to the true *cdf* than was the raw *cdf*. Conversely recalibration of a pessimistic prediction would be expected to make the raw predictive *cdf* everywhere smaller, and closer to the true *cdf*, providing the previous predictions were also similarly pessimistic.

The recalibration technique may be expected to efficiently adjust raw predictors which have more complicated departures of the predictions from the truth than simple consistent optimism and pessimism, but it should be noted that the key to recalibration eliminating bias in the raw predictions is that the predictions errors are approximately stationary. Equivalently, the shape of the *u*-plot taken over increasingly large intervals of the raw predictions should be approximately non-changing.

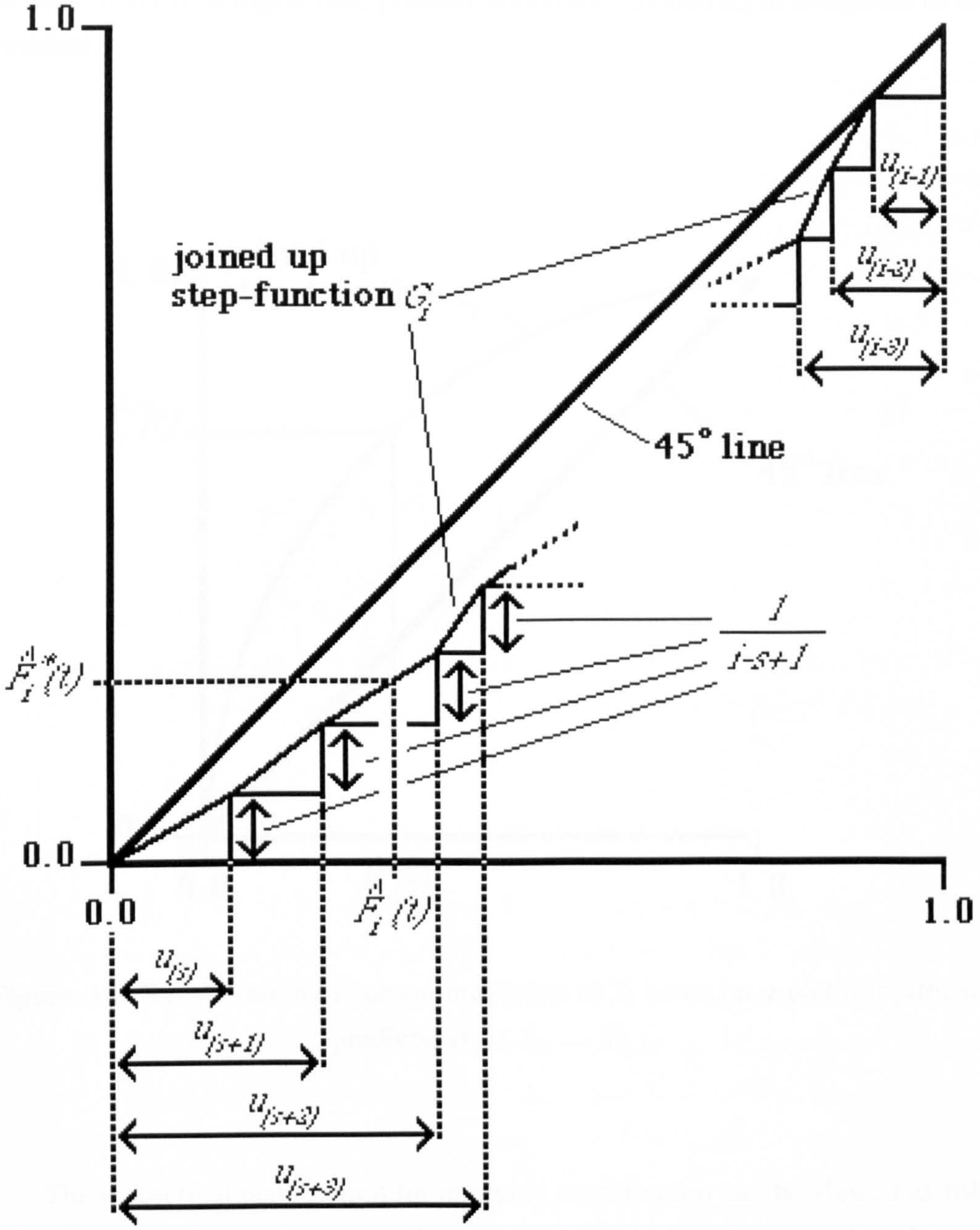


Figure 4-1. The joined up step-function, G_i, of the u-plot of predictions of T_s,..., T_{i-1}.

It is important to point out that after recalibrating a prediction from a raw parametric model the mathematical form of the new recalibrated prediction will be different in structure from the original parametric predictor. It will not correspond to the same mathematical structure with merely adjusted parameters. This is a benefit of the recalibration method. It may produce probabilistic predictions which cannot be adequately described by a parametric model and is thus far more flexible. The resulting predictions are driven by the data for which the predictions are required, as opposed to being

constricted by fairly stringent (and possibly unrealistic) modelling assumptions as are the raw models.

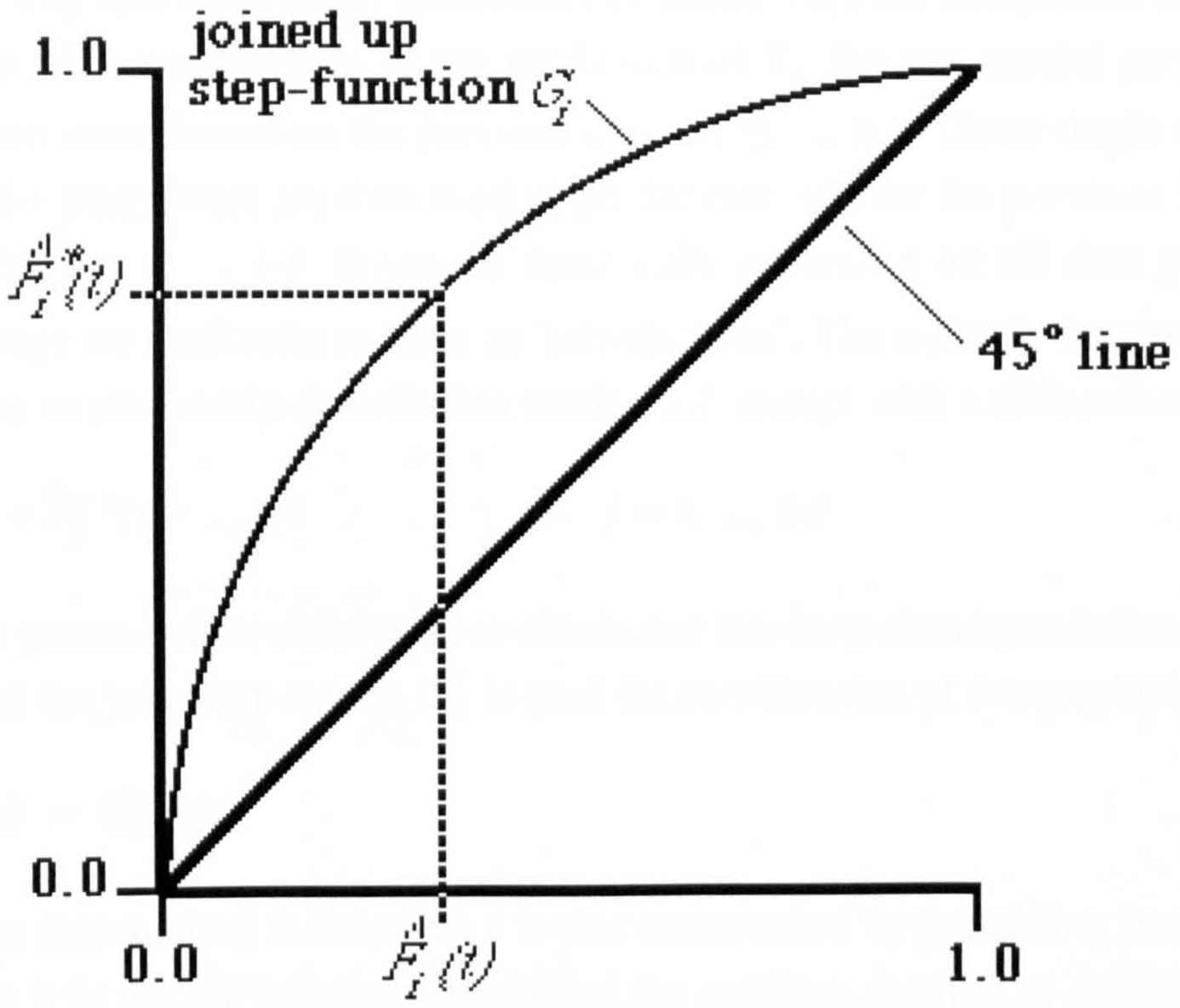


Figure 4-2. Recalibration of current prediction of T_i based on u-plot of optimistic predictions of T_s, ..., T_{i-1}.

The theoretical justification for applying recalibration can be viewed as follows. Let $U_i = \hat{F}_i(T_i)$, and let the *cdf* and *pdf* associated with the random variable T_i be denoted by $F_i(t)$ and $f_i(t)$, respectively and the *cdf* and *pdf* associated with the random variable U_i be denoted by $F_i^u(u)$ and $f_i^u(u)$, respectively. Then,

$$F_i^u(U_i) = F_i(T_i)$$

Hence, from *(4.1)*,

$$\hat{F}_i^*(T_i) = G_i(\hat{F}_i(T_i)) \simeq F_i^u(\hat{F}_i(T_i)) = F_i(T_i) \qquad \text{.....(4.2)}$$

as required.

From *(4.2)* it can be seen that the efficiency of recalibration will indeed depend on how good an estimate the joined up u-plot, G_i, is of the actual *cdf* of U_i, F_i^u. Clearly the

more stationary the errors in the predictions are, the more accurate our estimate of F_i^u, G_i, may be expected to be.

In [Littlewood and Keiller 1984] an alternative method for obtaining our estimate of F_i^u is proposed. This method is similar to the previous method, the only distinction being in the way that the original predictive raw model *cdf*s are obtained in the *u*-plot for recalibration. For recalibration of the prediction of T_i, the raw model parameters are estimated *only once*, based on the previous data, $t_1, t_2, ..., t_{i-1}$. These single estimates of the raw model parameters are then used to get the raw *cdf*s for the *previous* inter-failure times, $\hat{F}_j^{r\ i}(t), j = s, ..., i-1$. Since all these *cdf*s are based on all data *prior* to the prediction stage we shall refer to them as "retrodictions". The *u*-plot is then constructed in the same way as previously described in section 3.1, except with a different set of *u*s,

$$u_j^{r\ i} = \hat{F}_j^{r\ i}(t_j) \qquad j = s, ..., i-1 \qquad .....(4.3)$$

These same parameters are also used to obtain our one-step-ahead prediction of T_i, $\hat{F}_i(t)$, as before and the joined up *u*-plot, G_i^r, is used for recalibration of this prediction,

$$\hat{F}_i^{*r}(t) = G_i^r(\hat{F}_i(t)) \qquad .....(4.4)$$

Since this method is based on a *u*-plot constructed by predicting backwards, we shall refer to it as the *retrodictive* method and the previous method as *predictive* but it is important to point out that both methods for recalibration result in true predictions; the recalibrated one-step-ahead prediction for T_i in each case is only based on past data, t_1, $t_2, ..., t_{i-1}$. It can be seen that the effort in obtaining a single recalibrated prediction for the retrodictive method is much less than that for the predictive method, since only a single optimisation to get the raw model parameters will be performed (as opposed to $i-s+1$ optimisations for the predictive method).

We can repeat this recalibration procedure, using either the predictive (see *(4.1)*) or retrodictive (see *(4.4)*) method for obtaining our estimate of F_i^u (G_i and G_i^r) over $i = p, ..., q$, where $p-s$ is a number suitably large for making the first recalibrated prediction, and we then have a *recalibrated prediction system* for each of the methods,

$$\{\hat{F}_i^*(t), \hat{f}_i^*(t);\ i = p, ..., q\} \qquad .....(4.5)$$

and

$$\{\hat{F}_i^{*r}(t), \hat{f}_i^{*r}(t);\ i = p, ..., q\} \qquad .....(4.6)$$

From *(4.1)* it can be seen that the predictive recalibrated prediction system will consist of a series of one-step-ahead predictions which are at each stage based on a *u*-plot which

contains an increasing range of the raw predictions as the range of data evolves (i.e., one more u is added into the u-plot for recalibration at each successive prediction stage). In contrast, from *(4.4)*, the u-plot for recalibration for the retrodictive method will consist of a completely new set of us at each successive prediction stage. It is clear that the benefit of the small effort required for the retrodictive method as opposed to the predictive method, is more or less irrelevant when considering a series of predictions; to achieve these prediction systems the effort required is $q\text{-}p+1$ optimisations for the retrodictive method as opposed to $q\text{-}s+1$ in the case of the predictive method.

Since the new recalibrated predictions are truly predictive (see *(4.1)* and *(4.4)*) the u- and y-plots and the *PLR* may be used, as outlined in Chapter 3, in exactly the same manner as with the raw prediction system to assess the improvement gained via recalibration every time it is applied. In other words the analysis techniques can be used to assess the accuracy of our new recalibrated prediction systems (see *(4.5)* and *(4.6)*) and compare the improvement over the raw prediction system (see *(2.7)*).

4.1 Further Investigation of the Effectiveness of Recalibration

Early investigation of the predictive recalibration method (see [Littlewood and Keiller 1984] and [Chan and Littlewood 1986]) suggested that the u-plots of the resulting recalibrated prediction systems were an improvement over the u-plot of the raw predictions indicating that the method had, as expected, eliminated bias in the raw model predictions. It was then necessary to check that this decrease in bias was not bought at the expense of another kind of departure of the recalibrated predictions from the truth and that the recalibrated predictions were indeed generally better than the raw. For this purpose it is appropriate to examine the *log(PLR)* plots, as described in section 3.3, for the recalibrated versus the raw predictions. In this analysis genuine improvement would be indicated by steady increases in the plots. In [Chan and Littlewood 1986] it is shown that this comparison based on the *PLR* gave very disappointing results. Increases in the *PLR* plots were not seen, in fact the results suggested that on this global comparison of performance the predictive recalibration method resulted in *worse* predictive accuracy than the raw.

Considering the predictive recalibration method in terms of the *cdf*s only this is a surprising result. Under the conditions that the raw prediction errors are approximately stationary, elimination of bias (or a least improvement with respect to this) would be expected, and by comparison of the u-plots of the recalibrated predictions with the u-plots of the raw predictions, such improvement was evident. Further, since the u-plot used for this recalibration method is only changing slowly as the predictions proceed (only one

new u is added each time into the u-plot for recalibration for the predictive method) it is not expected that sequence of recalibrated *cdf*s should be significantly more noisy than the original raw prediction sequence (although, of course, noise in the original prediction sequence will not be eliminated by recalibration). On examination of the *PLR* in more detail, though, which depends upon the predictive *pdf*s, not the predictive *cdf*s, we can see how it is that the *PLR* measure reports badly about the recalibrated predictions.

From *(2.2)* and *(4.1)* we have the recalibrated *pdf*,

$$\hat{f}^{*}_{i}(t) = \hat{F}^{*\prime}_{i}(t) = g_i(\hat{F}_i(t))\hat{f}_i(t)$$

where $g_i = G_i'$ and so from *(3.1.1)* and *(3.3.1)* the *PLR* for the recalibrated versus the raw predictions is

$$PLR^{*raw}_{pq} = \frac{\prod_{i=p}^{q} \hat{f}^{*}_{i}(t_i)}{\prod_{i=p}^{q} \hat{f}_i(t_i)} = \frac{\prod_{i=p}^{q} g_i(\hat{F}_i(t_i))\hat{f}_i(t_i)}{\prod_{i=p}^{q} \hat{f}_i(t_i)} = \prod_{i=p}^{q} g_i(\hat{F}_i(t_i)) = \prod_{i=p}^{q} g_i(u_i) \quad \ldots\ldots (4.1.1)$$

It can be seen from figure 4-1 that the derivative of G_i, g_i, will be discontinuous and it is suggested in [Chan and Littlewood 1986] that it is this which gives rise to bad results according to the *PLR* for the recalibrated predictions.

6.4146099E+00	3.0098500E+00	5.4329097E-01	2.1422501E+01	1.0320000E+02
5.6656799E+01	5.7857800E+01	1.0448100E+02	3.1793900E+02	5.9356899E+01
2.0398399E+01	9.6389503E+01	9.0094000E-01	9.6875000E-01	4.0360500E+01
2.2058400E+02	9.2947899E+01	1.7023900E+02	3.6217499E+01	9.3968300E+01
2.0196100E+02	4.3393399E+02	3.8121899E+02	1.8233099E+02	3.0445299E+01
1.5192400E+01	1.1890600E+02	6.0466803E+02	1.1561600E+02	1.2200900E+01
1.9694099E+02	8.1464798E+01	2.2856900E+02	1.2170800E+02	7.7423798E+01
1.4530099E+02	1.7540500E+02	6.3112301E+01	1.3447701E+02	5.8479500E+01
2.9466000E+02	1.9296900E+01	4.3926901E+02	1.6530600E+02	2.3589500E+02
2.6872000E+02	1.6284200E+02	3.3893101E+01	4.3545499E+02	9.0464401E+01
4.1136401E+02	4.8315399E+01	9.5015602E+01	5.4050800E+01	1.2778300E+02
2.3063499E+02	6.3438702E+02	6.7494102E+01	8.1353699E+02	3.6915500E+02
6.2985303E+02	2.6394000E+02	4.0917999E+02	2.4075301E+02	1.5421201E+02
1.1430699E+03	4.6192001E+02	7.5745801E+02	1.2333000E+02	2.5070200E+02
1.3537199E+02	2.4265600E+01	3.5220901E+02	4.8819501E+02	8.3803998E+02
2.3425301E+02	3.1668399E+02	2.4603500E+01	1.4490199E+02	1.1750800E+02
6.1494702E+02	2.4644299E+02	4.0198401E+02	4.9175800E+01	3.8631799E+02
1.2851300E+03	2.5958600E+02	2.6665201E+02	8.6291000E+01	6.9162102E+01
1.3110400E+02	1.4163699E+02	3.7624600E+02	4.0350201E+02	8.5777298E+01
3.2732999E+02	1.7768300E+03	1.0564301E+03	1.7535201E+02	2.9024600E+02
3.7211099E+02				

Table 4.1-1. Data set *73* generated by the *DU* model with parameters $\gamma = 0.32$ and $\beta = 0.57$; *101* inter-failure times (read from left to right).

j	$\hat{N}$	$\hat{\phi}$	j	$\hat{N}$	$\hat{\phi}$
20	2.7000000E+01	7.7991700E-04	60	7.5000000E+01	1.5501800E-04
21	2.9000000E+01	7.0884300E-04	61	7.5000000E+01	1.5508800E-04
22	2.7000000E+01	7.8291103E-04	62	7.3000000E+01	1.6226299E-04
23	2.4000000E+01	9.3851698E-04	63	7.5000000E+01	1.5464700E-04
24	2.5000000E+01	9.3553303E-04	64	7.6000000E+01	1.5492800E-04
25	2.6000000E+01	8.6790300E-04	65	7.7000000E+01	1.4793601E-04
26	2.8000000E+01	6.9608801E-04	66	7.9000000E+01	1.4194100E-04
27	3.1000000E+01	5.8793498E-04	67	7.5000000E+01	1.5558900E-04
28	3.4000000E+01	5.3284201E-04	68	7.6000000E+01	1.5158400E-04
29	3.0000000E+01	6.2939897E-04	69	7.7000000E+01	1.5125499E-04
30	3.3000000E+01	5.5758603E-04	70	7.8000000E+01	1.4410701E-04
31	3.6000000E+01	4.7725299E-04	71	8.0000000E+01	1.3726900E-04
32	3.7000000E+01	4.3274401E-04	72	8.2000000E+01	1.3139901E-04
33	4.0000000E+01	3.8166801E-04	73	8.5000000E+01	1.2327600E-04
34	4.0000000E+01	3.8417001E-04	74	8.6000000E+01	1.2089800E-04
35	4.4000000E+01	3.4150199E-04	75	8.8000000E+01	1.1821800E-04
36	4.7000000E+01	2.9771999E-04	76	8.6000000E+01	1.2066800E-04
37	5.0000000E+01	2.8047699E-04	77	8.8000000E+01	1.1565500E-04
38	5.1000000E+01	2.6416799E-04	78	9.0000000E+01	1.1357800E-04
39	5.5000000E+01	2.3784200E-04	79	9.2000000E+01	1.0707600E-04
40	5.8000000E+01	2.2075701E-04	80	9.6000000E+01	1.0112000E-04
41	6.4000000E+01	1.9311201E-04	81	9.8000000E+01	9.5897798E-05
42	6.1000000E+01	2.0606500E-04	82	9.8000000E+01	9.5755400E-05
43	6.9000000E+01	1.7431000E-04	83	1.0000000E+02	9.2545997E-05
44	6.2000000E+01	2.0657400E-04	84	1.0200000E+02	9.1014903E-05
45	6.3000000E+01	1.9739600E-04	85	1.0400000E+02	8.6796201E-05
46	6.4000000E+01	1.9259000E-04	86	1.0500000E+02	8.5462598E-05
47	6.5000000E+01	1.9244300E-04	87	1.0100000E+02	9.1326699E-05
48	6.6000000E+01	1.8426700E-04	88	1.0300000E+02	8.8208697E-05
49	7.2000000E+01	1.6284399E-04	89	1.0500000E+02	8.5285101E-05
50	6.8000000E+01	1.8084201E-04	90	1.0800000E+02	8.1267499E-05
51	7.2000000E+01	1.6622900E-04	91	1.1200000E+02	7.6325501E-05
52	6.9000000E+01	1.7290600E-04	92	1.1600000E+02	7.3119401E-05
53	7.4000000E+01	1.5600600E-04	93	1.1900000E+02	7.0182403E-05
54	7.9000000E+01	1.4487200E-04	94	1.1900000E+02	6.9282301E-05
55	8.4000000E+01	1.3072800E-04	95	1.2000000E+02	6.8366702E-05
56	8.9000000E+01	1.2274200E-04	96	1.2400000E+02	6.4871499E-05
57	8.8000000E+01	1.2282400E-04	97	1.2700000E+02	6.3220301E-05
58	8.0000000E+01	1.4278700E-04	98	1.1800000E+02	7.1403003E-05
59	8.3000000E+01	1.3316800E-04	99	1.1600000E+02	7.3642099E-05
			100	1.1800000E+02	7.0247101E-05
			101	1.2000000E+02	6.8214802E-05

Table 4.1-2. Successively estimated parameters, $\hat{N}$ and $\hat{\phi}$, when the *JM* model is applied to data set 73 generated by the *DU* model (shown in table 4.1-1). At each stage, j, the estimated parameters are based on inter-failure times, $t_1, t_2, \ldots, t_{j-1}$.

Since, in practice, most reliability measurements of interest (for example, *(2.3)* and *(2.6)*) depend upon the predictive *cdf*s and not on the *pdf*s, these bad reports on the recalibrated predictor, according to the *PLR*, may not be of concern. However, further

investigation of the behaviour of these prediction systems is required so that it can be ascertained whether the *PLR* results in this case are of concern. A good way to do this is to simulate data so that the truth, $F_i(t)$, is known, and comparison of the recalibrated and the raw predicted *cdf*s can be made against the truth, to see which is actually closer.

A simulation such as this was conducted and consisted of randomly generating *100* samples of the realisations, $t_1, t_2, ..., t_{101}$ of times between failures from each of *5* of the parametric models referred to in section 2.1, *JM*, *D*, *L*, *LV* and *KL* (with constant parameters for each model). For example, table 4.1-1, above, shows the inter-failure time data generated for sample number *73*, from the *DU* model, using parameters $\gamma = 0.32$ and $\beta = 0.57$ [Crow 1977].

Then these same models were fitted to the data (for example to each *JM* sample models *D*, *L*, *LV* and *KL* were applied) to form combinations of generated samples with fitted models. Each combination of generated data with fitted model will be referred as a *case* in the following text. The models were repeatedly applied as described in Chapter 2, with, in the notation of *(2.7)*, $s = 20$ and $q = 101$. Table 4.1-2 shows the resulting estimated parameters, $\hat{N}$ and $\hat{\phi}$ [Jelinski and Moranda 1972], when the *JM* model is applied to the data of table 4.1-1.

Each of the resulting raw prediction systems from each combination of generated raw data and fitted model (i.e., each case) was then recalibrated using the retrodictive and predictive methods described previously in this chapter. So, for example, to obtain the recalibrated predictions of T_{101}, for the retrodictive method, only the single last estimates of the parameters shown in table 4.1-2 ($\hat{N} = 1.2000000E+02$ and $\hat{\phi} = 6.8214802E-05$) were used in the *u*-plot for recalibration, whereas for the predictive method all the estimated parameters for $j = 20, 21, ..., 100$ shown in table 4.1-2 were used in the *u*-plot for recalibration.

Two simple criteria were used in order to assess which of the recalibrated and raw *cdf*s were closer to the true *cdf* at each prediction stage *i* in each case. The first consisted of simply observing which of the medians (see *(2.6)*) were closer to the true median, i.e., which of $\hat{w}_i^* = \hat{F}^{*-1}_i(0.5)$ (or for the retrodictive method $\hat{w}_i^{*r} = \hat{F}^{*r-1}_i(0.5)$) or $\hat{w}_i = \hat{F}^{-1}_i(0.5)$ were closer to the true median, $w_i = F^{-1}_i(0.5)$. The second criterion was based on the Kolmogorov (K) distance (the maximum vertical distance) of the predicted *cdf*s from the true *cdf*.

Let

$$\hat{d}_i(t) = \hat{F}_i(t) - F_i(t), \qquad t \geq 0 \qquad \ldots\ldots (4.1.2)$$

$$\hat{d}_i^*(t) = \hat{F}_i^*(t) - F_i(t), \qquad t \geq 0 \qquad \text{.....(4.1.3)}$$

$$\hat{d}_i^{*r}(t) = \hat{F}_i^{*r}(t) - F_i(t), \qquad t \geq 0 \qquad \text{.....(4.1.4)}$$

Then the K distances are

$$\hat{k}_i = \sup_{t \geq 0} |\hat{d}_i(t)| = |\hat{d}_i(\hat{t}_i)| = |\hat{d}_i| \qquad \text{.....(4.1.5)}$$

$$\hat{k}_i^* = \sup_{t \geq 0} |\hat{d}_i^*(t)| = |\hat{d}_i^*(\hat{t}_i^*)| = |\hat{d}_i^*| \qquad \text{.....(4.1.6)}$$

$$\hat{k}_i^{*r} = \sup_{t \geq 0} |\hat{d}_i^{*r}(t)| = |\hat{d}_i^{*r}(\hat{t}_i^{*r})| = |\hat{d}_i^{*r}| \qquad \text{.....(4.1.7)}$$

Based on this criterion the predictor which has the smallest K distance is considered to be the most accurate predictor.

These criteria are used to compare the performance of the recalibrated predictions against the raw predictions but it will be shown later that plots of $\hat{d}_i$, $\hat{d}_i^{r\,j}$, $\hat{d}_i^*$ and $\hat{d}_i^{*r}$, and $\hat{w}_i - w_i$, $\hat{w}_i^{r\,j} - w_i$, $\hat{w}_i^* - w_i$ and $\hat{w}_i^{*r} - w_i$, against i (where $\hat{d}_i^{r\,j}$ and $\hat{w}_i^{r\,j}$ are the corresponding measures for the retrodictions $\hat{F}_i^{r\,j}(t)$ in *(4.3)*), can also be very informative.

The retrodictive method of recalibration was investigated first with comparison limited to predictions of T_{101}. That is, for each of the generated data sets, each of the model parameters were estimated once, based on $t_1, t_2, ..., t_{100}$ to obtain $\hat{F}^r{}_j^{101}(t)$, $j = 20, ..., 100$ and $\hat{F}_{101}(t)$, and

$$u^r{}_j^{101} = \hat{F}^r{}_j^{101}(t_j) \qquad j = 20, ..., 100$$

were used to obtain G_{101}^r and hence $\hat{F}_{101}^{*r}(t)$, according to *(4.4)*. A summary of the results of comparison of the resulting recalibrated *cdf*s, and raw *cdf*s, against the true *cdf* for T_{101}, based on the K distance and median criteria, is shown in table 4.1-3. This table shows the results for all cases and the results for those cases which correspond to some criteria of interest listed in the first column. u^r and y^r are the u- and y-plots of the retrodictions at stage *101*, i.e., the $\hat{F}^r{}_j^{101}(t)$. Let N be the total number of cases, N_c be the number of cases which comply with the criterion, c, listed in the first column and n_c be the number of cases which comply with c and for which the retrodictive recalibrated prediction for T_{101} is closer to the true prediction (according to either the K distance or median predictions) than the raw. The percentages $100\frac{N_c}{N}$, of the total cases which fit into each criterion, c, listed in the first column are shown in the second column and the remaining percentages are the proportion of *these* cases for which the retrodictive recalibrated prediction for T_{101} is better than the raw, i.e., $100\frac{n_c}{N_c}$.

CRITERION *c*	CASES %	K %	MEDIAN %
All cases	100	20	44
u^r significant	5	67	64
u^r significant at *1%*	2	88	82
y^r insignificant	71	18	45
y^r insignificant at *20%*	54	15	41
u^r significant, y^r insignificant	3	60	60

Table 4.1-3. Summary of performance of retrodictive recalibrated predictions of T_{101} compared with raw predictions; %s shown are the proportion of cases which are applicable for each criterion, *c*, and for which the recalibrated predictions are better than the raw. Unless otherwise listed significance levels for *u*- and *y*-plots are *5%*.

The results from this comparison were very disappointing. According to the *K* distance criterion only *20%* of all the recalibrated *cdf*s were closer to the true *cdf*s than the raw *cdf*s; this figure for the comparison based on median predictions is somewhat better but still only *44%* of the recalibrated median predictions were closer to the true medians than the raw median predictions. An important observation here is that the majority of the u^r-plots on which this recalibration is based (i.e., the G^r_{101}) are good. Based on the *K* test described in section 3.2 only *5%* of these u^r-plots are significant at the *5%* level. Limiting the comparison to only these cases where the u^r-plots indicate significant error in the retrodictions the percentage where the recalibrated predictions are closer to the true predictions increase to *67%* and *64%* for the *K* distance and median criteria, respectively. If we limit selection further to u^r-plots which are significant at the *1%* level these percentages improve even more to *88%* and *82%*. For most of the cases, then, the *K* distance gives worse results than the median predictions when good u^r-plots are included in the comparison, but these criterion give increasingly similar results as good u^r-plots are excluded.

From table 4.1-3 it can be seen that most of the y^r-plots (*71%*) are insignificant at the *5%* level. Limiting comparison to just these cases with good y^r-plots little improvement is seen over the results when evidence from the y^r-plots is not taken into account; in fact, in some cases results are marginally worse when limiting the comparison to good y^r-plots.

Table 4.1-4 shows the results when all the retrodictive recalibrated predictions of T_{40}, ..., T_{101} are considered. So here, a different G^r_i is used to construct the recalibrated

predictions for each T_i. Some criteria of interest, c, are listed in the first column. These criteria have been extended to include the u-plots of the sequence of retrodictive recalibrated predictions themselves, u^{*r}, in order to assess whether this method eliminates bias from the raw predictions. As with table 4.1-3, the percentage of the total cases, $100\ \frac{N_c}{N}$ which fit into each criterion, c, are shown in the second column. Let p_m be the percentage of the predictions of T_{40}, ..., T_{101}, for which the recalibrated *cdf*s are closer to the true *cdf*s than the raw (according to either the K distance or median predictions) for a single case, m. Let n_c be the number of cases which comply with criterion c for which $p_m > 50\%$. The remaining percentages are then $100\ \frac{n_c}{N_c}$. In other words they are the percentage of cases for which the recalibrated predictions are better than the raw for the *majority* of the prediction *sequence* for each case.

CRITERION c	CASES %	K %	MEDIAN %
All cases	100	9	43
u^r significant	5	46	58
u^r significant at *1%*	2	63	75
y^r insignificant	71	8	40
y^r insignificant at *20%*	54	7	38
u^r significant, y^r insignificant	3	47	56
u^{*r} insignificant	55	8	45
u^{*r} insignificant at *20%*	41	8	43
u^r significant, u^{*r} insignificant	3	43	62
u^r significant at *1%*, u^{*r} insignificant	1	56	72
y^r insignificant, u^{*r} insignificant	46	7	42
u^r significant, y^r insignificant, u^{*r} insignificant	2	41	56
u^{*r} better than u^r	16	17	45
u^r significant, u^{*r} better than u^r	3	42	60
u^r significant at *1%*, u^{*r} better than u^r	2	52	68
y^r insignificant, u^{*r} better than u^r	14	14	41
u^r significant, y^r insignificant, u^{*r} better than u^r	2	39	54

Table 4.1-4. Summary of performance of retrodictive recalibrated predictions of T_{40}, ..., T_{101}, compared with raw predictions; %s shown are the proportion of cases which are applicable for each criterion, c, and for which the recalibrated predictions are better than the raw. Unless otherwise listed significance levels for u- and y-plots are *5%*.

Comparing these results with the results in table 4.1-3 it can be seen that the results over the sequence of predictions are generally worse than just considering the prediction of T_{101}, particularly for comparison based on the K distance. For the smaller values (less than *50%* in table 4.1-3) this may be partly because the measure used in table 4.1-4 will tend to exaggerate the results in table 4.1-3 but for the larger values this cannot be the case. It seems then that performance of the earlier recalibrated predictions is not as good as performance at T_{101}. This may be because fewer u values are used in recalibration of early raw predictions than of later raw predictions or because there is less bias in the early raw predictions than in the later raw predictions. Apart from the percentages being generally lower in analysis of the sequence of predictions the pattern for performance based on the criteria listed in the first column is similar for both tables, with figures increasing as good u^r-plots are excluded while again little improvement is seen by limiting the analysis to good y^r-plots, and sometimes this makes things marginally *worse*.

Examination of the results based on the u^{*r}-plots in table 4.1-4 shows that approximately $\frac{1}{2}$ of these plots are insignificant at the *5%* level. Limiting comparison to just these cases with good u^{*r}-plots gives no improvement in the results. Only *16%* of cases resulted in u^{*r}-plots which were better than their corresponding u^r-plots and little improvement is seen when the comparison is limited to just these cases. Notice that about *50%* of cases for which the u^r-plots are significant at the *5%* level resulted in better u^{*r}-plots, and most cases for which the u^r-plots are significant at the *1%* level resulted in better u^{*r}-plots. These observations indicate that the retrodictive recalibration method does not appear to eliminate bias present in the raw predictions.

To summarise then, performance of the retrodictive recalibrated predictions is generally worse according to the K distance than according to the medians but the percentages which show improvement over the raw predictions come into closer agreement as good u^r-plots are excluded from the comparison. Further, the overriding factor affecting whether improvement is seen in the recalibrated predictors over the raw is whether the u^r-plot is bad or not, but very few cases actually result in bad u^r-plots. This indicates that either most of the raw one-step-ahead predictions are unbiased or that they are biased *but the retrodictive method for estimating this bias is inaccurate*. To gain more insight into these observations some examples will now be considered in detail.

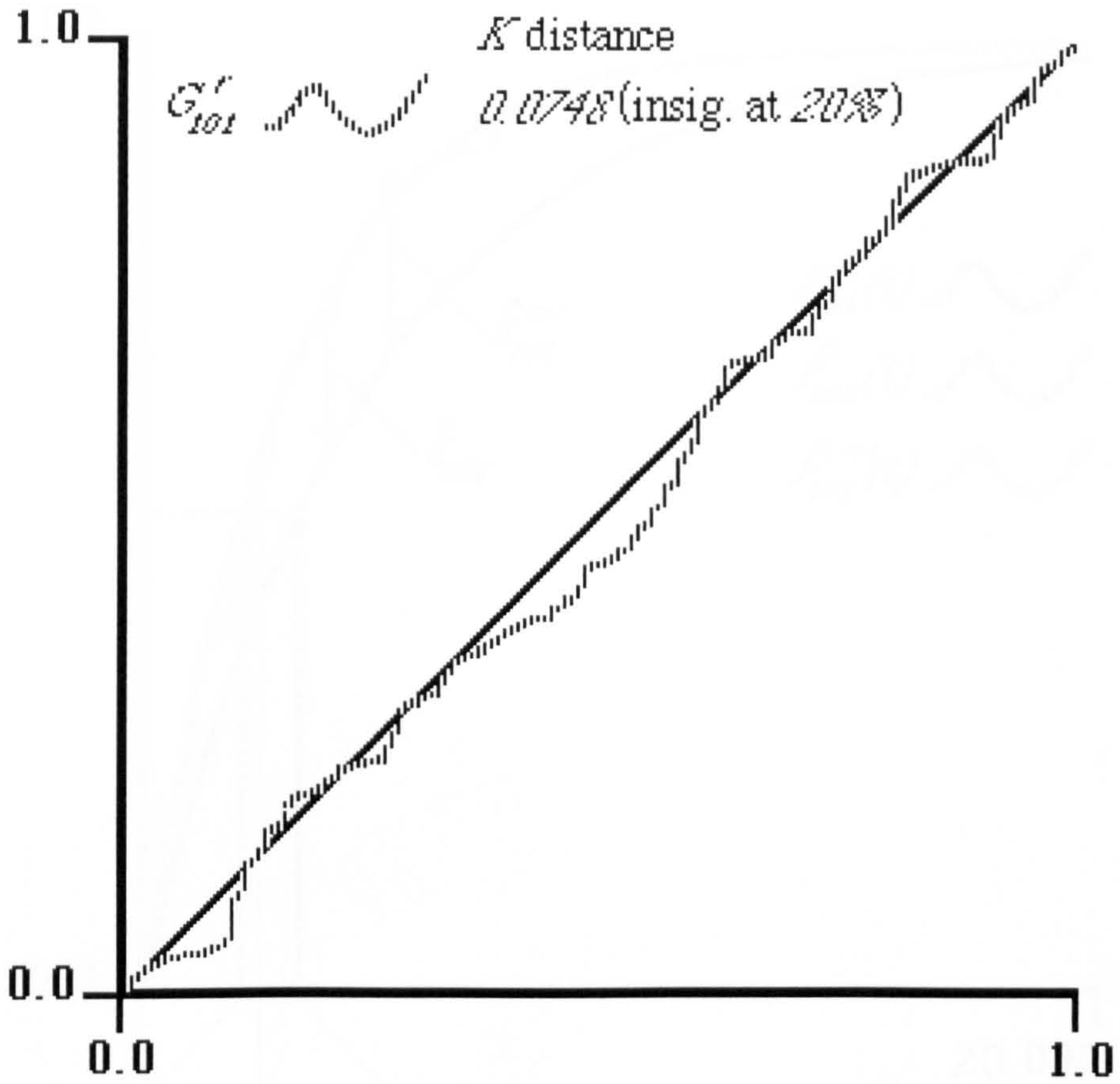


Figure 4.1-1. u^r-plot of retrodictions of T_{20}, ..., T_{100} (G_{101}^r) from the raw *DU* model and data set *7* generated by the *L* model.

In those cases where the u^r-plots are good G_{101}^r is approximately the identity function and so the recalibrated and raw *cdf*s can only be marginally different. The median results, that approximately $\frac{1}{2}$ of the raw are closer to the truth than the recalibrated, would be expected in such circumstances. The *K* distance results however favours the raw *cdf*s in far more proportions of the cases than $\frac{1}{2}$. There is, in fact, a simple explanation for this behaviour. If the u^r-plot used in the recalibration is good, then the G_{101}^r function, even though it is close to the *45°* line, will tend to be rather "irregular" (see figure 4.1-1) due to the randomness of the failure data. This results in the recalibrated *cdf* also being irregular (see figure 4.1-2) whereas the raw and true *cdf*s are smooth. It can be seen how in such cases, where there is little difference between the raw and recalibrated *cdf*s, the *K* distance criterion will tend to discriminate against the recalibrated *cdf*.

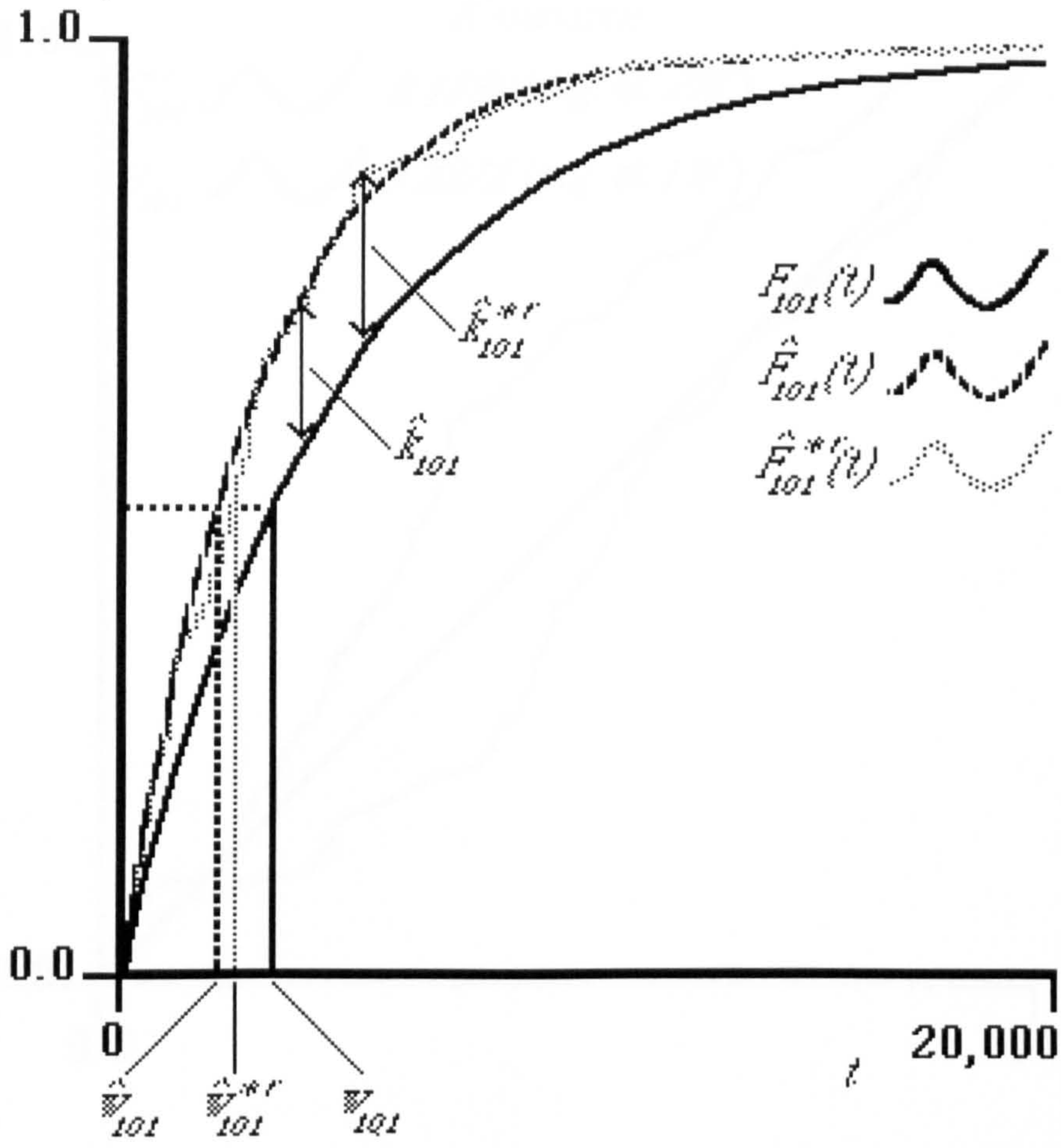


Figure 4.1-2. True, raw and retrodictive recalibrated *cdf*s of T_{101} for the *DU* model and data set *7* generated by the *L* model.

In any case, it is clear that even in cases where the u^r-plots (G^r_{101}) used in this method are significantly bad, the result that in only about *60%* of cases the retrodictive method of recalibration gives improvement is disappointing. Examining an example in more detail we can see the behaviour of the various predictions.

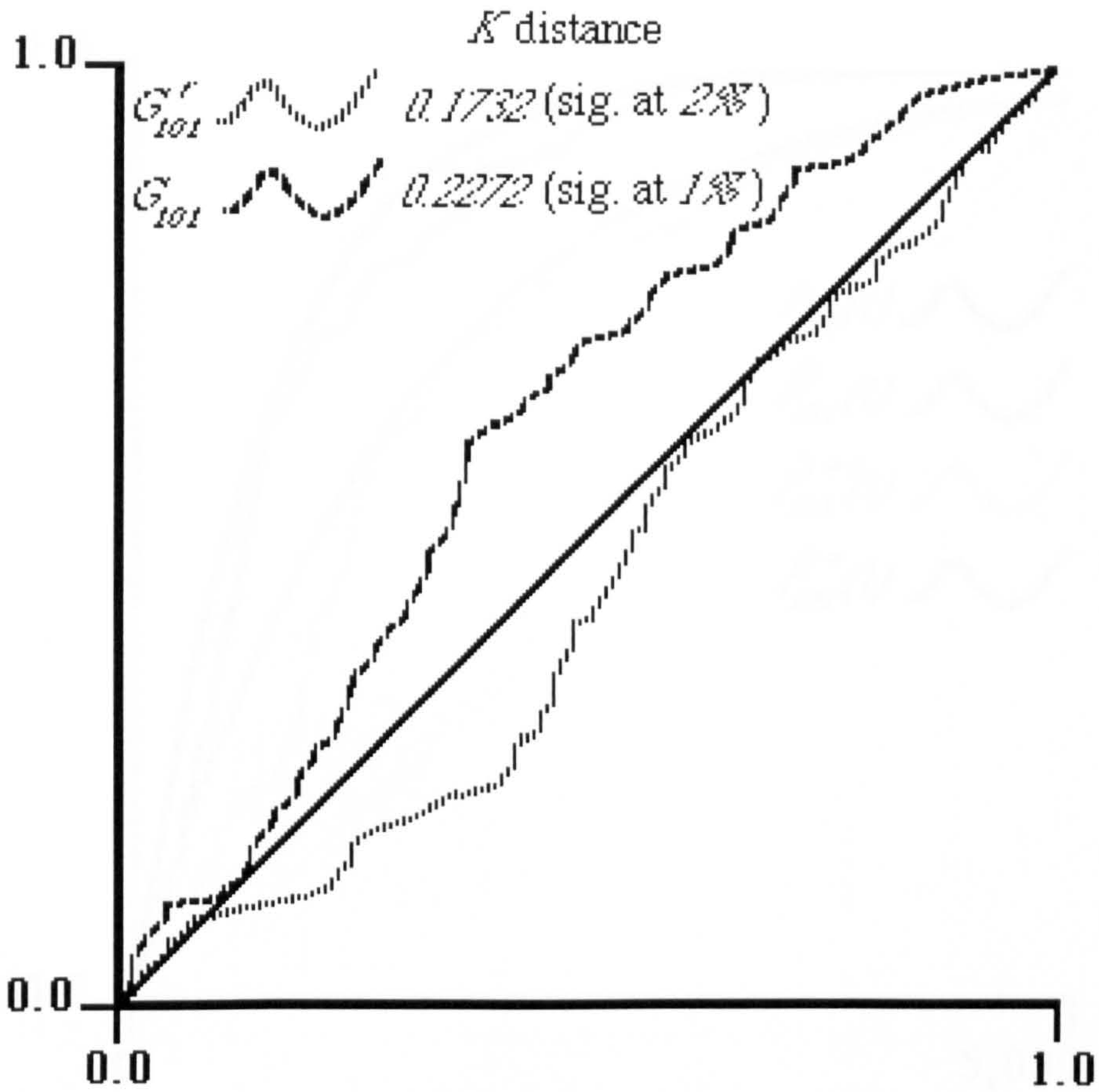


Figure 4.1-3. u^r-plot of retrodictions of T_{20}, ..., T_{100} (G^r_{101}) and u-plot of one-step-ahead predictions of T_{20}, ..., T_{100} (G_{101}) from the raw *JM* model and data set *75* generated by the *LV* model.

Consider figure 4.1-3 which shows the two u-plots which are used in the retrodictive and predictive methods of recalibration of a one-step-ahead prediction of T_{101} from the raw *JM* model. The u^r-plot for the retrodictive method is significant at the *2%* level and the shape of this plot indicates that the retrodictions are, on average, *pessimistic*. For the predictive method the u-plot is significant at the *1%* level, but in this case the shape of the plot indicates that the raw one-step-ahead predictions are, on average, *optimistic*. The one-step-ahead prediction to be recalibrated for both methods is the same, i.e., $\hat{F}_{101}(t)$, and clearly, in this particular case, each method will adjust this *cdf* in *opposite* directions.

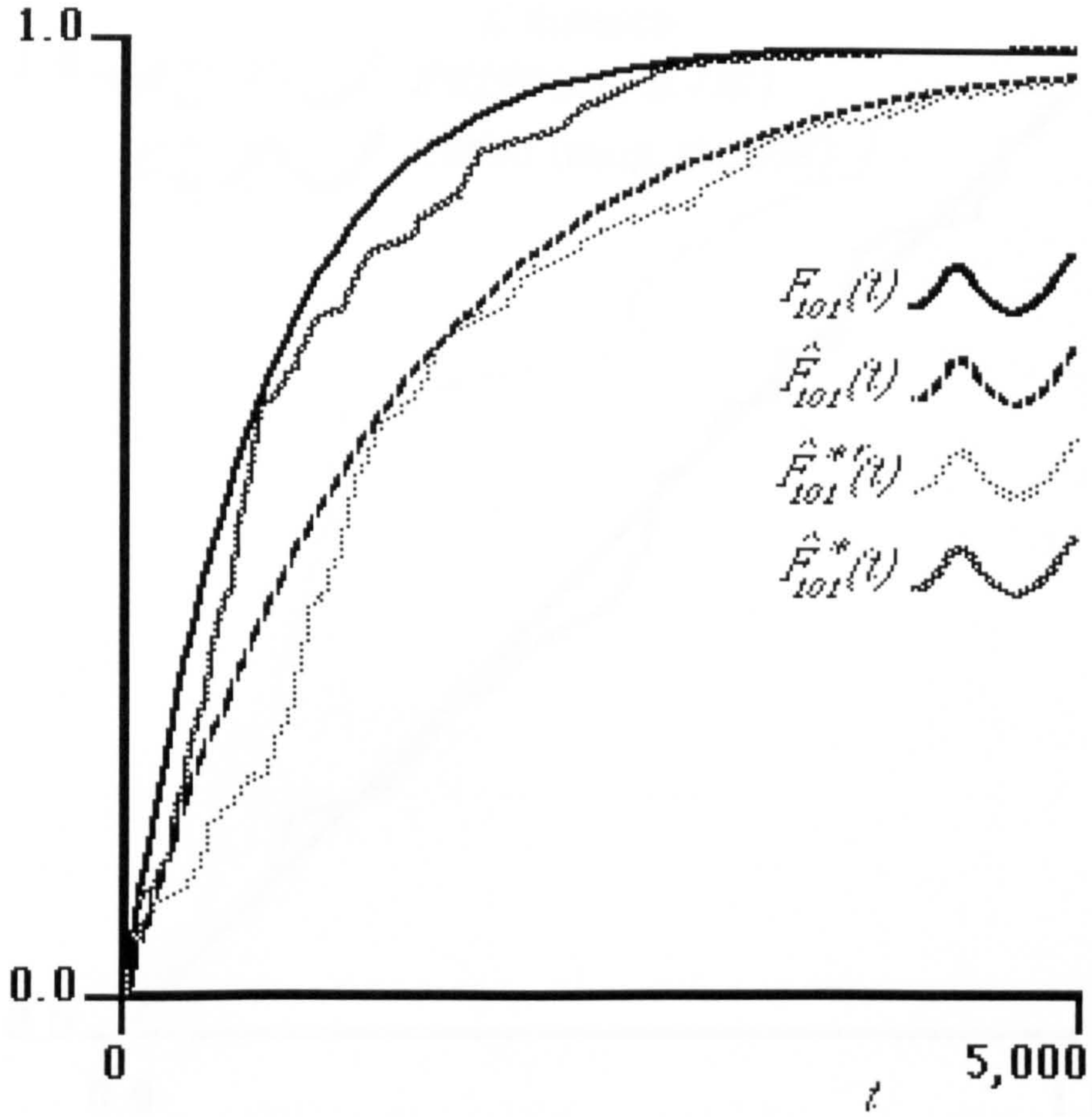


Figure 4.1-4. True and raw *cdf*s of T_{101} together with *cdf*s resulting from retrodictive and predictive methods of recalibration, based on the *u*-plots in figure 4.1-3 for the *JM* model and data set *75* generated by the *LV* model.

From figure 4.1-4, which shows the true, raw and recalibrated *cdf*s resulting from both methods, it is quite clear that the retrodictive method is adjusting in the wrong direction resulting in a *cdf* further from the truth than the original raw *cdf*, while the predictive method results in a *cdf* which is much closer to the true *cdf* than was the raw *cdf*.

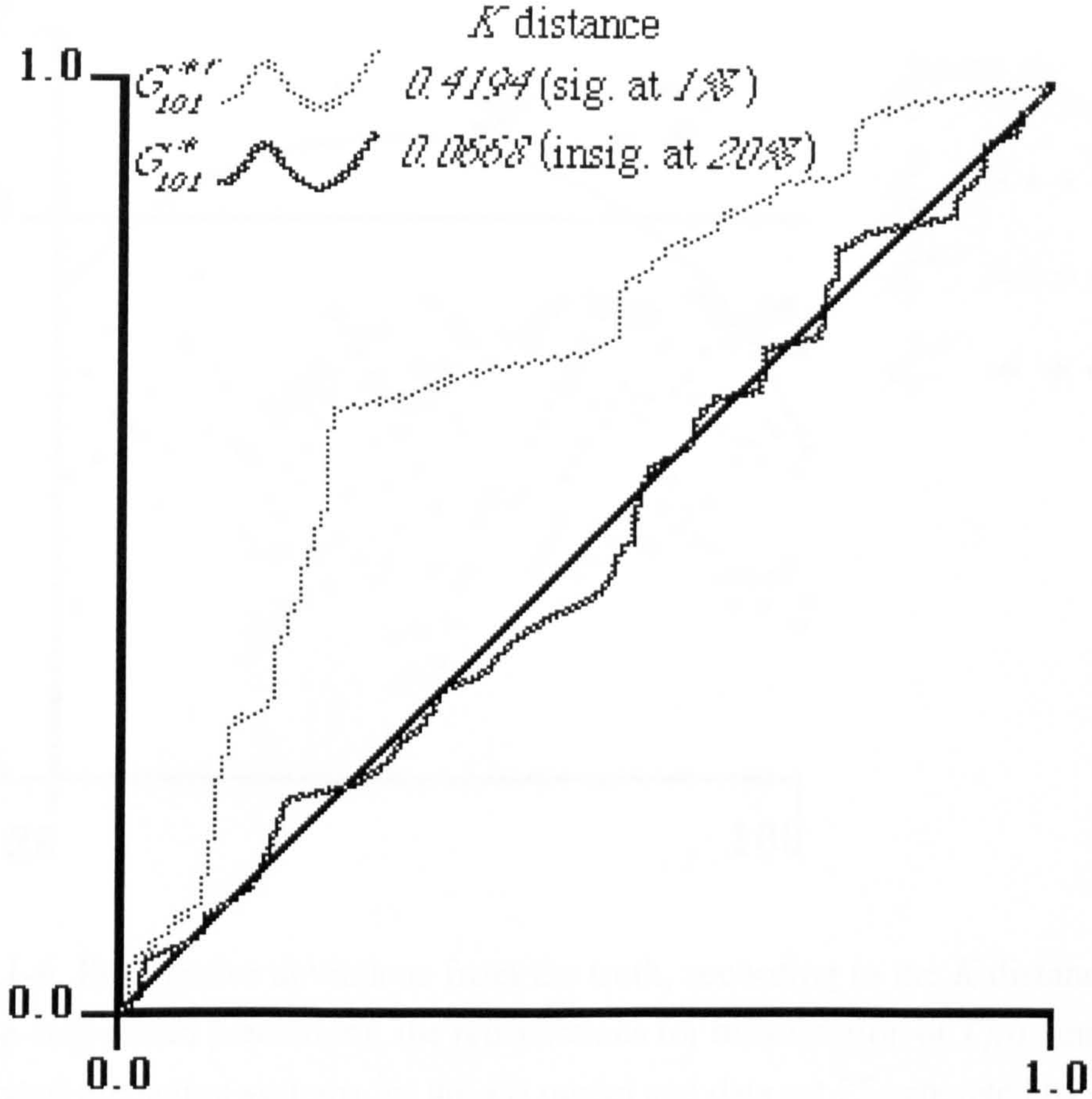


Figure 4.1-5. u^{*r}-plot of retrodictive recalibrated predictions of T_{40}, ..., T_{100} (G^{*r}_{101}) and u^{*}-plot of predictive recalibrated predictions of T_{40}, ..., T_{100} (G^{*}_{101}) for the *JM* model and data set *75* generated by the *LV* model.

Figure 4.1-5 shows the *u*-plots for the recalibrated predictions of T_{40}, ..., T_{101} themselves when the two methods are successively applied over the raw predictions for this data set. These *u*-plots indicate that the predictive method has indeed eliminated the bias from the raw predictions (the u^{*}-plot is insignificant at the *20%* level) while the retrodictive method has not (the u^{*r}-plot is significant at the *1%* level). The u^{*r}-plot indicates that the retrodictive recalibrated predictions are highly optimistic.

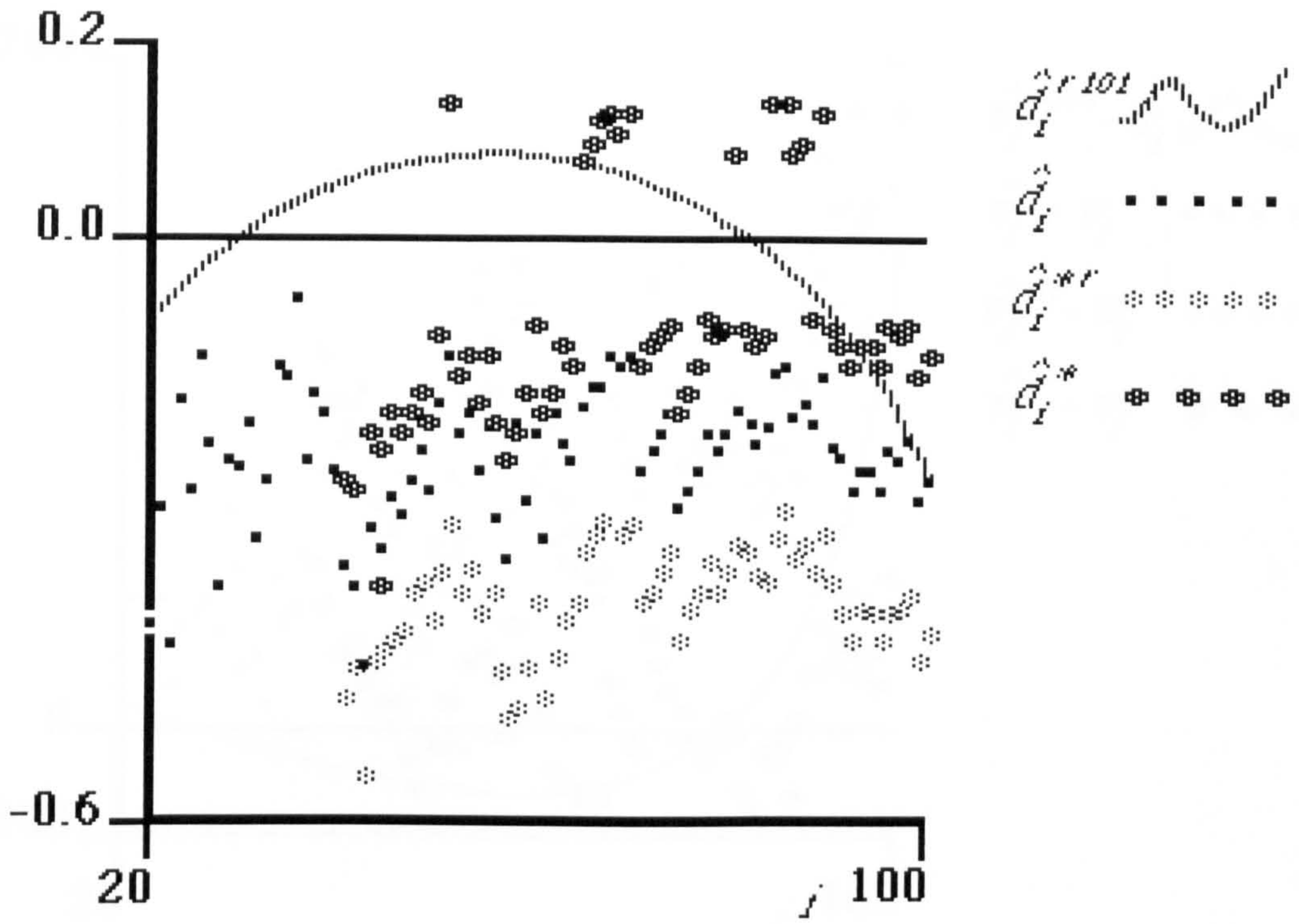


Figure 4.1-6. Progressive deviations from the truth, according to the *K* distance, of the raw one-step-ahead predictions, the retrodictions for recalibration of T_{101}, and both recalibrated prediction systems for the *JM* model and data set *75* generated by the *LV* model.

Figures 4.1-6 and 4.1-7 show the departures (see pages 38-39) of *cdf*s for the successive raw one-step-ahead predictions, the retrodictions (made at prediction stage *101*), and the predictions resulting from the two methods of recalibration, from the true *cdf*s as the data evolves. It can be seen that the raw one-step-ahead predictions are fairly consistently optimistic throughout the data. In contrast the errors in the retrodictions made for recalibration of T_{101} are highly non-stationary with, broadly speaking, optimism only present as the retrodictions approach $i = 101$. As a result it can be seen how there is insufficient adjustment for optimism in the raw prediction of T_{101} via the retrodictive method of recalibration and evidence from the u^{*r}-plot and from the deviations from the truth of the retrodictive recalibrated predictions shown in figures 4.1-6 and 4.1-7 suggests that this is the case throughout the data. For the predictive method it can be seen that the resulting recalibrated predictions are adjusted towards the truth, throughout the data.

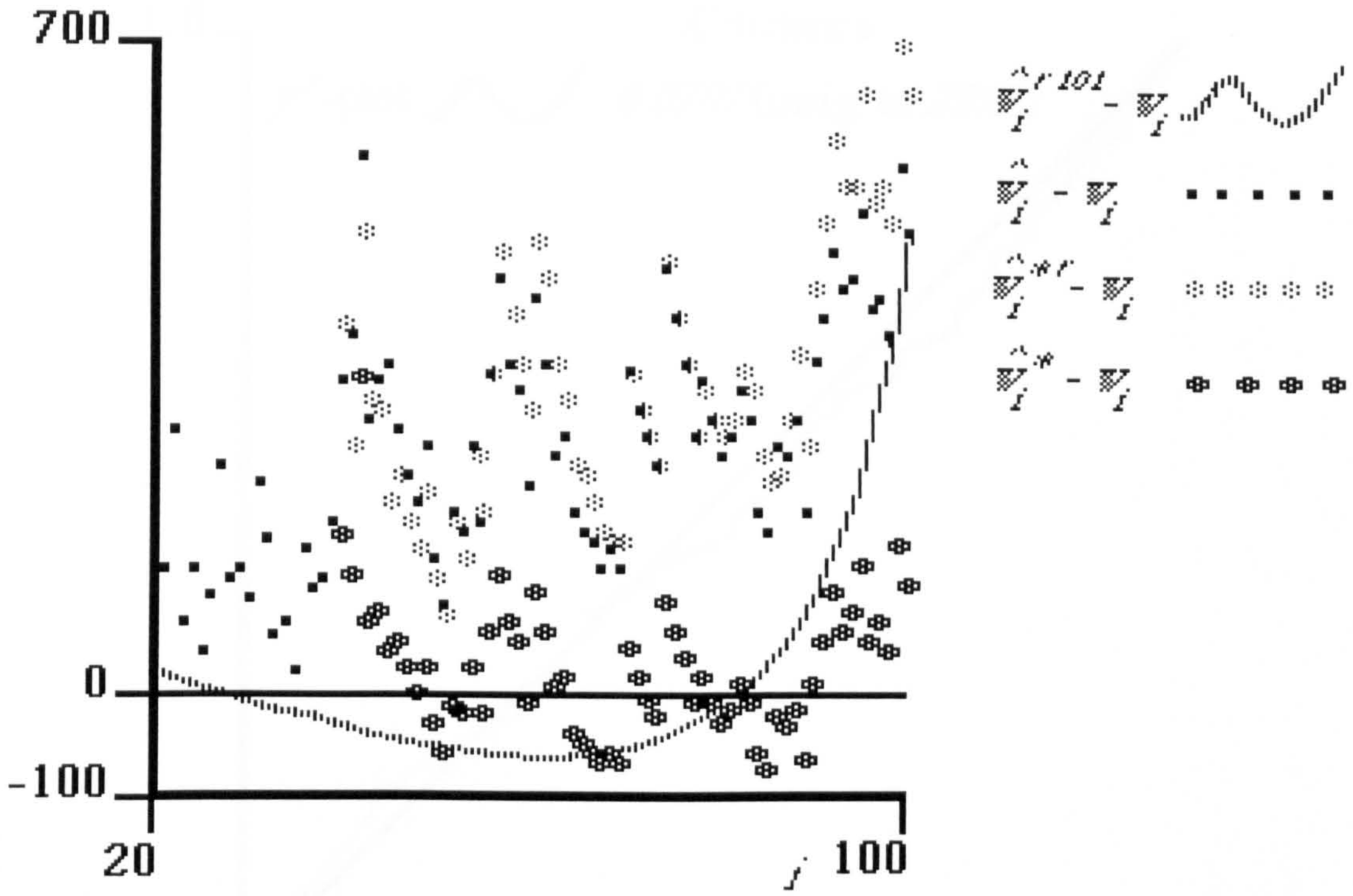


Figure 4.1-7. Progressive deviations from the truth, according to the medians, of the raw one-step-ahead predictions, the retrodictions for recalibration of T_{101}, and both recalibrated prediction systems for the *JM* model and data set *75* generated by the *LV* model.

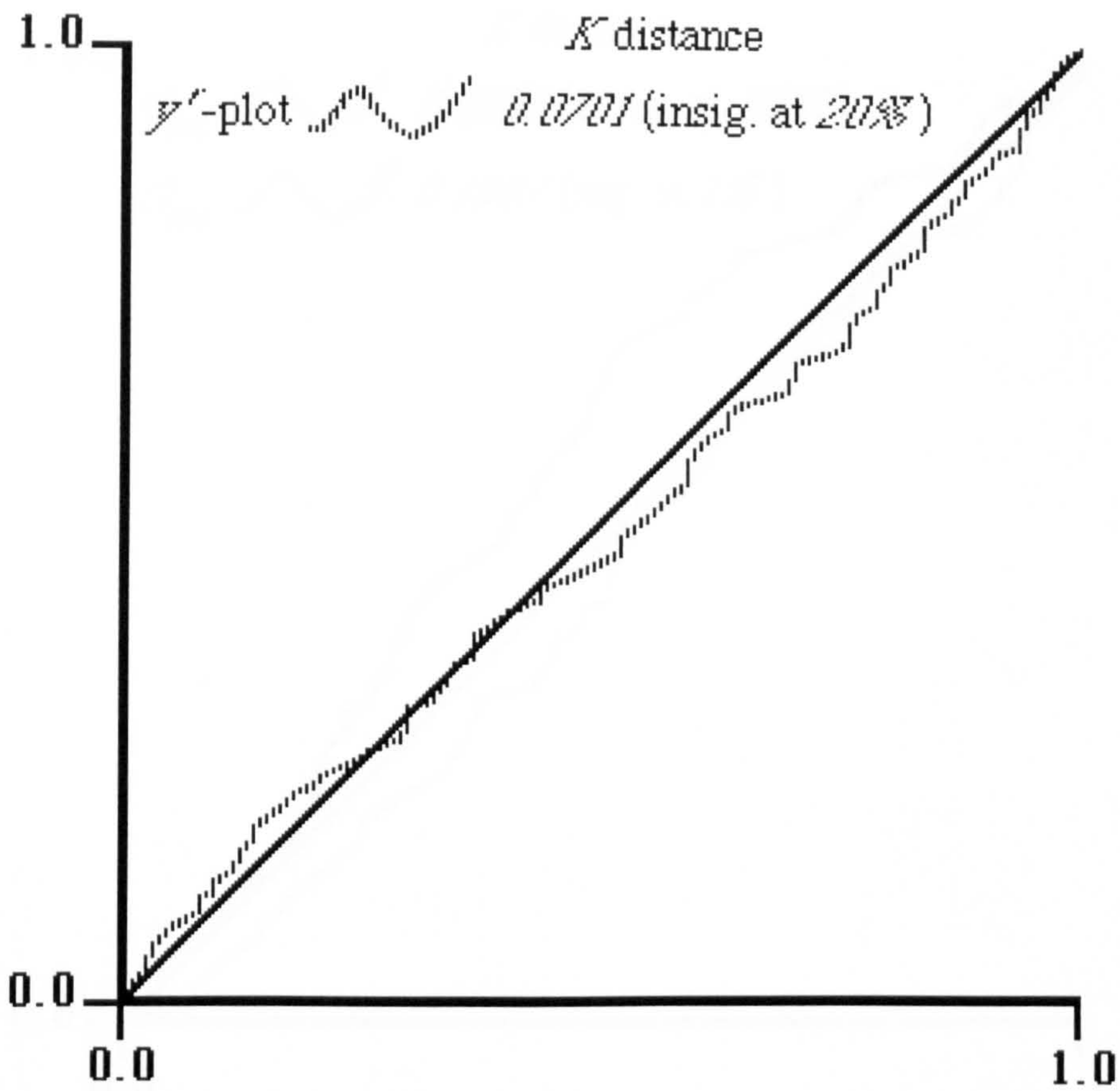


Figure 4.1-8. y^r-plot of retrodictions of T_{20}, ..., T_{100} made for recalibration of T_{101} from the raw *JM* model and data set *75* generated by the *LV* model.

It is interesting to observe (see figure 4.1-8) that the y^r-plot of the retrodictions made for recalibration of T_{101}, does not reveal the apparent non-stationarity in the error in the retrodictions, in fact this plot is insignificant at the *20%* level.

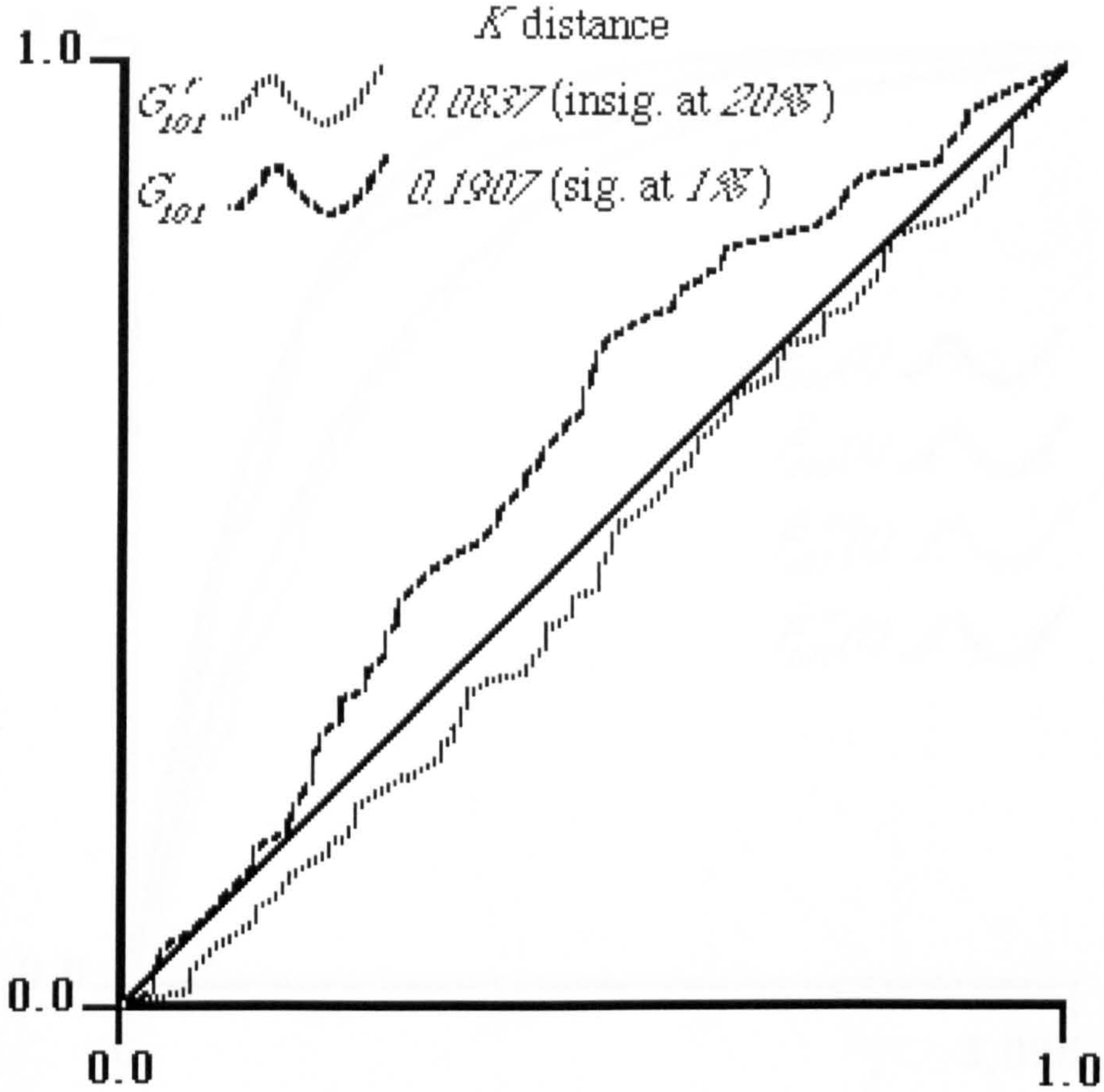


Figure 4.1-9. u^r-plot of retrodictions of T_{20}, ..., T_{100} (G^r_{101}) and u-plot of one-step-ahead predictions of T_{20}, ..., T_{100} (G_{101}) from the raw *JM* model and data set *73* generated by the *DU* model.

In the example just shown the u^r-plot for the retrodictions indicated significant bias and the adjustment made resulted in *worse* predictions than the raw. This was an extreme case. As stated previously, more commonly the u^r-plot for the retrodictions indicated no bias. Application of the raw *JM* model (see the estimated parameters in table 4.1-2) to the data generated by the *DU* model shown in table 4.1-1 is an example where the u^r-plot for the retrodictions indicated no bias. Figure 4.1-9 shows the two u-plots which are used in the retrodictive and predictive methods of recalibration of the one-step-ahead prediction of T_{101} from the raw *JM* model. Again these plots indicate that the series of one-step-ahead predictions are highly optimistic (the u-plot is significant at the *1%* level) and in this case, on average, there is no bias in the retrodictions (the u^r-plot is insignificant at the *20%* level).

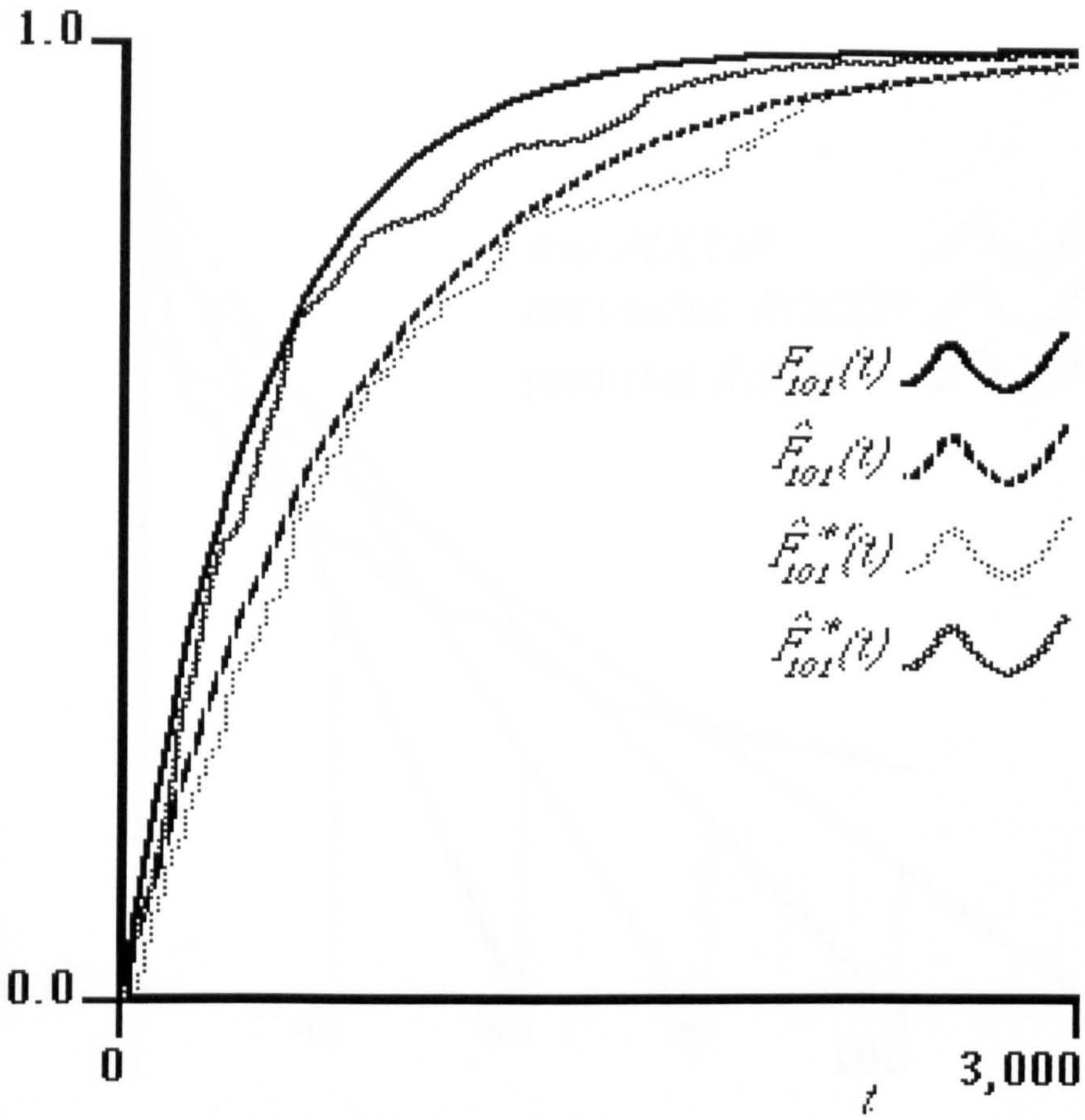


Figure 4.1-10. True and raw *cdf*s of T_{101} together with *cdf*s resulting from retrodictive and predictive methods of recalibration based on the *u*-plots in figure 4.1-9 for the *JM* model and data set *73* generated by the *DU* model.

From figure 4.1-10, which shows the true, raw and recalibrated *cdf*s resulting from both methods, it can be seen how the retrodictive method makes little adjustment for the error in the raw predictive *cdf* while, in contrast, the predictive method makes an appropriate adjustment for the optimism in the raw *cdf*.

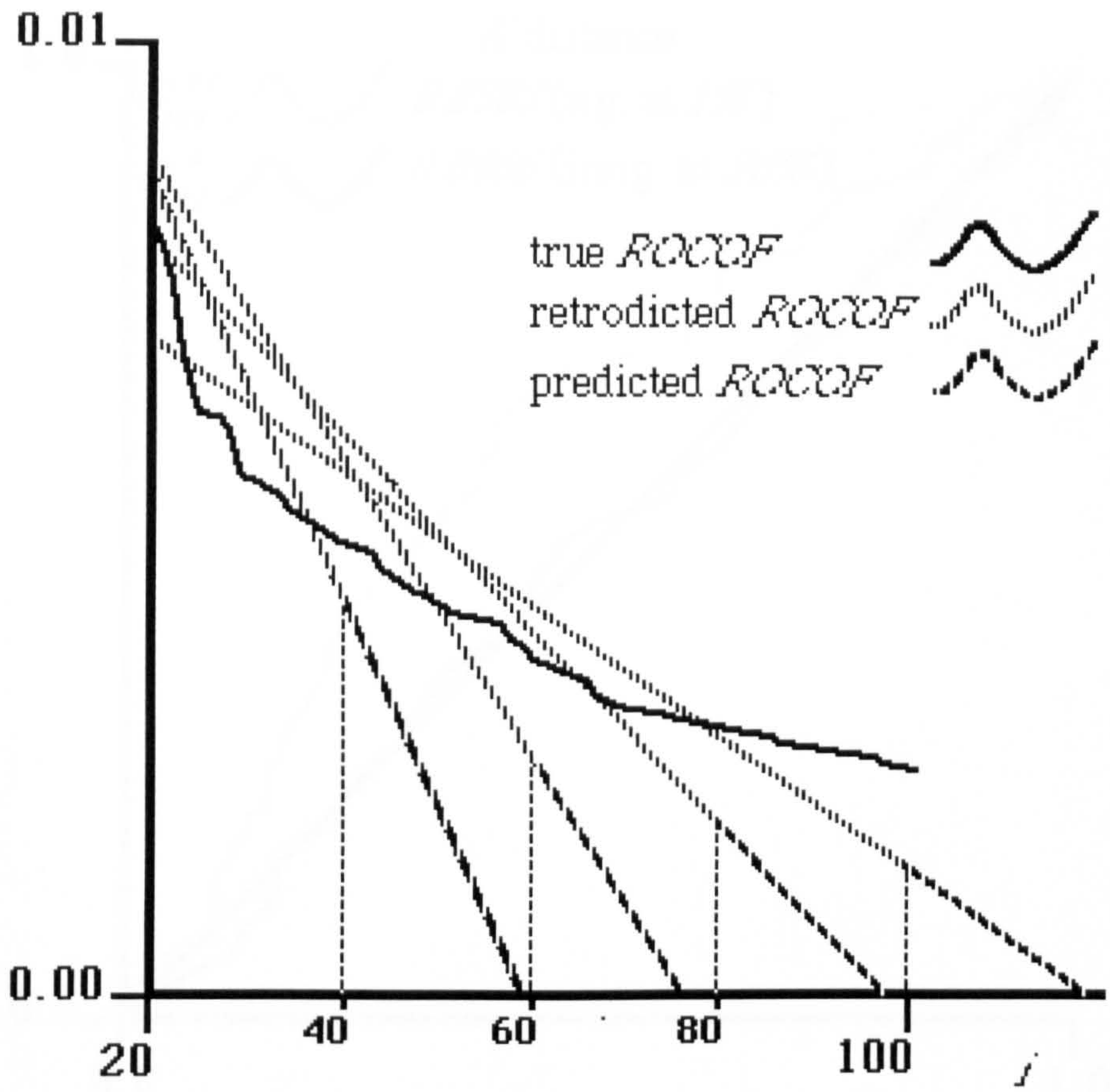


Figure 4.1-11. Actual rate from the *DU* model (the data in table 4.1-1) and estimated rate at stage *i*, for $i = 40, 60, 80$ and *100* from the raw *JM* model (the parameters in table 4.1-2).

Figure 4.1-11 shows the true rate from the *DU* model for the data of table 4.1-1 and the estimated rate, at a number of prediction stages, *i*, from the *JM* model (from the estimates shown in table 4.1-2). So, at each prediction stage, *i*, this indicates the nature of the errors in the retrodictions used for recalibration of T_i, and the nature of the error in the one-step-ahead prediction of T_i. It can be seen how the one-step-ahead rate *predictions* are fairly consistently optimistic while the *retrodictions* will have non-stationary errors with only optimism occurring as *i* is approached (and prior to this the retrodictions are mostly pessimistic). This will clearly result in insufficient adjustment for optimism in the raw one-step-ahead predictions for the retrodictive method throughout the data.

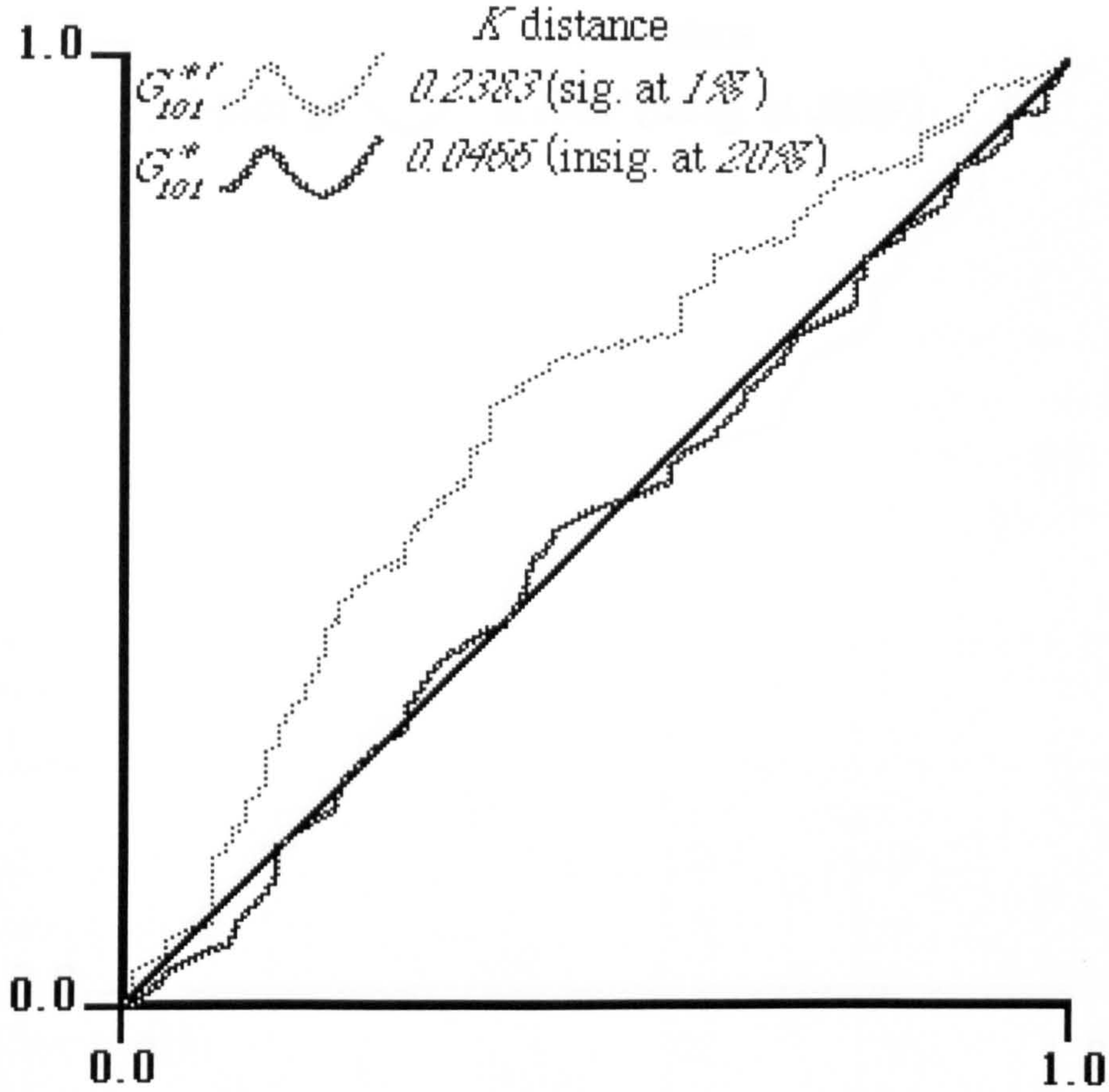


Figure 4.1-12. u^{*r}-plot of retrodictive recalibrated predictions of T_{40}, ..., T_{100} (G^{*r}_{101}) and u^{*}-plot of predictive recalibrated predictions of T_{40}, ..., T_{100} (G^{*}_{101}) for the *JM* model and data set *73* generated by the *DU* model.

Figure 4.1-12 shows the *u*-plots for the sequence of recalibrated predictions themselves. This confirms that the retrodictive recalibrated predictions remain optimistic (the u^{*r}-plot is significant at the *1%* level) while the predictive recalibration method has eliminated this optimism in the raw predictions (the u^{*}-plot is insignificant at the *20%* level).

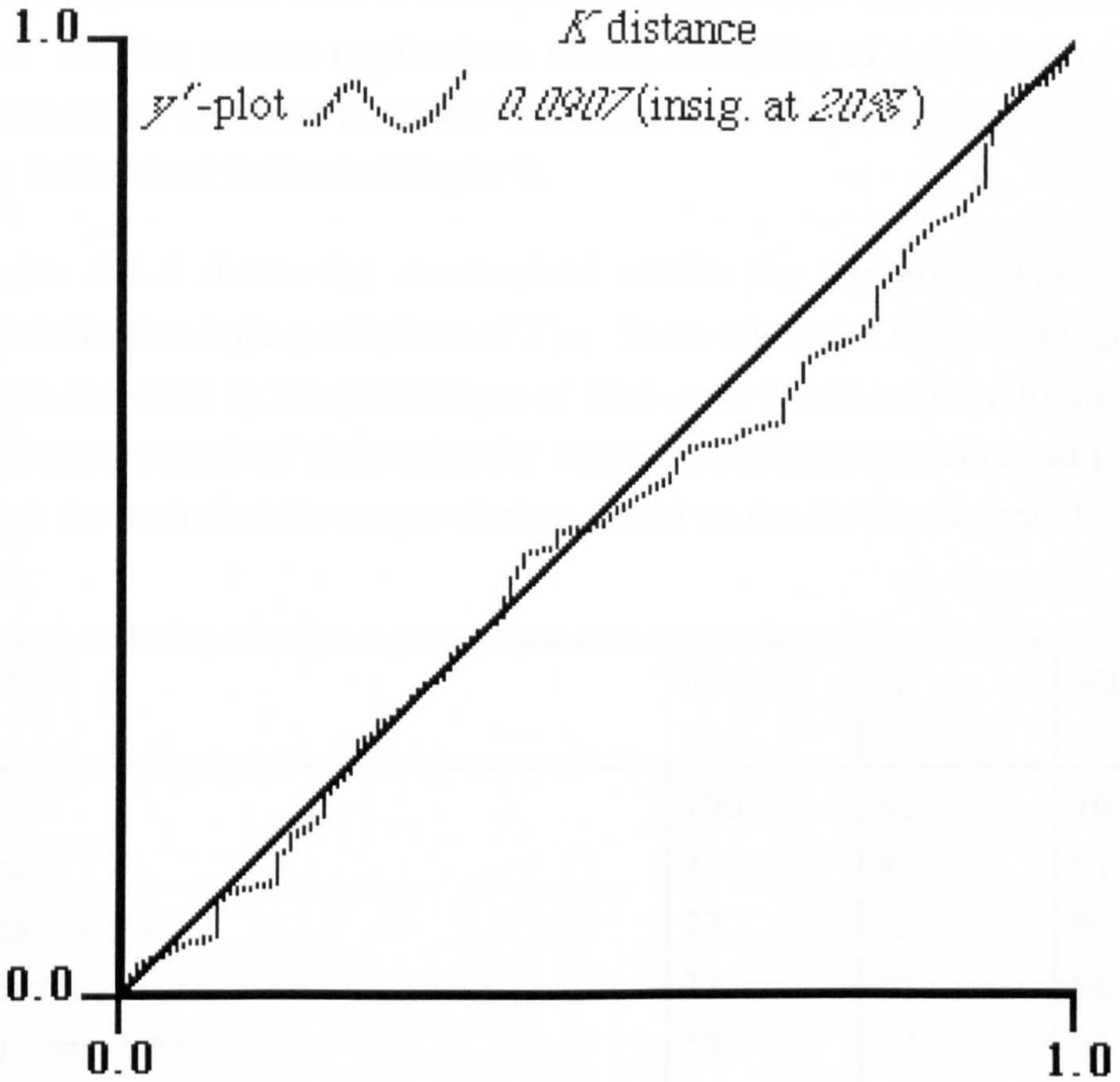


Figure 4.1-13. y^r-plot of retrodictions of T_{20}, ..., T_{100} made for recalibration of T_{101} from the raw *JM* model and data set *73* generated by the *DU* model.

It is interesting to observe (see figure 4.1-13) that, as for the previous example, the y^r-plot of the retrodictions for recalibration of T_{101} does not indicate that there is non-stationarity in the errors in the retrodictions since it is insignificant at the *20%* level.

In conclusion, it seems that the retrodictive method provides inaccurate estimates of the error in the raw one-step-ahead predictions and further, that the errors in the retrodictions used in the u^r-plot for this method of recalibration are often non-stationary, although the y^r-plots are not a very good indication of this non-stationarity.

It is clear that the inefficiency of the retrodictive method is due to such non-stationarity in the errors in the retrodictions. It is perhaps not surprising that these errors are non-stationary since at each stage the predictions are different kinds of predictions (i.e., one-step-behind, 2-steps-behind and so on for the retrodictions), and, more importantly, different in kind from the prediction which is being recalibrated (which is a one-step-ahead prediction). It seems that for recalibration of a particular type of

prediction, the predictions used in the *u*-plot for recalibration must be the same type of prediction. This has serious implications on the feasibility of using the recalibration technique to adjust for errors in predictions further into the future than one-step-ahead. This issue is discussed further in Chapter 9.

Table 4.1-5 shows the summarised results for the predictive method of recalibration for the single predictions of T_{101}. Some criteria of interest are listed in the first column, followed by the percentages of total cases which conform to these criteria and then the percentages of *these* cases for which the predictive recalibrated predictions are closer to the truth than the raw predictions, based on the *K* distances and the medians.

CRITERION *c*	CASES %	K %	MEDIAN %
All cases	100	61	70
u significant	43	89	91
u significant at *1%*	27	92	94
y insignificant	77	56	66
y insignificant at *20%*	55	52	64
u significant, *y* insignificant	33	88	90

Table 4.1-5. Summary of performance of predictive recalibrated predictions of T_{101} compared with raw predictions; %s shown are the proportion of cases which are applicable for each criterion, *c*, and for which the recalibrated predictions are better than the raw. Unless otherwise listed significance levels for *u*- and *y*-plots are *5%*.

Comparing table 4.1-5 with the equivalent results for the retrodictive method in table 4.1-3 it can be seen that the predictive method is much more effective than the retrodictive method. Now, for all cases, the percentage for which the predictive recalibrated predictions are closer to the truth than the raw are *61%* according to the *K* distance and *70%* according to the medians, while the equivalent percentages for the retrodictive method were *20%* and *44%*, respectively. Nearly half of the *u*-plots are significant at the *5%* level indicating that there are often errors in the one-step-ahead predictions while for the retrodictive method only *5%* of the u^r-plots were significant at the *5%* level confirming that the retrodictive method and the predictive method often give very different estimates of the error in the raw predictions. A similar pattern to the retrodictive method is seen for the predictive method when good *u*-plots are omitted from the comparison, with the *K* distance and median percentages levelling out. For significantly bad *u*-plots in about *90%* of cases the predictive recalibrated predictions are

closer to the truth than the raw. Limiting the analysis to good *y*-plots only, again shows no improvement in the results, in fact, surprisingly, the percentages become marginally *worse*.

Table 4.1-6 shows the results when all the predictive recalibrated predictions of T_{40}, ..., T_{101} are considered. The criteria, *c*, listed in the first column have been extended to include the *u*-plots of the sequence of predictive recalibrated predictions themselves, u^*, in order to assess whether this method eliminates bias from the raw predictions. The percentage of the total cases which fit into each criterion are again shown. The remaining percentages are, as in table 4.1-4, those proportion of *these* cases for which the predictive recalibrated *cdf*s are closer to the true *cdf*s than the raw, for the *majority* of each of the *sequence* of predictions of T_{40}, ..., T_{101} for each case.

CRITERION *c*	CASES %	K %	MEDIAN %
All cases	100	38	61
u significant	43	70	86
u significant at *1%*	27	82	93
y insignificant	77	38	60
y insignificant at *20%*	55	38	60
u significant, *y* insignificant	33	71	85
u^* insignificant	91	39	61
u^* insignificant at *20%*	74	37	59
u significant, u^* insignificant	39	72	86
u significant at *1%*, u^* insignificant	24	82	93
y insignificant, u^* insignificant	75	39	60
u significant, *y* insignificant, u^* insignificant	31	72	85
u^* better than *u*	68	51	72
u significant, u^* better than *u*	40	73	87
u significant at *1%*, u^* better than *u*	26	83	93
y insignificant, u^* better than *u*	56	49	69
u significant, *y* insignificant, u^* better than *u*	31	73	85

Table 4.1-6. Summary of performance of predictive recalibrated predictions of T_{40}, ..., T_{101}, compared with raw predictions; %s shown are the proportion of cases which are applicable for each criterion, *c*, and for which the recalibrated predictions are better than the raw. Unless otherwise listed significance levels for *u*- and *y*-plots are *5%*.

Comparing table 4.1-6 with the equivalent results for the retrodictive method in table 4.1-4 it can be seen that the predictive recalibration method is much better than the retrodictive method. Notice, particularly, that for the predictive method *91%* of the u^*-plots are insignificant at the *5%* level and *68%* of cases result in u^*-plots which are better than their corresponding u-plots, whereas these percentages for the retrodictive method are *55%* and *16%*, respectively.

Comparing table 4.1-6 with 4.1-5 it can be seen that the performance of the predictive recalibration method over the whole range of predictions is worse than for the single prediction of T_{101}. The difference in performance for all cases according to the K distance and median predictions is larger than before indicating that early u-plots are not showing much bias. Again these figures level out as good u-plots are omitted from the analysis, becoming close to *90%*, but again percentages for the K distance and medians are more disparate than for the single prediction of T_{101}. No improvement is seen when the comparison is limited to good y-plots only or to good u^*-plots only. Some improvement is seen when limiting the comparison to those cases for which the u^*-plot is better than the u-plot but not as much improvement as when limiting the comparison to bad u-plots only. Notice that most cases with significant u-plots at the *5%* level result in improved u^*-plots indicating that the predictive recalibration method usually reduces the bias in the raw predictions.

To summarise, then, the predictive recalibration method often results in improved predictions and the overriding factor affecting improvement is whether the u-plot is bad or not. Improvement is given in about *90%* of cases were the u-plot is bad while in those cases where the u-plot is good the recalibrated and raw predictions can only be marginally different.

It is interesting to observe that, given that the u-plot is bad, limiting analysis to good y-plots does not seem to eliminate those cases where non-stationarity in the prediction errors causes the recalibrated predictions to be worse than the raw predictions. On the other hand it should be noted that significant non-stationarity in the prediction errors does not necessarily mean that recalibration will not give *improvement* although it might be expected that the resulting recalibrated predictions would *still* have bias even if they are closer to the truth than the raw predictions.

Having established that the predictive recalibration method is effective we now need to investigate the *PLR* performance of the various recalibrated predictions. The *PLR* was evaluated for the recalibrated versus the raw predictions over the whole range of predictions, as defined in *(4.1.1)* (with $p = 40$ and $q = 101$). As previously reported the recalibrated *cdf*s were closer to the raw for the majority of this prediction sequence in

38% and *61%* of cases according to the *K* distance and medians, respectively. Yet, *only 0.2% of cases* gave *PLR*s that were greater than *1. It is clear that the PLR is a poor judge of the efficiency of the recalibration technique.*

More insight can be gained by investigating the *PLR* behaviour for an example. For this purpose sample number *45* generated by the *L* model, with raw predictions from the *KL* model, was chosen. This case was chosen since all the recalibrated predictions (over $i = 40, ..., 101$) were better than the raw according to both criteria for measuring closeness to the true *cdf*. Table 4.1-7 shows the $g_i(u_i)$ and the resulting *PLR* as the predictions progress.

i	$g_i(u_i)$	$PLR_{40\,i}^{*raw}$	i	$g_i(u_i)$	$PLR_{40\,i}^{*raw}$
40	5.23603E-01	5.23603E-01	70	4.34735E-01	7.07512E-05
41	1.05105E+00	5.50333E-01	71	6.06569E-01	4.29155E-05
42	2.06170E+00	1.13462E+00	72	1.33883E+00	5.74565E-05
43	9.17022E-01	1.04047E+00	73	6.86992E-01	3.94722E-05
44	3.07457E+00	3.19901E+00	74	3.25658E+00	1.28544E-04
45	6.85467E-01	2.19281E+00	75	6.76687E-01	8.69843E-05
46	2.75118E-01	6.03282E-01	76	1.70089E+00	1.47951E-04
47	1.08422E+00	6.54091E-01	77	1.46506E+00	2.16757E-04
48	1.68435E+00	1.10172E+00	78	1.21415E+00	2.63175E-04
49	6.61293E-01	7.28558E-01	79	2.16309E+00	5.69272E-04
50	1.87501E-01	1.36605E-01	80	3.67966E+00	2.09473E-03
51	2.24539E+00	3.06732E-01	81	5.16615E-01	1.08217E-03
52	2.91008E-01	8.92616E-02	82	2.81031E-01	3.04122E-04
53	4.21032E-01	3.75820E-02	83	5.77873E-01	1.75744E-04
54	4.20720E-01	1.58115E-02	84	3.39638E+00	5.96894E-04
55	3.81595E-01	6.03359E-03	85	4.20541E+00	2.51018E-03
56	1.96638E+00	1.18643E-02	86	1.28611E-01	3.22837E-04
57	6.11061E-01	7.24983E-03	87	2.97652E-01	9.60931E-05
58	6.39222E-01	4.63425E-03	88	7.08922E-01	6.81225E-05
59	2.53226E+00	1.17351E-02	89	3.31473E+00	2.25808E-04
60	3.25597E+00	3.82092E-02	90	6.28990E+00	1.42031E-03
61	2.51928E-01	9.62597E-03	91	4.79540E-01	6.81095E-04
62	7.12889E-01	6.86225E-03	92	6.31262E-01	4.29949E-04
63	6.40058E-01	4.39224E-03	93	6.12045E-01	2.63148E-04
64	5.09638E-01	2.23845E-03	94	2.39503E+00	6.30248E-04
65	5.13134E-01	1.14863E-03	95	1.33488E-01	8.41306E-05
66	5.61443E-01	6.44888E-04	96	1.92178E-01	1.61680E-05
67	9.64067E-01	6.21715E-04	97	4.83771E-01	7.82163E-06
68	5.62001E-01	3.49404E-04	98	4.40768E-01	3.44753E-06
69	4.65780E-01	1.62746E-04	99	6.48123E-01	2.23442E-06
			100	6.33741E-01	1.41604E-06
			101	2.77270E+00	3.92627E-06

Table 4.1-7. Successive values of $g_i(u_i)$ and *PLR* for the recalibrated versus the raw predictions, when the *KL* model is applied to the data set *45* generated by the *L* model.

It can be seen that the $g_i(u_i)$ are less than *1* at most stages and as a result the *PLR* generally (with some variation) decreases with *i* until at the end of the data set its value is very much less than *1*.

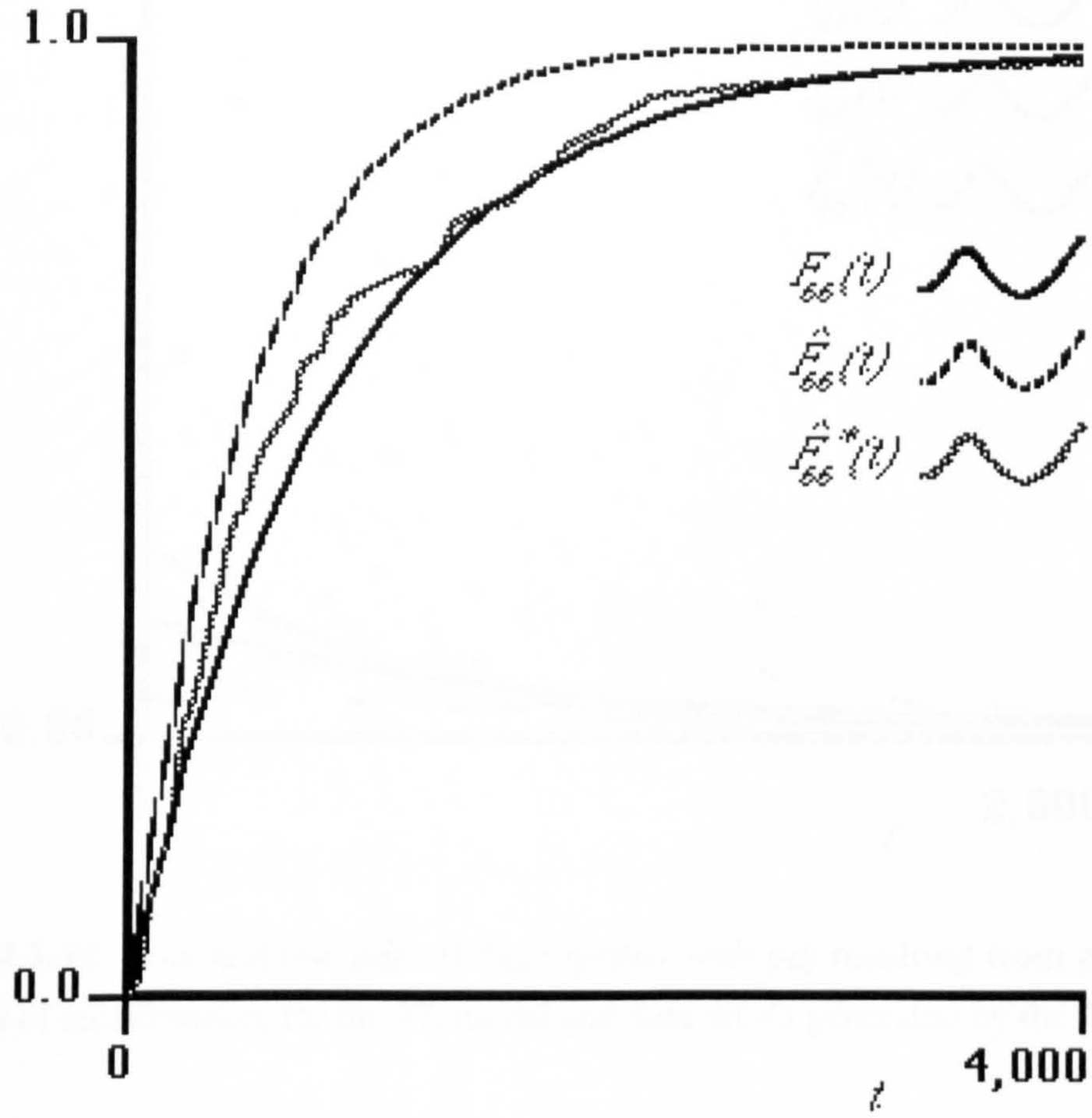


Figure 4.1-14. True and raw *cdf*s of T_{66} together with *cdf* resulting from predictive method of recalibration, for the *KL* model and data set *45* generated by the *L* model.

Figure 4.1-14 shows the true, raw and recalibrated predicted *cdf*s for T_{66}. The recalibrated *cdf* is indeed closer to the true *cdf* than the raw *cdf*, as our criteria reported.

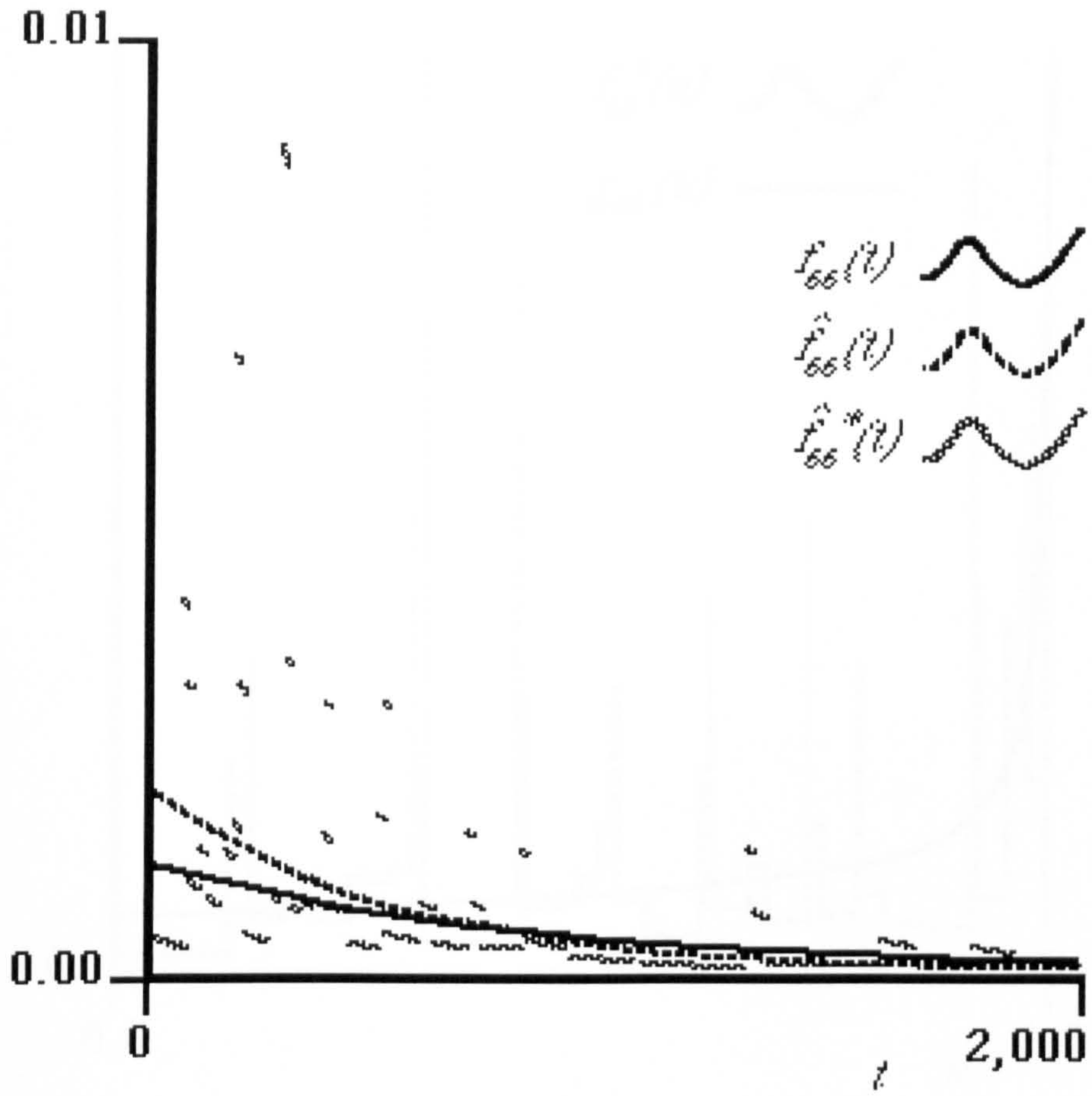


Figure 4.1-15. True and raw *pdf*s of T_{66} together with *pdf* resulting from predictive method of recalibration, for the *KL* model and data set *45* generated by the *L* model.

Figure 4.1-15 shows the corresponding predictive *pdf*s. The recalibrated *pdf* is very discontinuous over t (as distinct from noise in the predictions over i). It cannot be said that it is closer to the true *pdf* than the raw *pdf* and so it would be expected that the *PLR* would favour the raw *pdf* over the recalibrated. This is in spite of the good performance of the recalibrated *cdf*.

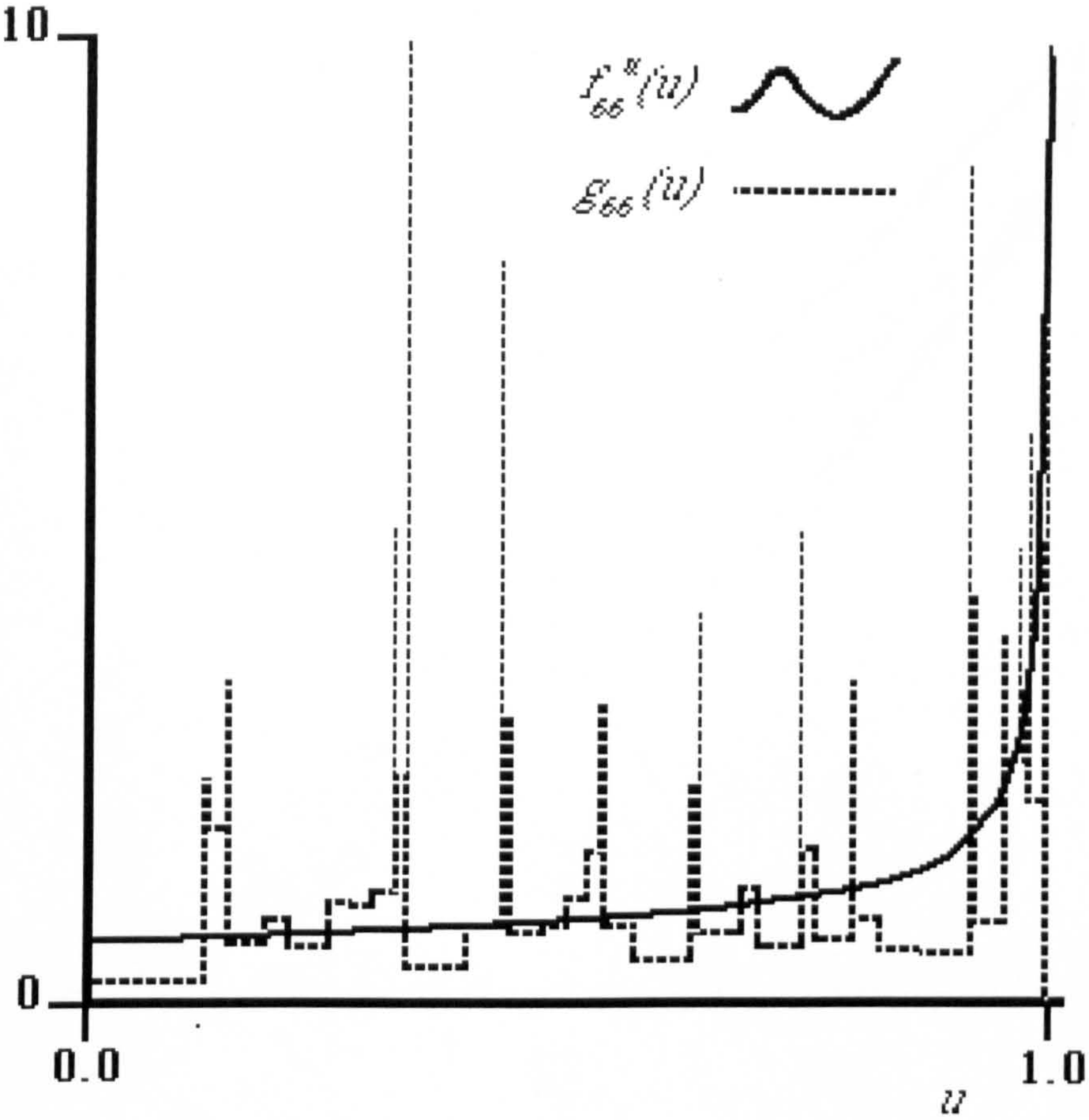


Figure 4.1-16. True and estimated *pdf*s of U_{66}, $f^u_{66}(u)$ and $g_{66}(u)$, for the *KL* model and data set *45* generated by the *L* model.

From *(4.2)* we have, $F^u_i(\hat{F}_i(T_i)) = F_i(T_i)$ and so, since $U_i = \hat{F}_i(T_i)$, differentiating with respect to T_i, gives

$$f^u_i(U_i) = \frac{f_i(\hat{F}^{-1}_i(U_i))}{\hat{f}_i(\hat{F}^{-1}_i(U_i))} \qquad \ldots\ldots (4.1.8)$$

Figure 4.1-16 shows the true *pdf* of U_{66}, $f^u_{66}(u)$ together with the estimated *pdf*, i.e., $g_{66}(u)$. It can be seen how the estimate is very discontinuous whereas the true *pdf* is smooth.

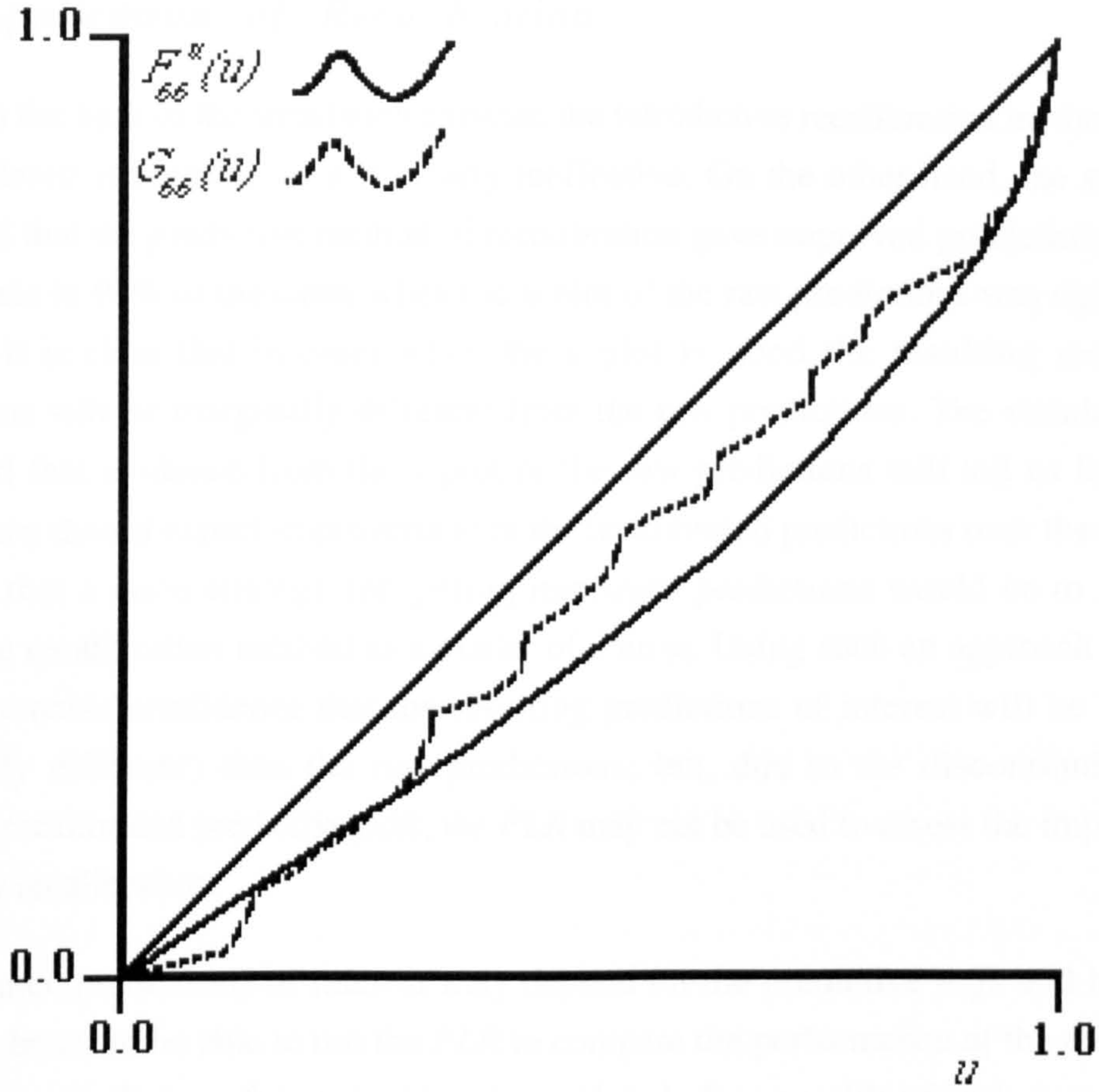


Figure 4.1-17. True and estimated *cdf*s of U_{66}, $F_{66}^{u}(u)$ and $G_{66}(u)$, for the *KL* model and data set *45* generated by the *L* model.

Figure 4.1-17 shows the corresponding true and estimated *cdf*s $F_{66}^{u}(u)$ and $G_{66}(u)$. Apart from the lack of smoothness in G_{66}, they are quite close.

It is the step-function nature of the predictive *cdf* of the *U* (see figures 4-1 and 4.1-17) which results in the discontinuity in the recalibrated predictive *pdf* and the *pdf* of *U* (see figures 4.1-15 and 4.1-16). It is clear that the bad results according to the *PLR* are due to the discontinuities in the predictive *pdf*s even though estimates of the *cdf*s are quite good. If the predictive errors are truly stationary, then as i gets larger we would expect our G_i function to approach the true *cdf* of the U_i but this does not necessarily mean that the g_i function will approach the true pdf of the U_i.

In fact it seems that this *discontinuity* in the recalibrated predictive *pdf*s is enough to outweigh any decrease in bias bought by the recalibration method in most instances and that the *PLR* may only be expected to report favourably about the recalibrated predictions in situations where the initial bias in the raw predictions is extreme.

4.2 Application of Recalibration

In the light of the simulation exercise the retrodictive recalibration method will not be considered any further as it is clearly ineffective. On the other hand, the simulation suggested that the predictive method of recalibration gave improved predictions over the raw models in *90%* of the cases when the *u*-plot of the raw predictions was significantly bad and it is clear that in cases when the *u*-plot is good the resulting recalibrated predictions will be marginally different from the raw predictions. The simulation also suggested that evidence from the *y*-plot of the raw predictions will tell us little about whether we should expect improvement in the recalibrated predictions over the raw. This suggests that a good strategy for getting improved predictions would be to apply the predictive recalibration method as a matter of course. Using such an approach we could have reasonable confidence that the resulting predictions of interest will be better (or marginally different) than the raw predictions, but, due to the discontinuity in the resulting recalibrated predictive *pdf*s, the *PLR* may not be used to assess the improvement gained by recalibration.

Since predictions of interest may depend on the predictive *pdf*s and because it would be better to be able to use the *PLR* to compare the performance of the recalibrated predictions a technique for *smoothing* the *u*-plots before recalibration is introduced in [Chan 1986] and [Chan and Littlewood 1986]. This technique consists of smoothing each G_i using *least squares cubic splines* before recalibration. This results in smooth recalibrated *pdf*s but clearly most *probability* predictions of interest will be altered little by smoothing. This technique obviously involves more effort than simply using the joined up *u*-plot for recalibration but this added effort is fairly negligible when compared with the effort in estimating some of the raw parametric model parameters referred to in section 2.1.

Application of this spline-recalibration technique has been shown in [Chan 1986] and [Brocklehurst et al. 1990] to frequently give dramatic improvements over the raw models, not only with respect to bias (i.e., improvement in the *u*-plot) but also on the basis of the *PLR*. Typically, when starting with a number of raw prediction systems which are in error to differing extents, the recalibration technique tends to improve these predictions and the resulting recalibrated predictions are often fairly close in accuracy.

Recent experience of applying the spline-recalibration procedure [Brocklehurst et al. 1991; Brocklehurst and Littlewood 1992] shows that there is sometimes evidence of non-stationarity in some of the raw model prediction errors on some of the data sets. Typical behaviour, for example, might be that the raw model predictions are accurate early on in the data set and optimistic later on. The presence of the early raw predictions

in the *u*-plot used for recalibration in this scenario would result in insufficient adjustment for optimism in the later predictions. It seems likely that, in the presence of such non-stationarity in the prediction errors, more recent errors in the raw predictions may give a clearer indication of the current predictive error than earlier predictions. It such circumstances it seems sensible to omit early predictions from the *u*-plot used for recalibration. It was decided, therefore, to recalibrate using fixed size moving windows, as well as on the whole sequence of predictions. In other words, for a window of size w, we use the w most recent raw one-step-ahead predictions (of T_{i-w}, ..., T_{i-1}) in order to form the *u*-plot for recalibration of the prediction of T_i. Clearly any gain in predictive accuracy by eliminating bias by recalibrating with windows (as opposed to without) may be countered by the increase in noise which is likely to be more prominent with smaller window sizes.

The subsequent analysis in Chapter 8 will be limited to the spline-recalibration technique since it is preferable to be able to use the *PLR* in order to assess the accuracy of the recalibrated prediction systems. Predictions resulting from the spline-recalibration technique applied both without windows and with some arbitrarily chosen fixed window sizes to all the raw prediction systems from the parametric models referred to in section 2.1 and the non-parametric models described later in Chapter 5, will be investigated. This results in a number of prediction systems (1 raw and several spline-recalibrated) to compare for *each* of the raw models on each of the data sets presented in Chapter 7.

5 Non-Parametric Reliability Models

As stated previously the recalibration technique often results in predictions which are fairly close in accuracy even when applied to a group of raw prediction systems which are giving quite different raw predictions. So, for example, the *JM* model, which frequently gives more optimistic raw predictions than the other 7 parametric models referred to in section 2.1, may, after recalibration, result in predictions very close to the recalibrated predictors from the other models. This suggests that it may not be necessary to apply sophisticated models since a simple model, together with the recalibration technique, may result in comparable predictions.

In this chapter some simple *non-parametric* models are described. These models do not have the stringent modelling assumptions of many of the parametric models and generally encapsulate just some of the more basic assumptions of many of the parametric models. It is felt that any resulting inaccuracy in the local distribution due to the simplicity of these models may be later eliminated by the recalibration technique.

The non-parametric models described here attempt to estimate the evolution of the rate of occurrence of failures of the system. Since these models are not as well-known as the parametric models referred to in section 2.1 they will be described in more detail.

5.1 Completely Monotone Model

The first model to be described is due to Miller and Sofer [Miller and Sofer 1985; Miller and Sofer 1986a; Miller and Sofer 1986b].

Miller [Miller 1986] observed that the rate functions of most existing software reliability models have the *complete monotonicity* property, i.e., if the rate function at time τ (the total elapsed time) is $r(\tau)$, then

$$(-1)^q \frac{d^q r(\tau)}{d^q \tau} \geq 0 \qquad \tau \geq 0,\ q = 0, 1, 2, ... \qquad .....(5.1.1)$$

This led to the formulation [Miller and Sofer 1985; Miller and Sofer 1986b] of a non-parametric approach to estimating the failure rate in which an approximation to this complete monotonicity property is used as a constraint.

As in Chapter 2, given inter-failure time data, $t_1, t_2, ..., t_{j-1}$, the objective is to obtain a one-step-ahead prediction for the next inter-failure time, T_j. The problem is first discretised by dividing the total elapsed time up to the $j\text{-}1^{th}$ failure, $\tau_{j-1} = \sum_{k=1}^{j-1} t_k$, into a

specified number, n, of equal intervals. The rates, r_k, are kept constant during each interval, k, $k = 1, ..., n$, resulting in a piece-wise constant rate function and are constrained to be a completely monotone sequence, i.e.,

$$(-1)^q \Delta^q r_k \geq 0, \qquad q = 0, 1, 2, ..., k = q+1, ..., n \qquad .....(5.1.2)$$

where $\Delta^q r_k = \Delta^{q-1} r_k - \Delta^{q-1} r_{k-1}$ and $\Delta^0 r_k = r_k$.

Crude estimates of the rates, r_k, are made from the previous data, $t_1, ..., t_{j-1}$, and the solution vector of rates, $\tilde{r}_n = \langle \tilde{r}_1, ..., \tilde{r}_n \rangle$, is found by finding the rates satisfying *(5.1.2)* up to a specified order, d, which are closest to these estimates, using the criterion of *least squares*. This leads to a quadratic programming problem. The last rate, $\tilde{r}_n$, together with the assumption of exponential inter-failure times is then used to obtain the one-step-ahead prediction for T_j, so the predictive *cdf* and *pdf* are

$$\hat{F}_j(t) = 1 - e^{-\tilde{r}_n t} \qquad .....(5.1.3)$$

and

$$\hat{f}_j(t) = \tilde{r}_n e^{-\tilde{r}_n t} \qquad .....(5.1.4)$$

respectively. As mentioned above, it is hoped that consistent inaccuracies in the predictions which may arise as a result of this simple approach of plugging in the most recently estimated rate (which in the presence of reliability growth in the data may be expected to result in marginal pessimism in the predictions), and the assumption of exponentiality, may be eliminated later by the recalibration technique described previously in Chapter 4.

Investigation of the performance of this completely monotone non-parametric model, which we shall refer to as *CM*, has been carried out on simulated data in [Miller and Sofer 1985] and [Miller and Sofer 1986b] and on real data in [Chan 1986]. The major finding of this work was that this model is a good candidate for one-step-ahead prediction when compared with the other more conventional parametric models. It was also found that application of this model often gave rise to quadratic programming problems for which the constraint matrix can be ill-conditioned [Miller and Sofer 1986a] and can become more so for higher values of d. The investigation in [Chan 1986] suggested that, in any case, there was little to be gained in predictive accuracy by considering differences which are higher than second order (i.e., $d > 2$). In the later analysis in Chapter 8 we shall limit our investigation to application of this model with $d = 1, 2$ and 3 and the performance of the raw and spline-recalibrated versions of these

predictors will be examined. The software used to apply this model is coded in FORTRAN and was executing on a Sun 3/80.

The remaining non-parametric models to be described in this chapter are similar to the *CM* model except that the solutions are constrained in order that the sequence of rates captures the *trend* in the failure data. Another, more trivial difference, is that the intervals for which the rates are kept piece-wise constant are precisely the inter-failure times.

The motivation for deriving a model which focuses on capturing the trend in the failure data, while giving little attention to local behaviour, initially came from the success of the recalibration technique. It was felt that, providing the trend was captured, any local inaccuracies could be eliminated later via recalibration. It should be noted, though, that while the sequence of retrodictions (as defined in Chapter 4) may capture the trend in the data, this does not necessarily imply that the one-step-ahead predictions themselves will have captured the trend; it is the latter which is required in order that recalibration will result in accurate predictions while the following models presented here attempt to attain the former situation.

5.2 Capturing the Trend using the y-plot

In [Chan 1986] the theory of a non-parametric model which concentrates on capturing the trend in the failure data is presented although its performance is not investigated. As before, given the previous inter-failure time data, $t_{j-1} = \langle t_1, ..., t_{j-1} \rangle$, the objective is to obtain a one-step-ahead prediction for the next inter-failure time, T_j. Let r_k be the rate during the k^{th} inter-failure time, $k = 1, ..., j-1$. The solution for the vector of rates, $\tilde{r}_{j-1} = \langle \tilde{r}_1, ..., \tilde{r}_{j-1} \rangle$, is found as follows.

Only the first 3 difference constraints from *(5.1.2)* will be applied, i.e.,

$$(-1)^q \Delta^q r_k \geq 0, \qquad q = 0, 1, 2, \; k = q+1, ..., j-1 \qquad \text{.....(5.2.1)}$$

The first of these constraints, with $q = 0$, is the assumption that the resulting rates are non-negative, an obvious requirement. The second, with $q = 1$, is the assumption that as debugging proceeds, and faults are progressively removed, the failure rate will decrease, which corresponds to the usual reliability growth assumption. The third, with $q = 2$, is the assumption that faults removed which manifest as failures later in the debugging process make less contribution to the change in the failure rate than those which have manifest as failures earlier in the debugging process: a law of diminishing returns. These assumptions are equivalent to those made by many existing parametric reliability growth models and, as mentioned earlier, there is little evidence that

considering any higher order difference equations will result in any gain in predictive accuracy.

Let $m_k = E[T_k]$, $k = 1, ..., j-1$. Then, assuming that the T_k are exponentially distributed, the requirement that the trend in the data is captured is that the $\frac{T_k}{m_k}$, $k = 1, ..., j-1$, are trend free. Since $m_k = \frac{1}{r_k}$, this is equivalent to the requirement that the sequence $r_k T_k$ be trend free.

One test against trend is to construct a y-plot of the $\{r_k t_k; k = 1, ..., j-1\}$ as described in section 3.2, with

$$x_k = r_k t_k \qquad k = 1, ..., j-1 \qquad \text{.....(5.2.2)}$$

If $\{r_k T_k; k = 1, ..., j-1\}$ *are* identically distributed then this y-plot should be close to the $45°$ line. The objective is thus to estimate the rates, r_{j-1}, so that the y-plot is as close to the $45°$ line as possible subject to the constraints given in *(5.2.1)*. This can be done by minimising the K distance of the y-plot.

Finally, application of an additional scaling constraint,

$$\sum_{k=1}^{j-1} r_k t_k = j-1 \qquad \text{.....(5.2.3)}$$

leads to the following linear programming problem (see Appendix A.1.1 for the derivation).

$$\underset{r_{j-1}}{Minimise}\ [O_y(r_{j-1}, t_{j-1})] \text{ subject to}$$

$$O_y(r_{j-1}, t_{j-1}) \geq 0 \qquad \text{.....(5.2.4)}$$

$$r_k \geq 0 \qquad k = 1, ..., j-1 \qquad \text{.....(5.2.5)}$$

$$r_k - r_{k-1} \leq 0 \qquad k = 2, ..., j-1 \qquad \text{.....(5.2.6)}$$

$$r_k - 2r_{k-1} + r_{k-2} \geq 0 \qquad k = 3, ..., j-1 \qquad \text{.....(5.2.7)}$$

$$\sum_{s=1}^{k} r_s t_s + O_y(r_{j-1}, t_{j-1}) \geq k \qquad k = 1, ..., j-1 \qquad \text{.....(5.2.8)}$$

$$\sum_{s=1}^{k} r_s t_s - O_y(r_{j-1}, t_{j-1}) \leq k - 1 \qquad k = 1, ..., j-1 \qquad \text{.....(5.2.9)}$$

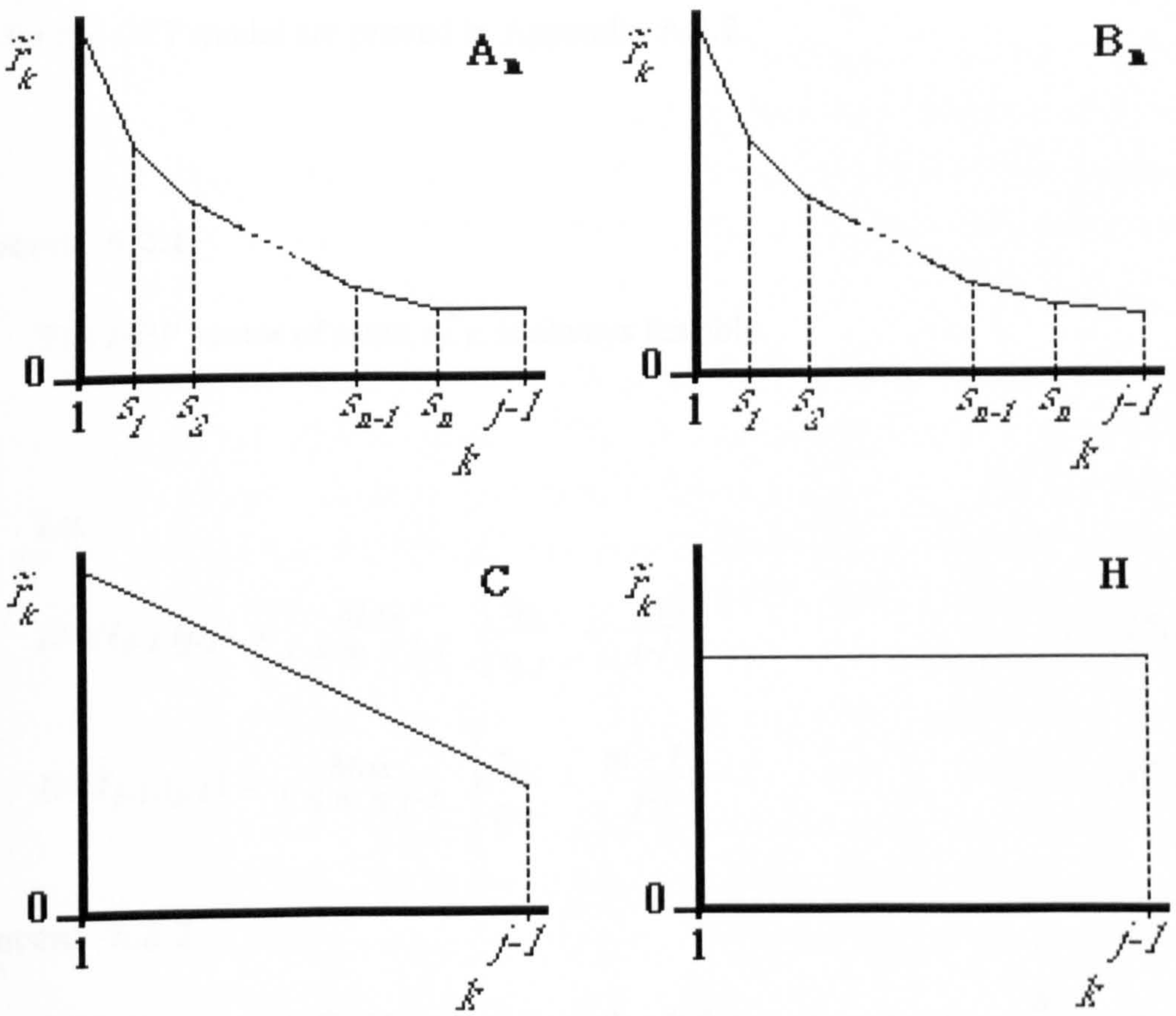
$$\sum_{k=1}^{j-1} r_k t_k = j-1 \qquad \text{.....(5.2.10)}$$

The optimum solution to this linear programming problem yields our vector of estimated rates, $\tilde{\boldsymbol{r}}_{j-1}$, and, as for the *CM* model, the most recently estimated rate, $\tilde{r}_{j-1}$, together with the assumption of exponentiality gives us our one-step-ahead prediction for T_j,

$$\hat{F}_j(t) = 1 - e^{-\tilde{r}_{j-1}t} \qquad \text{.....(5.2.11)}$$

and

$$\hat{f}_j(t) = \tilde{r}_{j-1}\, e^{-\tilde{r}_{j-1}t} \qquad \text{.....(5.2.12)}$$



A_n - n non-zero second differences, $\tilde{r}_{j-1} = \tilde{r}_{j-2} = \ldots = \tilde{r}_{s_n}$.

B_n - n non-zero second differences, first differences all non-zero.

C - second differences all zero, first differences all non-zero.

H - first and second differences all zero, $\tilde{r}_{j-1} = \tilde{r}_{j-2} = \ldots = \tilde{r}_1$.

Figure 5.2-1. Classification of optimum solutions resulting from application of the *OTY* model to a number of real data sets.

In [Brocklehurst 1989] it was observed that application of this model, which we shall call *OTY*, to some real data sets resulted in a variety of optimum solutions as shown in figure 5.2-1.

Observe that, as with the parametric models, this model will also result in optimum solutions which correspond to an *HPP* (**H** in figure 5.2-1) when no growth is exhibited in the data vector over which they are applied but again, the conditions on the data for "no growth" depend on the model in question. The conditions under which the *OTY* model result in an *HPP* optimum solution are given below.

The *HPP* has estimated rates for the successive inter-failure times, $\tilde{r}_k = \frac{j\text{-}1}{\tau_{j\text{-}1}} = r^{h}_{j\text{-}1}$, say, for all $k = 1, \ldots, j\text{-}1$, where $\tau_{j\text{-}1} = \sum_{k=1}^{j\text{-}1} t_k$. Let us denote this vector of rates which correspond to the *HPP* by $r^{h}_{j\text{-}1}$. Then the following conditions on the data relating to the *HPP* for the *OTY* model are proved in Appendix A.1.2.

Theorem 5.2.1

The *HPP* vector of rates, $r^{h}_{j\text{-}1}$, is always feasible.

Let

$$D^{+}(1_{j\text{-}1},t_{j\text{-}1}) = \max_{1 \le m \le j\text{-}1} \left| \frac{\tau_m}{\tau_{j\text{-}1}} - \frac{m}{j\text{-}1} \right| \qquad \text{.....(5.2.13)}$$

$$D^{-}(1_{j\text{-}1},t_{j\text{-}1}) = \max_{1 \le m \le j\text{-}1} \left| \frac{\tau_m}{\tau_{j\text{-}1}} - \frac{m-1}{j\text{-}1} \right| \qquad \text{.....(5.2.14)}$$

Theorem 5.2.2

If $D^{-}(1_{j\text{-}1},t_{j\text{-}1}) \ge D^{+}(1_{j\text{-}1},t_{j\text{-}1})$ then the *HPP* vector of rates, $r^{h}_{j\text{-}1}$, is feasible, and optimal and additionally if $D^{-}(1_{j\text{-}1},t_{j\text{-}1}) = \left| \frac{\tau_n}{\tau_{j\text{-}1}} - \frac{n-1}{j\text{-}1} \right|$ and $n \neq j\text{-}1$ and $t_k \neq 0$ for all $k = 1, \ldots, j\text{-}1$, then $r^{h}_{j\text{-}1}$ is uniquely optimal.

Theorem 5.2.3

If $D^{-}(1_{j-1},t_{j-1}) < D^{+}(1_{j-1},t_{j-1})$ then the *HPP* vector of rates, r^{h}_{j-1}, is feasible, but not optimal.

Thus we can see that under certain conditions on the inter-failure time data we can conclude that the optimum solution is $\tilde{r}_{j-1} = r^{h}_{j-1}$ for the *OTY* model and so no optimisation need be performed. Also, it can be seen that given that the *HPP* solution is optimal it will only be in very rare circumstances that it will be non-unique.

For the remaining classes of solutions shown in figure 5.2-1 it is not possible to derive conditions on the data which will indicate which type of solution will be optimal in a particular instance or to derive a general analytical solution.

The predictive performance of this model was investigated in [Brocklehurst 1989] on some real data using the techniques for assessing predictive accuracy described in Chapter 3 and it was found that the resulting raw (or spline-recalibrated) predictions were comparable with those (raw or spline-recalibrated) from the *8* parametric models referred to in section 2.1 against which they were compared.

On examination of the problem formulation *(5.2.4)* -*(5.2.10)* it can be seen that as j increases the number of constraints ($5j-6$ in all) increases *5*-fold. This results in very computationally intensive linear programming problems as j gets large and this problem is worsened by the fact that the optimisation is being repeated successively to attain the desired series of one-step-ahead predictions. Due to such computational difficulties it was necessary to fit this model by using fixed size moving windows along the data (i.e., instead of using all the previous data, $t_1, ..., t_{j-1}$, to make predictions about T_j, the w most recently observed data points, $t_{j-w}, ..., t_{j-1}$ were used). In the later analysis in Chapter 8 the performance of the raw and spline-recalibrated predictions from this model, when applied with some arbitrarily chosen fixed size moving windows, will be investigated. The SAS/OR package executing on a VAX was used to obtain the optimum solutions to the resulting linear programming problems.

5.3 Capturing the Trend using the Laplace Statistic

In [Brocklehurst 1989] a model very similar to *OTY* was also investigated. It was identical to the *OTY* model in that the difference constraints *(5.2.1)* and the scaling constraint *(5.2.3)* were applied except that in this case the test against trend utilised was the *Laplace statistic* [Cox and Lewis 1966] of $\{r_k t_k | k=1, ..., j-1\}$,

$$L(r_{j-1},t_{j-1}) = \frac{\sum_{k=1}^{j-1} r_k t_k}{2} - \frac{\sum_{k=1}^{j-2}\sum_{s=1}^{k} r_s t_s}{j-2} \qquad \ldots\ldots(5.3.1)$$

In this case the scaling constraint *(5.2.3)* is applied not to make the problem linear in the variables (it is already linear) but, since it can be seen that there will be an infinite number of optimum solutions (including $r_{j-1} = 0$) unless such a scaling constraint is applied. This leads to the following linear programming problem (see Appendix A.2.1 for the derivation).

$$\underset{r_{j-1}}{Minimise}\ [O_l(r_{j-1},t_{j-1})] \text{ subject to}$$

$$O_l(r_{j-1},t_{j-1}) \geq 0 \qquad \ldots\ldots(5.3.2)$$

$$r_k \geq 0 \qquad k = 1, \ldots, j-1 \qquad \ldots\ldots(5.3.3)$$

$$r_k - r_{k-1} \leq 0 \qquad k = 2, \ldots, j-1 \qquad \ldots\ldots(5.3.4)$$

$$r_k - 2r_{k-1} + r_{k-2} \geq 0 \qquad k = 3, \ldots, j-1 \qquad \ldots\ldots(5.3.5)$$

$$2\sum_{k=1}^{j-2}(j-k-1)r_k t_k + O_l(r_{j-1},t_{j-1}) \geq (j-1)(j-2) \qquad \ldots\ldots(5.3.6)$$

$$2\sum_{k=1}^{j-2}(j-k-1)r_k t_k - O_l(r_{j-1},t_{j-1}) \leq (j-1)(j-2) \qquad \ldots\ldots(5.3.7)$$

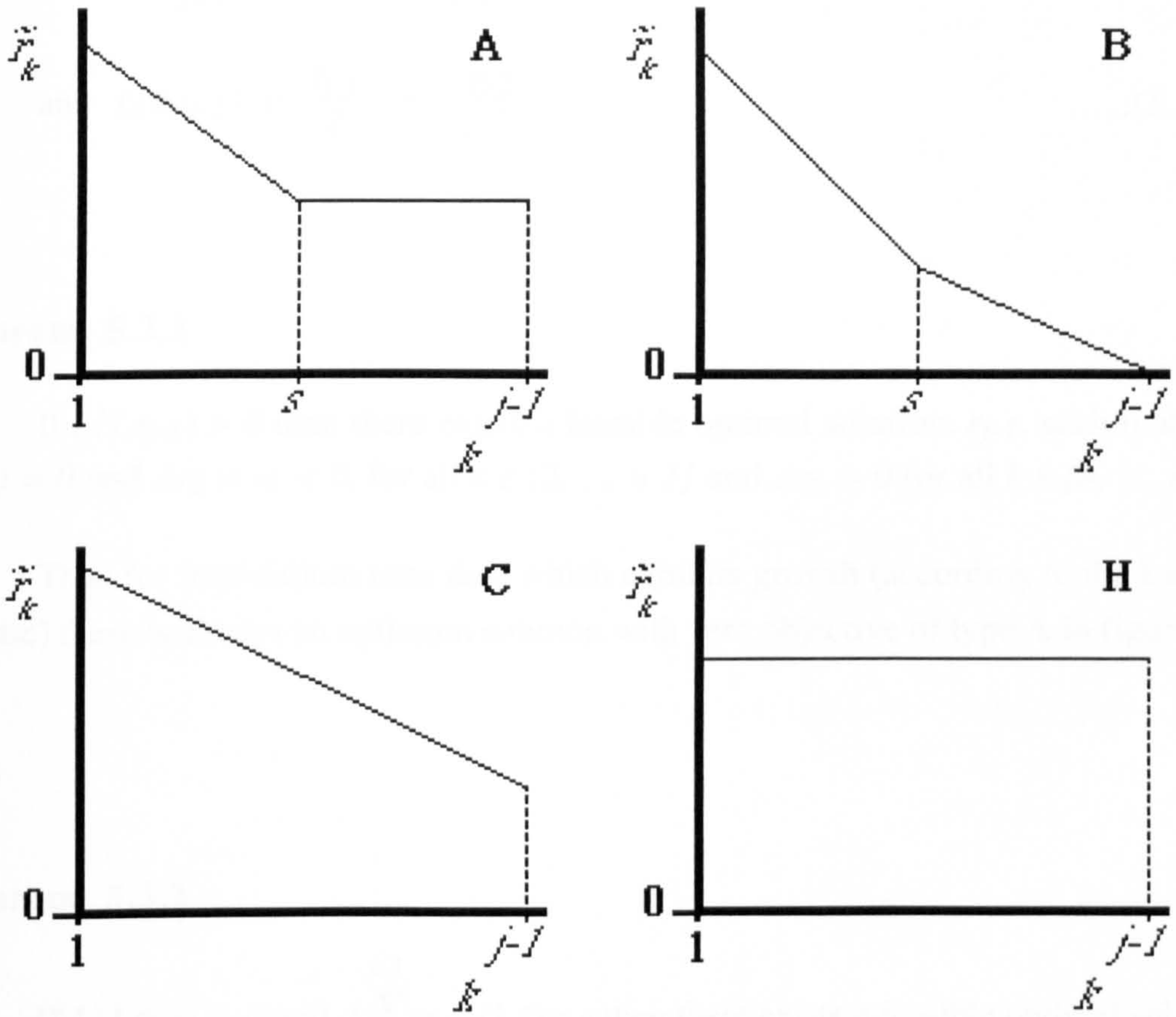
$$\sum_{k=1}^{j-1} r_k t_k = j-1 \qquad \ldots\ldots(5.3.8)$$

where $O_l(r_{j-1},t_{j-1}) = 2(j-2)|L(r_{j-1},t_{j-1})|$.

The optimum solution to this linear programming problem again yields our vector of estimated rates, $\tilde{r}_{j-1}$, and, as for the previous models, the most recently estimated rate, $\tilde{r}_{j-1}$, together with the assumption of exponentiality gives us our one-step-ahead prediction for T_j (see *(5.2.11)* and *(5.2.12)*).

It can be seen that this linear programming problem has fewer constraints than the problem that utilises the *y*-plot and so should be easier to solve. In contrast to the *OTY* model however it was found that this model, which we shall call *OTL*, did not perform very well when the resulting predictions (raw and spline-recalibrated) were compared using the methods described in Chapter 3 with those (raw and spline-recalibrated) from other models on some real data [Brocklehurst 1989]. The variety of optimum solutions

which resulted from application of this model to real data in [Brocklehurst 1989] is shown in figure 5.3-1.



A - *1* non-zero second difference, $\tilde{r}_{j-1} = \tilde{r}_{j-2} = \ldots = \tilde{r}_s$.

B - *1* non-zero second difference, first differences all non-zero, $\tilde{r}_{j-1} = 0$.

C - second differences all zero, first differences all non-zero.

H - first and second differences all zero, $\tilde{r}_{j-1} = \tilde{r}_{j-2} = \ldots = \tilde{r}_1$.

Figure 5.3-1. Classification of optimum solutions resulting from application of the *OTL* model to a number of real data sets.

It can be seen that all the resulting solutions are very simple with at most one non-zero second difference. It was observed in [Brocklehurst 1989] that the resulting solutions were either an *HPP* (**H** in figure 5.3-1) or of type **A** or **C** (and, very occasionally, **B**) in figure 5.3-1. It was also observed that for non-*HPP* optimum solutions the Laplace statistic of $\{r_k t_k | k=1, \ldots, j-1\}$ was always zero and further that these solutions were highly non-unique.

The following theorems are proved in Appendix A.2.2.

Let $\gamma_k = \sum_{s=1}^{k} \tau_s$ where $\tau_k = \sum_{s=1}^{k} t_s$ .....(5.3.9)

and $L(1,t_{j-1}) = \frac{\tau_{j-1}}{2} - \frac{\gamma_{j-2}}{j-2}$ .....(5.3.10)

Theorem 5.3.1

If $L(1,t_{j-1}) > 0$ then there exists a feasible optimal solution, r_{j-1}, such that $L(r_{j-1},t_{j-1}) = 0$ and $\Delta r_k = -c < 0$, for all $k \in \{2, ..., n-1\}$ and $\Delta r_k = 0$ for all $k \in \{n, ..., j-1\}$.

Thus for inter-failure time data which exhibits growth (according to the Laplace statistic) there is always an optimum solution with zero objective of type **A** in figure 5.3-1.

Theorem 5.3.2

If $L(1,t_{j-1}) > 0$ and $4\sum_{k=1}^{j-3} \gamma_k \geq (j-4)\gamma_{j-2}$ then there exists a feasible optimal solution, r_{j-1}, such that $L(r_{j-1},t_{j-1}) = 0$ and $\Delta r_k = -c < 0$ for all $k \in \{2, ..., j-1\}$.

Thus, given additional conditions on the inter-failure times, there exists optimum solutions with zero objective of both types **A** and **C** in figure 5.3-1.

Theorem 5.3.3

If $L(1,t_{j-1}) > 0$ and $4\sum_{k=1}^{j-3} \gamma_k \leq (j-4)\gamma_{j-2}$ then there exists a feasible optimal solution, r_{j-1}, such that $L(r_{j-1},t_{j-1}) = 0$, $r_{j-1} = 0$ and $\Delta r_k = -c-c'$ for all $k \in \{2, ..., n-1\}$ and $\Delta r_k = -c$ for all $k \in \{n, ..., j-1\}$, with $c > 0$ and $c' > 0$ or $c = 0$ and $c' > 0$ or $c > 0$ and $c' = 0$.

Thus, there are circumstances under which there exists optimum solutions with zero objective of type **B** in figure 5.3-1, or of types **A** or **C** in figure 5.3-1 with the most recent rate zero.

Corollary 5.3.4

If $L(1,t_{j-1}) > 0$ then there are at least $n(j)+1$ optimum solutions with zero objective and at most one non-zero second difference where

$$n(j) = \begin{cases} \frac{j}{2} & \text{if } j \text{ is even} \\ \frac{j-1}{2} & \text{if } j \text{ is odd} \end{cases}$$

Thus we can see that in the presence of growth (according to the Laplace statistic) in the inter-failure time data there are many possible optimum solutions with at most one non-zero second difference and with zero objective. Clearly there will be a tendency for the simplest solutions, as shown in figure 5.3-1, to result, but investigation in [Brocklehurst 1989] revealed that, in most instances, there were many other alternative optimum solutions with zero objective and more than one non-zero second difference.

Theorem 5.3.5

The *HPP* vector of rates, r^{h}_{j-1}, is always feasible.

Theorem 5.3.6

If $L(1,t_{j-1}) > 0$ then the *HPP* vector of rates, r^{h}_{j-1}, is feasible, but not optimal.

Theorem 5.3.7

If $L(1,t_{j-1}) \leq 0$ then the *HPP* vector of rates, r^{h}_{j-1}, is feasible and optimal and additionally if $t_k \neq 0$ for all $k = 1, ..., j-1$, then r^{h}_{j-1} is uniquely optimal.

Thus, when the inter-failure times do not exhibit growth (according to the Laplace statistic) the *HPP* solution results.

The proofs of the above Theorems in Appendix A.2.2 show that an analytical solution may be obtained for all inter-failure time data for this model. However, clearly the bad performance of this model is due to the optimisation criterion being too weak resulting in highly non-unique optimum solutions. This model will not be investigated further, although a more highly constrained variation of this model, which is described in the next section, will be included amongst our models the performance of which will be analysed in Chapter 8. This variation arises from Theorems 5.3.1 and 5.3.7 which state that given any vector of inter-failure times, the *OTL* model results in either a solution with zero objective if the data exhibits growth, or an *HPP* solution, otherwise.

5.4 Extension to the OTL Model

The objective is to use the vector of inter-failure times, $t_{j-1} = \langle t_1, ..., t_{j-1} \rangle$ to estimate the vector of associated rates $r_{j-1} = \langle r_1, ..., r_{j-1} \rangle$ in order to make one-step-ahead predictions about the next inter-failure time, T_j. As for the previous models, the most recently estimated rate, $\tilde{r}_{j-1}$, together with the assumption of exponentiality gives us our one-step-ahead prediction for T_j (see *(5.2.11)* and *(5.2.12)*).

Let

$$L(a_{m,j-1},t_{m,j-1}) = \frac{\sum_{k=m}^{j-1} a_k t_k}{2} - \frac{\sum_{k=m}^{j-2} \sum_{s=m}^{k} a_s t_s}{j-m-1} \qquad m = 1, ..., j-2 \qquad .....(5.4.1)$$

where $t_{m,j-1} = \langle t_m, ..., t_{j-1} \rangle$ and $a_{m,j-1} = \langle a_m, ..., a_{j-1} \rangle$.

Consider the full vector of inter-failure times $t_{1,j-1} = \langle t_1, ..., t_{j-1} \rangle$. If there is no growth in this data according to the Laplace statistic then we know, from Theorem 5.3.7, that the *HPP* solution will be optimum for the *OTL* model, so let $r_k = r$, say, for all $k = 1, ..., j-1$. If there is growth in the data, then we know, from Theorem 5.3.1, that there is an optimum solution, $r_{1,j-1}$, to the *OTL* model with $L(r_{1,j-1},t_{1,j-1}) = 0$ so let $L(r_{1,j-1},t_{1,j-1}) = 0$. In the presence of growth in $t_{1,j-1}$ consider $t_{2,j-1} = \langle t_2, ..., t_{j-1} \rangle$. If there is no growth in $\langle t_2, ..., t_{j-1} \rangle$ then let $r_k = r$, say, for all $k = 2, ..., j-1$. If there is growth in $\langle t_2, ..., t_{j-1} \rangle$ then let $L(r_{2,j-1},t_{2,j-1}) = 0$ and then consider $t_{3,j-1}$. We proceed with this process considering progressively smaller vectors of inter-failure times until we find a vector $t_{n,j-1}$ which has no growth and then we let $r_k = r$, say, for all $k = n, ..., j-1$. The objective of this is to constrain the $\{r_k t_k;\ k = m, ..., j-1\}$ to be trend free over all consecutive vectors of the data, $t_{m,j-1}$, which exhibit growth ($m = 1, ..., n-1$), while letting the rates be equal over the largest vector of data, $t_{n,j-1}$, which does not exhibit growth. The problem may thus be formulated as follows.

Let

$$n = \begin{cases} j\text{-}1 & \text{if } L(1_{m,j-1}, t_{m,j-1}) > 0 \text{ for all } m = 1 \ldots j\text{-}1 \\ Min[m \in \{1 \ldots j\text{-}2\}; L(1_{m,j-1}, t_{m,j-1}) \leq 0] & \text{otherwise} \end{cases} \quad \ldots\ldots (5.4.2)$$

noting that, if we define $L(1_{j-1,j-1}, t_{j-1,j-1}) = \frac{t_{j-1}}{2}$, then this will always be positive for positive t_{j-1}. Then, let

$$L(r_{m,j-1}, t_{m,j-1}) = 0 \qquad \text{for all } m = 1, \ldots, n\text{-}1 \qquad \ldots\ldots (5.4.3)$$

$$\text{and} \quad r_k = r, \text{ say,} \qquad \text{for all } k = n, \ldots, j\text{-}1 \qquad \ldots\ldots (5.4.4)$$

From *(5.4.3)* and *(5.4.4)* we can see that we have n variables and $n\text{-}1$ equations and so in order to get a unique solution we must again apply a scaling constraint. The full details of the solution to this problem are contained in Appendix A.3, but the solution, using the usual scaling constraint (see *(5.2.3)*), for the most recent rate (which we are going to use for prediction purposes) is as follows.

$$\tilde{r}_{j-1} = \begin{cases} \frac{(j-n)(j-n+1)}{j-1} & n = 2 \ldots j\text{-}1 \\ 2\sum_{k=n}^{j-1}(k-n+1)t_k & \\ \frac{(j-1)}{j-1} & n=1 \\ \sum_{k=1}^{j-1} t_k & \end{cases} \quad \ldots.(5.4.5)$$

It is proved in Appendix A.3 that, providing $t_k \neq 0$ for all $k = 1, \ldots, j\text{-}1$, then this simple analytical solution is unique.

It can be seen from equation *(5.4.2)* that when the full vector of inter-failure times exhibits growth the estimate of the expected value of T_j (i.e., $1/\hat{r}_j = 1/\tilde{r}_{j-1}$) is simply a weighted sum of the largest vector of inter-failure times which exhibits no growth giving more weight to the most recent inter-failure times. So, in spite of the rather stringent requirements (i.e., that in the presence of growth in the full vector of times the $r_k t_k$ be trend free for all larger vectors which exhibit growth), we have arrived at a very simple solution. As with the previous models, the *HPP* solution results when the full vector of rates does not exhibit growth.

Notice that the first two difference constraints (see *(5.2.1)*) are not applied and the solution of the full vector of rates, $\tilde{r}_1, \ldots, \tilde{r}_{j-1}$, for this model (see Appendix A.3) in the general case would not be expected to satisfy these difference constraints. This extension to the *OTL* model is thus different from the other non-parametric models presented in this

chapter and many of the parametric models in that it does not have the usual reliability growth assumptions.

Although experience of applying the *CM* and *OTY* non-parametric models shows that they perform quite well the major disadvantage of them is that, in spite of their conceptual simplicity, they are computationally intensive. The *y*-plot optimisation model in particular takes a long time to run. The way in which the two models overcome this problem is for the first (i.e., the *CM* model), to divide the total time into a small number, *n*, of equal intervals, as described in section 5.1, and for the second (i.e., the *OTY* model), to use moving windows, as described in section 5.2. It is clear that if the extension to the *OTL* model, or it's recalibrated version, did indeed perform well then it should certainly be included amongst our group of models which we apply to new data in our "multi-modelling" approach simply because it is analytical and thus very easy to apply. Again, it may be expected that any inaccuracy resulting from using such a simplistic model may be eliminated later via the recalibration technique.

Since application of these models with moving windows may be a sensible strategy (in the presence of changes in the data which cannot be adequately modelled) even if it is not a necessity due to computational difficulties, this extension to the *OTL* model will be applied in the usual way over the whole data set *and* with some arbitrarily chosen fixed size moving windows. The performance of the resulting raw and spline-recalibrated prediction systems will be examined in Chapter 8. The software used to apply this model is coded in Pascal and was executing on a Sun 3/80.

Note that when these models are fitted over different intervals of data we will not, in fact, get the same predictions from their *HPP* equivalents, since averaging will occur over the applicable interval of data. More specifically, when the non-parametric models are applied with moving windows we will not get identical *HPP* predictions (unless by coincidence) as the parametric models applied without windows.

6 Automating the Choice of Best Model

From the previous chapters it can be seen that, for each data set, we now have a large number of prediction systems from which to choose. As mentioned previously, in section 3.3, it is commonly the case that the preferred model (according to the *PLR* plots) switches between the various prediction systems not only from one data set to another, but over different intervals of one-step-ahead predictions *within a single data set*. This means that looking at the various plots for analysis in order to decide which model to use for the next prediction has become rather impractical. We wish, therefore, to automate this choice between prediction systems.

Since the prequential likelihood ratio is a global measure of goodness of prediction it is the natural measure on which to base our automatic choice. Since we wish the mechanism for choosing to be truly predictive the best we can do is to compare the *PL* for the various prediction systems (or rather the *PLR*) in the past, and use the model which gives the best *PL* in the past for predictions in the future. For simplicity we shall again be referring to one-step-ahead predictions.

We propose the following. For our choice at stage i, compare the *PL* for all the prediction systems for previous predictions, over $j = i\text{-}w,\ldots,i\text{-}1$, and choose the prediction system for stage i which wins in this comparison. More specifically, if we define our existing set of prediction systems as Ω, we form a new "*meta-prediction system*",

$$\{\hat{F}_i^m(t),\hat{f}_i^m(t) \mid i = s+w+1,\ldots,q\}$$

$$\{\hat{F}_i^m(t),\hat{f}_i^m(t)\} = \left\{\hat{F}_i^m(t),\hat{f}_i^m(t)\,;\ \prod_{j=i-w}^{i-1}\hat{f}_j^m(t_j) = \max_{\omega\in\Omega}\left[\prod_{j=i-w}^{i-1}\hat{f}_j^{\omega}(t_j)\right]\right\} \qquad \ldots\ldots\ldots(6.1)$$

From *(6.1)* it can be seen that this new prediction system will simply be switching between the original prediction systems, $\omega\in\Omega$ within a data set. Clearly, our choice of the window size, w, will be an important factor in the resulting predictive performance of this meta-predictor. In circumstances where comparative predictive performance of the initial group of predictors shows much local variation small window sizes may result in better performance. Conversely, if local variations are erratic and comparative predictive performance is generally global, the predictor may perform better when larger window sizes are used. The performance of the meta-predictor with a suitable range of arbitrarily chosen fixed window sizes will be investigated in the later analysis in Chapter 8. From *(6.1)* it can be seen that these new prediction systems are *truly predictive* and so we can

use the analysis techniques described in Chapter 3 to compare the meta-prediction systems with the original raw and recalibrated prediction systems.

Since this meta-predictor is choosing from amongst all the raw and recalibrated predictions it seems likely that when recalibration of the initial raw prediction systems is effective in eliminating bias there will not be significant errors in the meta-predictors. Further, since the meta-predictor will be switching between the various predictors, bias, if present, may be non-stationary. This implies that we should not expect substantial improvement in the predictive accuracy via recalibration of these meta-predictors and so the recalibrated version of this new predictor will not be investigated.

It is clear that a more sophisticated mechanism could have been used at this stage. For example we might dynamically choose an optimum window size as we move through the data set. In other words use a window on which to base the next comparison which gave us the best performance for the most recent one-step-ahead prediction. In order that this be truly predictive we would need to be making comparisons on the last two one-step-ahead predictions. But since, at this stage, we are only trying to automate the choice that the user would make, using fixed window sizes seems sufficient. Other more sophisticated techniques for combining over a group of prediction systems are discussed in Chapter 9.

7 The Data Collection Activity

The first two data sets of inter-failure times analysed in the following chapters, which we shall label *CISI1* and *CISI2* (from [Gaudoin 1988]), are listed in tables B.1-1 and B.2-1 in Appendix B in Volume II.[2] For confidentiality reasons we cannot include details of the data collection process involved for these two data sets or about the systems to which they relate. They contain *168* and *394* data points, respectively, which are inter-failure times measured in seconds of C.P.U. execution time.

Figure B.1-3 in Appendix B in Volume II shows the *CISI1* data plotted as the cumulative number of failures against the total elapsed time. This shows that very early on in the data there is a period of stable reliability followed by reliability decay while reliability growth appears to start at about the 75^{th} inter-failure time when the total elapsed time is approximately *1500* seconds. This behaviour is fairly typical of such failure data; the fact that reliability growth is only seen in the later data can often be attributed to "teething problems" which occur early on in the testing process or in early operation. From table B.1-1 it can be seen that at the point at which reliability growth starts the data is notable for two inter-failure times ($t_{77} = 66$ and $t_{78} = 95$) which are significantly large when compared to immediately proceeding times. Similarly large times are also seen at other points in the data ($t_{34} = 149$, $t_{101} = 440$ and $t_{114} = 825$).

The equivalent plot for *CISI2* is less typical (see figure B.2-3). In general, reliability growth seems present after about the 150^{th} failure, but this growth is not smooth and local periods of reliability decay occur even late on in the data. In particular notice the clustering of failures near the end of the data set between the 320^{th} and 380^{th} failures. In this region the failure rate[3] has increased by about four times compared with the rate just preceding and just after this cluster of failures. Similar clustering also occurs earlier in the data, from failures *170-210* and *240-265*. Prior to the 150^{th} failure there is not significant growth but there are many local fluctuations in the trend between growth and decay. In general the trend in the data is highly irregular. From table B.2-1 it can be seen that some particularly large inter-failure times also occur for this data set (for example, $t_{224} = 723$, $t_{225} = 563$, $t_{226} = 669$ and $t_{239} = 1337$).

---

2 All tables and figures referred to throughout this Chapter are contained in Appendix B in Volume II.

3 The *failure rate* referred to throughout this section is a crude estimate obtained by taking the number of failures and dividing this by the elapsed time.

As discussed in Chapter 2, the reliability models may be applied in accordance with the failure data behaviour by taking account of changes in the trend and of outliers in the data, but this approach will not be adopted here. As we shall see in the following analysis in Chapter 8, these aspects of the failure data can have a large influence on the behaviour and comparative predictive performance of the resulting prediction systems.

The second collection of data to be analysed comes from four years of operational use of a single user work station which was installed at the City University on the 18th March, 1985. Data was collected from the user's viewpoint and included real (or wall-clock) time of occurrence of each (user-perceived) failure (recorded to the nearest minute), together with the identity of the particular fault which caused the failure (so that time to first occurrence of each unique fault can be recovered and hence the required inter-failure times). Additionally details of the type of usage and the version of the operating system in use at the time of each failure, the type of the associated fault and the severity of failure were recorded. Such records allow various subdivisions of the data to be extracted and analysed separately, for example failures due to a particular fault type, failures occurring under a particular product version and so on. A large number of data sets result from subdividing the data in this way. Due to space constraints only a limited number of these will be analysed in Chapter 8.

There are generally two reasons for subdividing a data set. The first is to eliminate possible irregularities or change-points which are difficult to model in the full data set and the second is to examine failure behaviour of particular classifications as being interesting in themselves. In fact, as we shall see, the subdivisions which can be made from the recorded data do not seem to successfully eliminate such changes present in the complete data set. Further, since the major objective here is to validate the various techniques for achieving prediction systems (as opposed to examining the data), data sets selected for further analyses in Chapter 8 will be those which exhibit unique behaviour since those with similar trend will tend to give similar results for the relative predictive accuracy of the prediction systems.

All start times of different usages were recorded and these usages included power-on, power-off and idle time (times when the machine was powered on but not being used) together with usages such as document preparation and program development. The end time of one usage could thus be taken as the start time of the next usage. In this way it was possible to extract inter-failure time data with respect to a number of different time metrics. Failures with respect to calendar time could be extracted but, as discussed in Chapter 2, this is clearly an inappropriate time metric since it includes times when the machine is switched off and so failures cannot occur.

The first data set extracted was thus times between successive failures of unique faults (i.e., ignoring failures of previously seen faults) for all usages, versions etc., only omitting "switched-off" time from the time axis and including "idle", "power-on", and "power-off". The time metric for this data is therefore "switched-on" time. This data, which we have labelled *USCOM*, is listed in table B.3-1 and consists of *414* inter-failure times in all[4].

Since failures were recorded as a result of observation by the user, and because many of them were "usability problems", i.e., the encountering of features of the system which caused difficulty for the user, even though the system was behaving according to specification, it is likely that "hands-on" time would be a more appropriate time metric than "switched-on" time. This results in a data set which we shall label *USBAR* (see table B.4-1), which consists of inter-failure times of all first occurrences of unique faults which occur when the machine is switched-on and in usages other than idle, power-on and power-off, with time intervals in these usages omitted from the total time axis. We would not expect many failures in these usage modes as the machine is not doing much at these times; there were only *20* unique faults which resulted in failures in these usage modes, *3* of which also resulted in failures in other usage modes, giving a total of *397* failures in *USBAR*. We would thus expect the rate of occurrence of failures to be higher in *USBAR* than in *USCOM* (due to intervals of idle time which are almost failure-free).

Observation of figures B.3-2 and B.4-3 indicates the trend in *USCOM* and *USBAR* to be fairly similar although, as expected, the failure rate for *USBAR* is much higher (about double) than for *USCOM*. The similarity in the trend seen for both these data sets indicates that the time corresponding to idle, power-on and power-off, is approximately evenly distributed over the total time axis. The time when the system was inactive was omitted from the time axis because we wish to model the failure behaviour against a time axis which represents, crudely speaking, a constant "stress" on the system. In other words, if there are long periods of inactivity, the resulting failure behaviour against total time will look very non-stationary with large periods of time with no (or few) failures. As this is not the case it may not have been necessary to omit inactive time from the time-axis, although it is likely that it may be necessary to do so in the case of further subdivisions of the data (see below) because such periods of inactive time may become more significant. Since the trend in *USCOM* and *USBAR* is similar, in the later analysis

---

4 The inter-failure times in this data which are shown as *0.5* actually came out to be zero after extraction and were changed to *0.5* before application of the models. This is because these zeros are due to rounding errors, and so they have been set to a suitably small value.

in Chapter 8, we shall analyse *USBAR* and not *USCOM* since hands-on time seems the more appropriate time metric.

Further examination of figure B.4-3 shows that prior to the 200^{th} failure the trend in the data is fluctuating, although generally there is no significant growth, while in the latter part of the data there would appear to be reliability growth. There appears to be a definite and sudden decrease in the failure rate (by about ten times) at the 260^{th} failure, when the total elapsed time is about *16,000* minutes and a number of particularly large inter-failure times occur. This is an example where it is inappropriate to discard these large times as outliers before application of reliability models since they clearly correspond to a change in the trend. Particularly large times are also seen in the later data (for example $t_{340} = 4973$).

As mentioned above, using records of usage, operating system version, fault type, and failure severity, it is possible to partition the data into subdivisions, each of which can be analysed separately. For these data sets, as for *USBAR*, "hands-on" time is chosen as the measure of use. Note that, in certain circumstances, it is possible for the same fault to cause a failure in more than one subdivision; for example, the same fault may cause a failure to occur under two different versions of the operating system. This means that extraction of the times to failure of first occurrence of unique faults for particular subdivisions must be done by first finding the subset of all failures which occur under the particular subdivision, and then omitting failures from repeated faults within that subdivision. For subdividing via the usage or product version the time metric is adjusted to be hands-on time for that particular usage or product version while for the remaining subdivisions the time metric is the total hands-on time, as for *USBAR*.

The first subdivisions that we shall consider are those based on the *usage under which the failures occur*. The category of usage during which the largest number of unique failures occurred, *155* in all, was document preparation *USROFF* (see table B.5-1), which consisted of a UNIX word processing package. Figure B.5-3 shows that for *USROFF*, as for *USBAR*, there is a marked decrease (also by about ten times) in the failure rate from just after the 85^{th} failure where some particularly large inter-failure times occur (for example, $t_{88} = 1613$). Other particularly large times also occur later on in the data (for example, $t_{100} = 2264$ and $t_{101} = 3218$). Also, as with *USBAR* there is a point (about *60*) prior to which the trend in the data is fluctuating and after which there is significant growth.

A further *104* failures occurred when compiling, running and so on, Pascal programs, and these form data set *USPSCL* (see table B.6-1). Here (see figure B.6-3) there is also a sudden decrease in the failure rate, although not as dramatic as for *USBAR*

and *USROFF*, from the 45^{th} failure, with stable reliability prior to this point followed by very gradual reliability growth. Most of the relatively large (compared with previous data) inter-failure times, except for $t_{38} = 289$ and $t_{84} = 1596$, correspond to this change point in the data. Note that the comments relating to reliability growth refer to global growth, while in the region over which the later analyses in Chapter 8 will be conducted, no significant growth is present.

Finally we have general usage, *USGENL* (see table B.7-1), which consisted of all usages other than the previous two stipulated usages. Here the cumulative failure plot in figure B.7-2 looks very different from the failure plots of the other two usages. There is stable reliability throughout the data set until the last few failures where there is sudden and dramatic reliability growth.

Comparison of the three cumulative failure plots for these different usages show that, although the most failures occurred during document preparation, we cannot immediately conclude that this is more unreliable, since it was a very heavily used utility; document preparation has, in fact, the lowest failure rate when calculated over the complete data set, followed by Pascal programming and then the general usage category. The initial failure rates are about the same for these usages and so the low failure rates considered over the whole data set for *USROFF* and *USPSCL* are due to reliability growth later on during continued use of these two utilities. It is perhaps not surprising that little growth is seen in the general usage category since this is made up of a number of infrequently used utilities. Since there is not much growth in *USGENL* we shall limit our analyses in the following chapter, for the subdivisions based on usage, to just *USROFF* and *USPSCL*.

During the use of the work station a number of *operating system versions* were installed. Thus, *USBAR* can be divided into a number of data sets according to the version in use at the time. We shall label these data sets, *PV200, PV220, PV400* and *PV502* to denote product versions *2.00, 2.20, 4.00* and *5.02* respectively (see tables B.8-1, B.9-1, B.10-1 and B.11-1). As with usage, times to first occurrence of a fault running under each product version were considered and so some faults may be included in more than one of these data sets. In fact there are only *12* more failures in total for these data sets than for *USBAR*; thus, since the product versions where installed in sequence, these data sets will be approximately equivalent to slicing the complete data, *USBAR*, into four.

It can be seen from figure B.8-2 that version *2.00* was installed for a very small part of the systems' life (with respect to hands-on time) as compared with the later 3 operating systems shown in figures B.9-2, B.10-2 and B.11-2. From these figures is can

also be seen that the failure rate for product version *2.00* and for about half of the installed time of version *2.20* is an order of magnitude higher than the failure rate in the remainder of the systems' life-time. The marked point of change to a lower failure rate while version *2.20* is installed clearly coincides with the previously identified point of change in the complete data set, *USBAR*, when the total elapsed time is about *16,000* minutes. For *PV200* there is no reliability growth (in fact there is a cluster of failures at the end of this data set) while for *PV220* there is growth from about the 170^{th} failure while previously there is fluctuation in the trend. For the final two product versions again there is fluctuation with only significant growth right at the end of each data set.

In general there is nothing to suggest that the different rates (which decrease with higher versions of the operating system) are related to anything other than the fact that there is reliability growth over the full data set. Neither are there are any significant signs of reliability decay at the beginning of these data sets which one might expect due to the "teething" problems likely to be encountered on installation of a new operating system, or that any changes in the failure rates, or the trend, in *USBAR* correspond to the times when there is a change of operating system. This indicates that any effect that may be present due to different operating systems is negligible when compared to other factors which affect the overall reliability of the system. In view of this we shall not analyse the subdivisions based on operating system version any further.

Next the failures were classified according to the *fault type*, which consisted of software, hardware, documentation, usability and user errors:

- *software failure* - the system software did not behave as required, e.g., incorrect output, operating system crash.
- *hardware failure* - e.g., hard disk error.
- *documentation* - user documentation found to be incorrect.
- *user* - incidents which were due to mistakes, or lack of knowledge, on the part of the user, e.g., not realising that a certain utility required each of its directives to start on a new line.
- *usability problem* - the system software is behaving as intended, but it is difficult to use, e.g., output from a certain utility could not be printed directly, but had to be written to a file which was then printed. (It is often difficult to distinguish this type of failure from user).

The data sets resulting from a subdivision via fault type are labelled *TSW, THW, TDOC, TUSER* and *TUSAB* (see tables B.12-1, B.13-1, B.14-1, B.15-1 and B.16-1).

Since each failure results from a unique fault, and each fault has a unique type, faults will not overlap over these subdivisions. Hence summing the total number of failures over the types, gives *397*, the total number of failures in *USBAR*, and extraction of first occurrences of unique faults may be done before, or after, subdivision into fault types.

Corresponding to the previously identified sudden decrease in the failure rate in *USBAR*, from figures B.12-3, B.13-2, B.14-2 and B.16-3 we can see similar behaviour near this region. For *TSW* this is mainly due to a occurrence of a single particularly large inter-failure time at this point ($t_{79} = 9549$). Again, reliability growth is present just prior to this point of change and continues for the remaining data. Other particularly large inter-failure times occur in the data, $t_{86} = 4179$ and $t_{105} = 7984$, but these are not as large with respect to previous data as t_{79}.

For *THW* there is reliability decay prior to *12000* minutes where the failure rate suddenly decreases by a huge amount and after which there is stable reliability. In fact, this is because the hard disc, which was faulty, was replaced at this time. For *TDOC* there is also reliability decay in the early data while in the later data there are signs of reliability growth. For *TUSER* (see figure B.15-2) growth starts just after the beginning of the data.

For *TUSAB* there are two change points in the data (after t_{77} and after t_{105}) but these are less dramatic than for the *THW* and *TDOC*. Here the cumulative failure plot is typical of reliability growth data although growth does not start until after the 25^{th} inter-failure time and later in the data there are some local clusters of failures (e.g., between t_{60} and t_{77} and between t_{95} and t_{105}). After these clusters there are some relatively large inter-failure times (e.g., $t_{78} = 1104$, $t_{82} = 1553$, $t_{106} = 1622$, $t_{111} = 1710$ and $t_{115} = 3569$) although these, particularly the second group, appear to mark the beginning of increased reliability growth, rather than being unique outliers.

The final subdivision was the *failure severity* which was classified into major (*SMA*), minor (*SMI*) or negligible (*SNE*) (see tables B.17-1, B.18-1 and B.19-1). In this subdivision it is theoretically possible to have different manifestations of the same fault in more than one severity class but in practice, for this data set, each fault had a unique severity classification. It can be seen that there were very few major failures; most failures had either minor or negligible severity from the user's point of view. For the major failures, from figure B.17-2, we can see no significant signs of reliability growth. The pattern in the trend for the minor and negligible failures (see figures B.18-2 and B.19-2) is very similar to *USBAR* with fluctuation early on but with no significant growth until after *10,000* minutes of total elapsed time. Again a period of sudden decrease in the failure rate can be identified for both *SMI* and *SNE* when the total elapsed time is about

16,000 minutes. The failure rates for these two severity classifications are approximately the same. Due to the similarity of the trend in these two data sets and *USBAR* these will also not be investigated further.

Due to the large number of data sets available via the subdivision of *USBAR*, the smaller data sets (*THW*, *TDOC*, *TUSER* and *SMA*) will also be omitted from the later analysis in Chapter 8. This is not because the models and techniques described cannot be applied to small data sets and the approach would be exactly the same as for larger data sets (although the stage at which we start to apply the models and the recalibration must be chosen as earlier than for larger data sets). It is simply due to space constraints.

8 Analyses of Resulting Prediction Systems

In this chapter we shall discuss and compare the predictive accuracy, using the techniques described in Chapter 3, of all the resulting prediction systems outlined in previous chapters on the data sets selected in Chapter 7.

In application of the *8* raw parametric models described in section 2.1 *20* data points were chosen quite arbitrarily for making the initial raw predictions for each of the data sets; so, in the notation of *(2.7)*, $s = 21$. Then for each of the raw parametric models a sequence of one-step-ahead predictions was made. We shall refer to the resulting raw prediction systems as *JM*, *GO*, ... and so on.

For the non-parametric models described in Chapter 5, different data were used to get the initial predictions depending on the model being applied and on the window size, if applicable, being used. The *CM* model was applied without windows and the number of equal intervals into which the total time interval was divided was kept constant at *30* (i.e., in the notation of section 5.1, $n = 30$). Three variants of this model were used by applying the difference constraints (see *(5.1.2)*) up to orders of *1*, *2* and *3* (i.e., in the notation of section 5.1, $d = 1, 2$ and *3*). A series of one-step-ahead predictions were made using the first *30* inter-failure times to obtain the initial raw predictions; so, in the notation of *(2.7)*, $s = 31$. The resulting raw prediction systems will be referred to as *CM1*, *CM2* and *CM3*, respectively. The *OTY* model was applied with fixed size moving windows of size *20* and *50*, so, in the notation of *(2.7)*, $s = 21$ and *51*, respectively and we shall refer to the resulting raw prediction systems as *OTY20* and *OTY50*. The extension to the *OTL* model was also applied with fixed size moving windows of size *20* and size *50* and without windows, using, as for the parametric models, the first *20* inter-failure times to obtain the initial raw predictions; we shall refer to the resulting raw prediction systems as *OTL20*, *OTL50* and *OTL* and in the notation of *(2.7)*, $s = 21, 51$ and *21*, respectively. Thus, for each data set, we shall be investigating *8* non-parametric raw prediction systems.

All the raw prediction systems were spline-recalibrated using the predictive recalibration method described in Chapter 4. First, the spline-recalibration method was applied using the first *15* raw predictions to obtain the u-plot for the initial recalibrated prediction (so, in the notation of *(4.5)* $p = s + 15$, where the first raw prediction is made at stage s). The series of one-step-ahead recalibrated predictions was then obtained by using all the raw predictions available at each stage; so, one new u is added into the u-plot used for recalibration at each successive prediction stage. An *S* added onto the end of the

model names will be used to denote the resulting spline-recalibrated prediction systems (e.g., *JMS*, *CM3S*, *OTY50S*, ... and so on).

Next the spline-recalibration method was applied using a number of fixed size moving windows across the raw predictions, as described in section 4.2. Window sizes of *20*, *30*, *40* and *50* were applied. An *Sw* added onto the end of the model names will be used to denote the resulting spline-recalibrated prediction systems with windows of size *w* (e.g., *JMS20*, *CM3S40*, *OTY50S30*, ... and so on). For each window size, *w*, *15* raw predictions were used to obtain the first recalibrated predictions (so, in the notation of *(4.5)*, $p = s + 15$, as for recalibration without windows) and then a new *u* is added to the *u*-plot for recalibration at successive prediction stages until the *u*-plot is based on *w* raw predictions. Thereafter only the most recent *w* raw predictions are used in the *u*-plot for recalibration at each prediction stage.

Since there are *16* raw prediction systems, application of the recalibration technique with and without windows, as outlined above, results in a total of *96* (*16* raw and *80* spline-recalibrated) different prediction systems for each data set. Using the method described in Chapter 6 dynamic selection via the *PLR* using fixed size moving windows will be made over these *96* different prediction systems. As discussed in Chapter 6, there are circumstances under which forming these meta-predictors using small window sizes across the initial group of predictors may be preferable while in other circumstances larger window sizes may be better. In view of this a range of window sizes were applied, *1, 2, 5, 10, 20, 30, 40* and *50*. The resulting meta-prediction systems will be referred to as *M1, M2, M5, M10, M20, M30, M40* and *M50*, respectively.

In the following analyses the raw non-parametric predictors will be compared against the *"best"* raw parametric predictor and the recalibrated non-parametric predictors will be compared against the *"best"* recalibrated parametric predictor in order to assess whether the non-parametric models result in prediction systems which are as good as the conventional parametric models. In order to assess the efficiency of the spline-recalibration technique (applied without windows) the recalibrated prediction systems will be compared, in each case, with their raw equivalents (e.g., *JMS* versus *JM*, *OTY50S* versus *OTY50*, ... and so on). Then, in order to assess whether the recalibration technique can be made more effective via the application of windowing the prediction systems resulting from recalibration applied with the various window sizes will be compared against the equivalent prediction systems which result when recalibration is applied without windows (e.g., *JMS20* versus *JMS*, *OTL50S40* versus *OTL50S*,.... and so on).

Finally the meta-predictor is compared against the single predictor which is chosen as *"best"* according to all the previous *PLR* analyses. In each case that which is chosen as *"best"* for a particular data set is not necessarily the best according to the *PLR* analysis throughout the data, but is just generally best overall. There is usually no single predictor which is truly consistently best throughout the data. In applying the meta-predictor we are really just crudely formalising the process of choosing the next prediction we should use from the past *PLR* analyses. The judgement as to which predictor is *"best"* is a retrospective judgement whereas our meta-predictor is truly predictive. It would thus be very unlikely that the meta-predictor did out-perform the *"best"* single predictor. When assessing the performance of the meta-predictor we are thus generally satisfied if it is about as good as the (retrospectively identified) *"best"* single predictor.

All the figures and tables relating to the following analyses in this chapter are contained in Appendix B in Volume II. Due to space constraints it is necessary to limit the number of plots shown; in the following analyses we have selected plots which are of particular interest although some of the observations made also consider those omitted. Due to the difference in the stages at which the different predictors make their initial predictions (i.e., different s and p depending on the predictor) for consistency all the analyses will be carried out over a range of predictions from $i = 66$[5] up to prediction of the final data point except for the meta-predictors where analysis will start from $i = 71$.

Tables B.1-2.1, B.1-2.2, B.2-2.1, ... in Appendix B in Volume II show the y- and u-plot significance levels for all the prediction systems on all the data sets analysed. These significance levels are based on the K distance of the plots from the $45°$ line, as discussed in Chapter 3. Although the y-plots are included in the tables little attention will be given to these plots in the following analyses. This is because, as observed in the simulation exercise in Chapter 4, the y-plot results do not seem to adequately describe changes in the trend which are of very much interest, particularly with respect to the extent to which we can expect recalibration to eliminate bias in the raw predictions.

8.1 Data set CISI1

For this data set the *JM, GO, MO, LM* and *LNHPP* models give *HPP* predictions for $i \sim 25$ to *100*. Although in Chapter 7 it was observed that growth in the data starts prior to the 100^{th} failure it should be noted that this related to local reliability growth whereas these models are applied over all the previous data. From $i \sim 100$ to *150* the *LM*

---

5 i is used to denote the number of the inter-failure time, t_i, throughout this chapter.

and *LNHPP* models go to the limiting cases of the *JM* and *GO* models respectively; thus, their estimated rates on each fitting of the models (as opposed to successive one-step-ahead predictions) decrease linearly with *i* in this interval. The raw median predictions in figure B.1-4 coincide in the early data for the *5* models for which *HPP* predictions result while later on in the data there is huge disagreement, increasing with failure number, between the median predictions from the *8* models and the *u*-plots (see figure B.1-5) indicate that the "true" median may lie near the *MO* models' predictions. From table B.1-2.1 we can see that the *DU*, *LV* and *KL* models have highly significant *u*-plots and according to figure B.1-5 predictions from these models are grossly pessimistic, particularly the *LV* and *KL* models. According to table B.1-2.1 the remaining models are not giving significantly biased predictions.

The *log(PLR)* plot (see figure B.1-6) confirms that the *DU, LV* and *KL* models are poor predictors compared with the remaining parametric models over most of this data set except prior to *75* where they are better and in a small interval near the end where the plot suggests that the *DU* and *MO* models are marginally better than the rest. Although the plots of raw predictions from failure *21* are not shown here one-step-ahead median predictions from the raw parametric models decreased prior to *75* and according to the *PLR* the *DU, LV* and *KL* models performed better than the rest over this interval. This is to be expected since these are the only parametric models which model reliability decay. In the presence of reliability growth or stable reliability the other models seem to be performing better on this data set. The large jumps (or drops) in the *log(PLR)* plot (figure B.1-6) coincide with the occurrence of a number of comparatively large inter-failure times previously identified in Chapter 7. The jumps indicate that the predictive *pdf* of the *DU* model evaluated at these points is small relative to the other models (with the exception of the *LV* and *KL* models at $i = 77, 78$). This suggests that the *pdf* for the *DU* model has smaller values at the larger end of the scale except when the *LV* and *KL* models are also predicting reliability decay, in which case they appear to have even smaller tails. It seems likely that the remaining models' inability to cope with reliability decay, and their tendency towards optimistic predictions, means that we should expect these jumps upward in the presence of exceptionally large inter-failure times. Even not taking account of these jumps at large times the *JM*, *GO*, *MO*, *LM* and *LNHPP* models are steadily better than the other three models over much of the data set.

The application of the non-parametric models with windows allows them to capture more quickly the local changes in the data. Thus, *OTL50* and *OTY50*, for example, give *HPP* predictions only until about the 80^{th} failure whereas the *OTL* model, gives *HPP* predictions right up until the 100^{th} failure, identical to those from the parametric models. For window size *20*, as we would expect, the models fluctuate frequently between *HPP* predictions and deviations from this, throughout the data. Note

that the *HPP* predictions will only be identical for different models when they are applied with the same window size. Corresponding coincident median predictions can be seen in figure B.1-7. Notice also that the *OTL*, *OTL20* and *OTL50* predictions frequently coincide when the predictions are not *HPP*. This is not very surprising since this will occur whenever the largest vector of data which exhibits no growth is the same for the different intervals of data on which these models are based. It can also be seen that the medians for the *CM1*, *OTL*, *OTL20* and *OTL50* models seem to be very noisy. Comparing this with the raw data (see table B.1-1 and figure B.1-3) it can be seen that these models, particularly *CM1*, are responding to the very large inter-failure times by jumping upwards suddenly on the next prediction, immediately followed by a more gradual decrease as subsequent, less extreme data, is taken into account and the effect of these large times becomes less significant. From figure B.1-4 it can be seen that similar, although much smaller, jumps upwards in response to large times occur for some of the raw parametric median predictions. Comparison of figure B.1-7 with figure B.1-4 shows that apart from these sudden jumps for some of the non-parametric models, the predictions are more similar than the predictions from the parametric models. The *CM1* model appears to be more optimistic than the other non-parametric models and this is confirmed by the *u*-plot (see figure B.1-8). In fact table B.1-2.1 reveals that from among the raw non-parametric models this is the only *u*-plot which is significant for this data set.

The *log(PLR)* plot (see figure B.1-9) suggests that the *OTY50* model seems to be slightly better than the other non-parametric models but the difference in accuracy between the predictions from these models is pretty marginal; apart from the jumps which again coincide with particularly large inter-failure times little significant steady increases or decreases are seen in the plots. Closer examination of this *PLR* analysis, and comparison with figure B.1-7, shows slightly sharper decreases in the *PLR* for the *CM1* model coinciding with those regions where its median predictions jump suddenly upward in response to large inter-failure times, indicating that it is in these regions that the *CM1* model is too optimistic. Comparing figure B.1-9 with figure B.1-6, though, it can be seen that the non-parametric models are generally much closer in accuracy than were the parametric models. Figure B.1-10 shows that some of the non-parametric models (in particular *OTY20, OTY50* and *OTL20*) are as good as the best of the parametric models (compare with figure B.1-6) while the remaining non-parametric models are only marginally worse. It would seem, then, that the increased noise in the non-parametric model predictions has not resulted in worse predictive accuracy than the parametric models, as might be expected.

Comparison of figure B.1-11 with B.1-4 shows that recalibration brings the median predictions from the parametric models closer together, although the predictions still differ, and more so towards the end of the data set. The *JM, GO, MO, LM* and

LNHPP median predictions are adjusted for optimism throughout (the *MO* model being only adjusted slightly). The *DU* median is adjusted for optimism in earlier stages, while later on there is an adjustment for pessimism. The *LV* and *KL* medians have slight adjustments for optimism very early on and large adjustments for pessimism in the latter part of the data set. This suggests non-stationarity for the errors in the raw *DU*, *LV* and *KL* models (i.e., the shape of their *u*-plots is changing over time) and this is also indicated by the *u*-plots for these recalibrated predictions in table B.1-2.1 and figure B.1-12. The *u*-plots for the recalibrated versions of these three models are still significant and indicate that the recalibrated predictions are still pessimistic although comparison of figures B.1-12 and B.1-5 shows a marked improvement in the *u*-plots after recalibration particularly for the *LV* and *KL* models. All the rest of the models, with the exception of *MO*, have insignificant *u*-plots after recalibration but since the *u*-plots for these raw models were originally insignificant we would not expect much improvement to be gained via recalibration. For the *MO* model recalibration has caused the *u*-plot to become significantly pessimistic while it was previously insignificant; this indicates that there is also non-stationarity in the raw *MO* prediction errors.

Figure B.1-13 confirms that there is a marked improvement in predictive accuracy to be gained by recalibration for the *LV* and *KL* models and for the *DU* model in the latter part of the data set while performance seems to fluctuate, although not dramatically, between raw and recalibrated for the other models. Figure B.1-14 shows that, apart from the large jumps again coincident with extreme data points, which are carried through to the recalibrated versions, the *DUS*, *LVS* and *KLS* predictors are marginally worse than the remaining recalibrated predictors in the latter half of the data but comparison with figure B.1-6 shows that the recalibrated predictors are marginally closer in predictive accuracy than were the raw predictors.

Figure B.1-15 shows the *K* distances of the *u*-plots constructed using a moving window of size *30* across the raw predictions from the *8* parametric models. This clearly shows that for the *DU*, *LV* and *KL* models there is non-stationarity in the departure of the raw predictions from the truth; the predictors appear to be becoming more and more pessimistic as *i* increases. Since early predictions in the data set are not pessimistic we can see quite clearly how an insufficient adjustment via recalibration will be made for the pessimism later on in the data set. For the remaining models this non-stationarity is not so pronounced. In particular it is not obvious why the *u*-plot for *MOS* is significant. Notice that the significance levels for the *y*-plots in table B.1-2.1 are contradictory to the observed result for recalibration of the parametric models; those with bad *y*-plots for the raw predictions result in good *u*-plots after recalibration and those with good *y*-plots initially result in bad *u*-plots after recalibration. In other words, the *y*-plots do not appear to detect the non-stationarity in the prediction errors for *MO*, *DU*, *LV* and *KL*.

For the non-parametric models comparison of the recalibrated medians with the raw medians (figures B.1-16 and B.1-7) show that for most of the models small adjustments have been made for optimism while for the *CM1* model this adjustment is greater than for the other models and now most of the predictors are in slightly closer agreement than before. Notice how noise in the raw median predictions is carried through to the recalibrated median predictions. Table B.1-2.1 shows that the *u*-plot for *CM1* is dramatically improved by recalibration, for *CM2* the originally insignificant *u*-plot has become significant and for the remaining models the significance levels remain about the same. Figure B.1-17 shows that *CM2S* is pessimistic in places. Figure B.1-18 shows that, as for some of the parametric models (i.e., *JM*, *LNHPP* etc.), there would appear to be fluctuation in the effectiveness of recalibration except in this case these fluctuations for some of the models are much more extreme. The *CM1* model shows most improvement via recalibration. Figure B.1-19 shows that there are not great differences in accuracy between the resulting recalibrated non-parametric predictors and from figures B.1-19, B.1-20 and B.1-14 we can see that some of the recalibrated non-parametric predictors (e.g., *CM1S* and *OTL20S*) are as good as the best of the recalibrated parametric predictors while the remaining recalibrated non-parametric predictors are only marginally worse.

Figure B.1-21 shows that the raw non-parametric models are in fact fluctuating between being significantly biased and not so, as the data evolves and it is presumably this non-stationarity in the prediction errors which results in the fluctuations seen in the *PLR* analyses of the recalibrated versus the raw predictors. It is not clear, though, from this plot why *CM2S* has a significant *u*-plot. Comparing figure B.1-21 with figure B.1-7 seems to confirm the earlier suggestion that the optimism in the *CM1* predictions mainly occurs just after large inter-failure times where the median predictions have jumped upwards.

Next we shall consider the recalibration technique applied with windows in order to assess whether this eliminates bias in those instances where recalibration without windows did not (i.e., particularly for the *DU*, *LV* and *KL* models). Table B.1-2.1 shows the resulting *y*- and *u*-plot significance levels when windows of size *20, 30, 40* and *50* are applied (*S20, S30, S40* and *S50* respectively). Here it can be seen that the only significant *u*-plot is for *KLS50* for which marginal pessimism is still present. In fact, for recalibration with windows of size *20*, *30* and *40* all the *u*-plots for all *16* models are insignificant at the 20% level, while for window size *50*, all the *u*-plots are the same, or improved, when compared with the recalibrated applied without windows. This indicates that, as we would expect, smaller window sizes tend to eliminate those effects of non-stationarity in the raw prediction errors previously identified. We now need to assess whether the application of the recalibration technique using moving windows has

indeed resulted in better predictive accuracy than recalibration when applied without windows, or whether this decrease in bias has been bought at the expense of an increase in noise.

The medians for the parametric models recalibrated with window size *40*, shown in figure B.1-22, do indeed look noisier than the earlier recalibrated medians (figure B.1-11) but they are also in closer agreement than previously. According to figure B.1-23 the predictions are marginally closer in accuracy than when recalibration is applied without windows (compare with figure B.1-14). This tendency towards closer agreement and more noise increases as the window size used for recalibration decreases. From figure B.1-24, it can be seen that recalibration with windows has actually given improvement with respect to the jump coincident with the large inter-failure time, t_{101}, for some of the models where without windows there was no such improvement. Apart from the jump this figure shows that there is marginal improvement in some regions in the latter half of the data for *DUS40*, *LVS40* and *KLS40* over recalibration without windows and for window size *50* (see figure B.1-25) the improvement for these models is greater than for window size *40* but for both window sizes there do seem to also be regions where the predictions are becoming marginally worse for these 3 models. In general, for regions where recalibration without windows successfully eliminated bias, application of windowing seems to have just added more noise to the predictions resulting in steady decreases in the *PLR* plots, which become larger as the window size decreases. A similar pattern is seen for the non-parametric models since for most of these no bias was present in the recalibrated (without windows) predictors.

Table B.1-2.2 shows the significance levels of the meta-predictor applied with a number of different window sizes. It can be seen that all the resulting *u*-plots are good and only one of the *y*-plots (for *M10*) is significant at the 5% level. Figure B.1-26 shows that noise in the median predictions is more prolific for smaller window sizes than for larger ones. Comparison with figures B.1-4 and B.1-7 shows that, even for larger window sizes, there is more noise in the medians for the meta-predictors than for the parametric models but about the same as for the non-parametric models. Figure B.1-27 shows that apart from the jump in these plots[6] differences between the predictive accuracy resulting from the application of different window sizes are fairly marginal. Note, in particular, that *M1*, which is only based on knowledge of predictive accuracy from the

---

6 Here, some of these meta-predictors (e.g., *M20*, *M30*) have switched to predictors which behave favourably with the large inter-failure times, while some (e.g., *M10*, *M40*) have not.

previous prediction each time, performs as well as any other and there is a region in the second half of the data where both *M1* and *M2* seem to be marginally better than the rest.

Comparing figures B.1-28 with figures B.1-6, B.1-13, B.1-14, B.1-10 and B.1-18, shows that these meta-predictors are better than the majority of the original prediction systems and only marginally worse than the best of the original prediction systems.

8.1.1 Summary for *CISI1*

In general, it seems that some of the parametric models and most of the non-parametric models gave unbiased predictions on this data set and according to the *PLR* analyses this group of unbiased raw predictors were fairly close in accuracy, in spite of the non-parametric medians being more noisy than the parametric medians. It is interesting to note that the variants of the very simple non-parametric model, *OTL*, *OTL20* and *OTL50*, gave raw predictions which were comparable with those from the other models, and further, that the non-parametric models tend to be in much closer agreement than the parametric models (apart, perhaps, from *CM1*).

For those models which were initially biased (i.e., *DU*, *LV*, *KL* and *CM1*) improvement with respect to this bias could be achieved via the recalibration technique but in some cases (e.g., *LV* and *KL*) there was still room for further improvement, due to non-stationarity in the raw prediction errors. It was noted that the raw *y*-plots were not a good indication of this non-stationarity and thus of when we should expect improvement via recalibration and when we should not. In those cases where bias was still present after recalibration further improvement could be achieved through the application of the recalibration technique with windows but in such instances the improvement according to the *PLR* analyses over recalibration without windows was marginal and if the window size was too small then any improvement with respect to elimination of bias was outweighed by increase in noise in the predictions.

For those raw models which were initially unbiased there was generally little to be gained by recalibration although there was marginal fluctuation (favouring raw or recalibrated predictions over different intervals of data) in the *PLR* plots. For some models (*CM2* and *MO*) the recalibration technique seemed to result in biased predictions when the raw models were initially unbiased, but, again, only marginal fluctuation was seen in the *PLR* analyses. For those models for which recalibration without windows eliminated bias recalibration with windows seemed to make things worse with respect to noise, particularly for smaller window sizes, than recalibration without; the *PLR* analyses indicated that the application of windows in such cases had resulted in inferior predictive accuracy which worsened as the window size decreased.

For the meta-predictor it was seen that the predictive accuracy varied marginally with the window size with which it was applied but in general the resulting predictions were about as accurate as the best of the group of initial predictors over which selection was made, again, in spite of the apparently more noisy median predictions than were seen in some of the initial predictors. This noise in the median predictions increased as window size decreased but, surprisingly, window size *1* gave comparatively accurate predictions according to the *PLR* analyses.

8.2 Data set CISI2

For this data set the *JM, GO, MO, LM* and *LNHPP* models go to the limiting case of an *HPP* in the intervals $i \sim$ *27-40* and $i \sim$ *50-70*. Where this is not the case in the interval $i \sim$ *20-190* the *LM* model goes to the limiting case of *JM* while the *LNHPP* model fluctuates between the *GO* and the *MO*. Then from $i \sim$ *200-215* the *LNHPP* model goes to *MO*. Further, for $i \sim$ *215-244*, the *LM* and *LNHPP* models go to the limiting cases of the *JM* and *GO* models, respectively. From $i \sim$ *360-394* both these models go to the limiting case of the *MO* model. The frequent fluctuations between different models here clearly arises from the fact, noted in Chapter 7, that the trend in this data set is very irregular. Notice how the median predictions in figure B.2-4 respond to these changes in the original data. Although there is generally growth in these predictions when considered over the whole data, there are local fluctuations which show decay. In particular there is a large region over which the median predictions from all these models decrease which coincides with the clustering of failures previously noticed in Chapter 7, between the 320^{th} and the 380^{th} failures. Other regions for which the medians show local decay similarly coincide with clusters of failures previously identified in Chapter 7. Prior to each of these regions of decay the median predictions suddenly become larger where particularly large inter-failure times occur. As for the previous data set the predictions from these models are in great disagreement in the second half of the data.

From table B.2-2.1 it can be seen that all the raw parametric models have highly significant *u*- and *y*-plots for this data set. Figure B.2-5 indicates that the *JM, GO, MO, LM* and *LNHPP* models are, on average, giving very optimistic predictions while the *LV* and *KL* models are, on average, giving pessimistic predictions. The *DU* model appears to be resulting in pessimistic predictions for large inter-failure times. The *log(PLR)* plot (see figure B.2-6) indicates that the models are very different in their predictive capabilities on this data set. In detail *DU* seems to be marginally better than the other models from $i \sim$ *90-160*, while from *160-210 DU, LV* and *KL* are marginally better than the others, although the differences in predictive accuracy in this early data are fairly slight. From *210-300 DU* seems generally worse than the others. In the remaining part of the data set *DU, LV* and *KL* seem to be much better than the rest; here *LV* and *KL* are generally better than *DU*

and *JM* and *GO* are performing particularly badly. There is a great deal of switching in the comparative performance of the predictions from the various models and this is clearly caused by the fluctuations in the data. It can be seen, for example, that *DU*, *LV* and *KL* are giving more accurate predictions than the other models in those regions of the data previously identified in Chapter 7, where there is local reliability decay (e.g., from $i \sim 320\text{-}380$). In addition it can be seen from the jumps upwards in the *PLR* plots that the *DU* model and to a lesser extent the *LV* and *KL* models seem to cope particularly badly where there are particularly large inter-failure times.

The non-parametric models applied with windows switch between *HPP* and non-*HPP* predictions over many intervals throughout the data while the *OTL* model gives *HPP* predictions over the same region as did the parametric models. Comparison of figures B.2-7 and B.2-4 show that the medians for some of the non-parametric models seem to be noisier than those for the parametric models but are in much closer agreement over much of the data. In general, as for the parametric models, there are fluctuations between growth and decay in these predictions throughout the data. It seems that the non-parametric models are responding rather more quickly to the variations in the data than the parametric models. In particular, after a jump upward in response to large inter-failures they return more rapidly to smaller values as subsequent data is taken into account. The *CM1* median predictions appear to jump suddenly upward more frequently than the other non-parametric predictors, and also appear to be marginally more optimistic in other regions of the data (e.g., for the period of decay from $i \sim 320\text{-}380$). The *OTY50* model is giving a zero rate prediction (and so infinite median) for T_{226}, which, as we shall see later, has an impact on the *PLR* analysis.

From table B.2-2.1 it can be seen that, as for the parametric models, the non-parametric model *u*-plots are all highly significant, although the *CM1* model and the non-parametric models which are fitted with moving windows have insignificant *y*-plots while the raw parametric *y*-plots were all highly significant. Figure B.2-8 indicates that all the non-parametric models are giving optimistic predictions, particularly *CM1*. According to the *log(PLR)* plot (figure B.2-9), this optimism does not have the effect of the *CM1* model being much worse in predictive accuracy than the rest although it can be seen to be marginally worse towards the end of the data set. It can also be seen that the comparative model performances vary slightly over different intervals of data, although comparison with figure B.2-6 shows that there is not as much deviation in the comparative predictive performance as there was for the parametric models. After $i \sim 150$, though, *OTY20* and *OTL20* appear to be the best, followed closely by *OTL50* and *OTY50* (ignoring the jump

downwards for *OTY50*[7]) and then *OTL*. It can be seen that the level of windowing applied seems to be the over-riding factor affecting model performance; i.e., *OTY20* and *OTL20* are very similar and *OTL50* and *OTY50* are also very similar in predictive accuracy. Careful examination of the medians shows that these predictions for a given window size frequently coincide and, of course, this will occur whenever they are both giving *HPP* predictions. Notice though, how the smaller window size gives an advantage in those regions where the data is clustered. For example, in the region $i \sim 320\text{-}380$ it can be seen that accuracy in predictions seems to increase as the window size is decreased. This is not surprising since it is likely that smaller window sizes will allow a model to respond more accurately to changes in the trend in the data and the use of larger window sizes in the presence of local reliability decay is likely to result in more optimistic predictions, since these non-parametric models do not model decay.

Comparison of figures B.2-10 and B.2-6, shows that all of the non-parametric models are much better than most of the parametric models and some of them (*OTL20, OTL50* and *OTY20*) are generally better than the best of the parametric models (although, again, there is fluctuation in relative predictive accuracy over different regions of the data), while there is little to choose between most of them and the best parametric model. It is interesting to note that the application of the non-parametric models with a small window size seems to have bought them into comparable accuracy with *KL* in some of those regions were there is reliability decay in the data (e.g., $i \sim 320\text{-}380$) even though they do not model decay. In general we would expect that the application of models with small windows will give better results when there is much variation in the data. Clearly though, it is the recalibrated predictions that we are interested in for this data set, since, according to the *u*-plots, the predictions from all the raw models are biased.

Comparison of figures B.2-11 and B.2-4 show that the medians for the parametric models after recalibration are marginally closer together than previously but that they are still in great disagreement, particularly at the end of the data set. Most of the raw predictors have been adjusted for optimism, while *DU, LV* and *KL* have, later on in the data set, been adjusted for pessimism. Further comparison of these predictions shows that for the *DU* model there is non-stationarity in the errors in the predictions since right at the end of the data set there is adjustment for optimism. The recalibration median predictions, as the raw, tend to fluctuate with the changing trend in the raw data with local regions of reliability decay even though the general trend is toward growth.

---

7 This jump downwards is due to the zero rate prediction of T_{226} from the *OTY50* model.

From table B.2-2.1 we can see that for *JMS, GOS, MOS, LMS* and *LNHPPS* the *u*-plots are still significant at the 1% level. The significance levels of the *u*-plots for the remaining models have improved via recalibration although for *DUS* the plot is still significant at the 5% level. The recalibrated *u*-plots in figure B.2-12 indicate that many of the predictors are still highly optimistic after recalibration, suggesting the presence of non-stationarity in the departure of these raw prediction systems from the truth, although comparison with figure B.2-5 shows that most of these plots have improved. Figure B.2-13 indicates that recalibration generally, and sometimes dramatically, improves predictions later on in the data set (from $i \sim 160$) although there are some regions where it is not efficient; in fact sometimes recalibration seems to have resulted in *worse* accuracy than the raw, e.g., for *LV* and *KL* in the region $i \sim 320\text{-}380$. The improvement for *JM* and *GO* is very large in magnitude, but this is mainly because they were originally so bad (see figure B.2-6), and from B.2-14 we can see that these are still giving predictions which are much worse than the other recalibrated predictions. Notice the similarity of the shape of the *log(PLR)* plots in figures B.2-6 and B.2-14. Here, by noticing the change in the scales, we can see that recalibration has indeed resulted in the various predictions being closer in accuracy, but there is still a great deal of difference in accuracy between the recalibrated predictors in the second half of the data set with the "best" predictions switching between the various recalibrated predictors as the data evolves. All these various *log(PLR)* plots suggest that the best predictor, from amongst the raw and recalibrated parametric, might be *LVS* or *KLS* although, as previously stated, there are changes to the best predictor which we might select as the data evolves.

Figure B.2-15 confirms that there is indeed non-stationarity in the errors in the predictions from all of the raw parametric models, even for *LV* and *KL* which had good *u*-plots after recalibration. In general the raw predictors are not significantly biased in the early data, while later on they switch between being biased and not so over different intervals of data. It can be seen that the observations from the previous *u*-plots (figure B.2-5) over the whole data where really average statements - most of the optimism or pessimism indicated by these are almost certainly the result of later predictions. It can be seen how, over regions where the bias is great, there will not be sufficient adjustment via recalibration due to the fact that the *u*-plots incorporate previous less biased data.

Comparison of the raw and recalibrated median plots for the non-parametric models (see figures B.2-16 and B.2-7) shows how these raw predictions have been adjusted for optimism and how they are now in slightly closer agreement after recalibration than before. Again many of them appear to be more noisy than the recalibrated parametric medians (see figure B.2-11) and this is to be expected since the noise in the raw non-parametric predictions will be carried through to the recalibrated; the zero rate prediction for *OTY50* will also not be eliminated by recalibration. From table

B.2-2.1 it can be seen that for the non-parametric models there is improvement in some of the *u*-plots after recalibration but most of them are still significant at the 5% level. This indicates that, as for the parametric models, there is non-stationarity in the raw prediction errors. In spite of this non-stationarity, comparison of figure B.2-17 with figure B.2-8 shows that all the *u*-plots have been improved by recalibration. Figure B.2-18 shows that there is a pretty steady improvement after recalibration for all the non-parametric models in some regions of the data while in others there is little to choose between the raw and the recalibrated predictions. The large improvement, particularly in the region $i \sim 320$-*380*, for *CM1*, is due to the fact that it was very optimistic in the first place. The *log(PLR)* plot in figure B.2-19 shows that the *OTL20S, OTL50S, CM1S* and *OTY20S* are generally better than the other recalibrated non-parametric predictors and comparison with figure B.2-9 suggests that the recalibrated non-parametric predictions are marginally closer in accuracy than were the raw. Comparison of figures B.2-20 and B.2-14 shows the best of the recalibrated non-parametric predictors to be generally better than the best of the recalibrated parametric predictors, although, again, there are some regions where they are worse.

Figure B.2-21 shows that, as suggested by the significant *u*-plots after recalibration, there is, indeed, non-stationarity in the raw non-parametric prediction errors. Again, there is less error in the early data and in the later data the departures switch between significant optimism and not so. Here we can see how recalibration using *all* available previous predictions will again give insufficient adjustment for optimism in certain regions of data, while in other regions the predictions are likely to be adjusted too much.

It is clear that, for this data set, due to the variation in the original data, we have much variation in the prediction errors for all the raw models. This results in recalibration using all previous predictions giving improvement but leaving scope for further improvement. This is clearly a candidate for the application of recalibration with a moving window across the raw predictions. From table B.2-2.1 we can see that application of this technique to the raw predictors improves all the *u*-plots when compared to recalibration without windows. For the non-parametric models and for *DU*, *LV* and *KL* all the *u*-plots for all recalibration window sizes are now insignificant at the 20% level. For the remaining parametric models some of the *u*-plots remain significant at the 20% level but all are now insignificant at the 5% level.

Again we have to assess that the decrease in bias bought by this technique is not countered by an increase in noise. Comparing figures B.2-22, B.2-23 and B.2-11, we can see that the median predictions do appear to be a little more noisy than those from recalibration without windows and this noise increases slightly as the window size

decreases. The median predictions are now in very close agreement (becoming closer as the window size decreases) where for recalibration without windows there was great disagreement in the second half of the data. Observation of figures B.2-24 and B.2-25 show that over some regions of the data, for some of the models, there are improvements in the predictions from the application of windows. Notice, in particular, the huge improvement shown for the *JM* and *GO* models in the previously identified region of reliability decay (i.e., $i \sim 320\text{-}380$). Here the application of recalibration with windows appears to have successfully eliminated optimism the raw *JM* and *GO* predictions while recalibration without windows did not make a sufficient adjustment for this optimism. The large improvement shown for *DU* is mainly due to jumps upwards in the *log(PLR)* which are coincident with extremely large inter-failure times rather than a steady improvement and these jumps get larger as the window size used for recalibration gets smaller. In some regions of the data, however, the predictions from recalibration with windows are worse according to this *log(PLR)* analysis, than for recalibration without windows. In some cases this is probably due to increased noise arising from the application of windows which appears to out-weigh any decrease in bias bought by this technique, and this problem appears to get worse as the window size is decreased. For example, in the early data for window size *20* the windowing has degraded the performance of the recalibrator while for *50* there is little to choose between recalibration with or without windows. Comparison of figures B.2-26 and B.2-27 with figure B.2-14 show that, apart from the large drops downward for window size *20*[8], the application of windowing has made these predictors closer in accuracy.

For the non-parametric models the median predictions after recalibration with windows do not look much more noisy than the recalibrated without windows (compare figures B.2-28 and B.2-29 with B.2-16) but are in slightly closer agreement than before, and more so as the window size decreases. Recalibration using windows has generally made things worse than recalibration without and more so with smaller windows (see, for example, B.2-30 and B.2-31) in spite of the insignificant *u*-plots for all the window sizes. There are some regions where improvements can be seen and it is presumably elimination of optimism in these regions which is resulting in better *u*-plots. The application of recalibration with windows has resulted in predictors which are marginally

---

8 These drops are actually due to the improvement of *DUS20* over *DUS* coincident with extremely large inter-failure times.

closer in accuracy than those from recalibration without windows and these become closer as the window size decreases (compare figures B.2-32 and B.2-33 with B.2-19)[9].

Again we now have a large number of different predictors from which to choose and the relative performance of these different predictors frequently varies over different regions of the data. This is clearly a result of fluctuations in the trend in the raw data. It is circumstances such as these that we might expect our meta-predictor to give better predictions than if we were to choose any particular single predictor.

Table B.2-2.2 shows that all the *y*- and *u*-plots for all windows sizes (i.e., *M1*, *M2*, ... , *M50*) are insignificant at the 10% level. The median predictions (figure B.2-34) appear to be quite noisy, and more so for the smaller window sizes. The *log(PLR)* plot (figure B.2-35) shows that, apart from the large drop for *M1*[10], the different window sizes result in fairly close predictive accuracy and this is in spite of the marginally noisier predictions for the smaller window sizes; in fact, the larger window sizes seem to generally result in *worse* accuracy than the smaller window sizes in the latter half of the data. Notice how, over this region, *M1*, which is based only on the previous prediction each time, gives quite good predictions when compared with the other window sizes. The better performance of the small window sizes here is probably due to the frequent local fluctuations in the relative accuracy of the various initial predictors as the data evolves. This ability for the smaller window sizes to result in a meta-preditor which more quickly switches to the most recently best single predictor appears to marginally out-weigh any increase in noise which results when using a small window size, for this particular data set.

Comparing figure B.2-36 with figures B.2-14, B.2-13, B.2-20 and B.2-18, shows that some of these meta-predictors, e.g., *M5*, are as good as the best of the initial single predictors, and that most of the meta-predictors are dramatically better than many of the initial single predictors.

---

9 The large jumps down shown for *OTY50S20* and *OTY50S50* are again a result of the zero rate prediction given by *OTY50* at this point, which cannot be eliminated by the application of recalibration with or without windows.

10 The *M1* model chooses *OTY50S50* at this point and so gives a zero rate prediction.

8.2.1 Summary for *CISI2*

For this data set, in contrast to *CISI1* where most of the initial raw predictors were unbiased, *all* of the raw models gave predictions which were on average *grossly in error*. The nature of the bias in the predictions varied depending on the model, with all except the *DU*, *LV* and *KL* models giving generally very optimistic predictions, particularly *JM*, *GO* and *CM1*. The non-parametric models, which were again in closer agreement, although they exhibited more noise gave predictions which were better than all of the parametric models, subject to some local variation. Evidence suggested that the application of the non-parametric models with small windows helped them to better capture the frequent fluctuations in the trend in the data. It is interesting to note that, again, variants of the simple non-parametric predictor *OTL20* and *OTL50*, performed as well as all of the others.

General improvement (reduction in the bias) for all the raw predictors could be achieved by the application of the recalibration technique with more improvement shown for those raw predictors which were initially worst. The best of the recalibrated non-parametric predictors were better than the best of the recalibrated parametric predictors, again subject to some local variation. However, for many of the predictors bias, the nature of which again depended on the initial raw model, was still present even after recalibration. On further investigation it was found that there was non-stationarity in the prediction errors for most of the raw models. Application of recalibration with windows was shown to give improvement (sometimes dramatic) in regions of the data where bias was still present after applying recalibration without windows. In other regions of the data applying windows seemed to make things marginally worse than recalibration without windows, and often unwanted noise was added to the predictions, which tended to increase as the window size used for recalibration decreased.

For the meta-predictor performance varied marginally with the window size with which it was applied. Recall that in general there was quite a lot of local variation in the relative performance of the initial predictors. The result of this is that application of the meta-predictor with smaller window sizes was able to better capture these local variations, and in particular *M1* again performed fairly well compared with the larger window sizes. Most importantly, though, for this data set, where there was clearly no single initial predictor (raw or recalibrated) which could be chosen as consistently best throughout the data, the meta-predictor resulted in predictions which were as good as the "best" (on average) of the initial predictors.

8.3 Data set USBAR

The *JM, GO, MO, LM* and *LNHPP* models go to the limiting case of an *HPP* for *i ≈ 20-87, 110-200.* For *i ≈ 88-109* and *i ≈ 200-300* the *LM* and *LNHPP* models go to the *JM* and *GO* models respectively. These regions were the predictions from different models coincide can be seen from the median plot (see figure B.4-4). From this plot we can see that there is reliability growth in the raw predictions after the sudden decrease in the failure rate at the 260^{th} failure, previously identified in Chapter 7. After this point there are differences between the median predictions and this discrepancy gets larger as the data proceeds.

From table B.4-2.1 it can be seen that the *u*- and *y*-plots for all the raw parametric models are highly significant. The *u*-plots (figure B.4-5) indicate that the *JM* and *GO* models are giving very optimistic predictions, the *LM* and *LNHPP* models are giving predictions which are optimistic except for large inter-failure times and the *LV* and *KL* models are giving very pessimistic predictions. The *S*-shaped *u*-plots for the *MO* and *DU* indicate that their predictions are optimistic for small inter-failure times and pessimistic for large inter-failure times and that the true medians probably lie closest to the predictions given by these two models.

Figure B.4-6 suggests that the raw *DU* model is performing relatively badly compared with the other raw parametric models for this data set, except at the end of the data where the *JM* and *GO* models are marginally worse than *DU*. It is clear from this plot that there is much variability in the models' predictive accuracy from the 260^{th} failure onwards. The *log(PLR)* plots for the models jump upwards at the occurrence of the large inter-failure times in the raw data directly after this point and this is clearly where the *DU* model, and to a lesser extent the *MO* model, experiences particular difficulty. The *KL* and *LV* models seem to be performing steadily better than the other *6* raw parametric models throughout the data.

For the non-parametric models with windows of size *20*, there are fluctuations between *HPP* and non-*HPP* predictions throughout the data and these tend to coincide for both *OTY20* and *OTL20*. The same is also true for window size *50* but these fluctuations are less frequent. The *OTL* model gives *HPP* predictions over the same regions as the parametric models. The median predictions for the non-parametric models (see figure B.4-7) accordingly frequently coincide and reliability growth in these medians is present after the 260^{th} failure. Comparing these median predictions with those given by the raw parametric models (compare with figure B.4-4) we can see that the raw non-parametric medians are in much closer agreement than the parametric, although they are more noisy. There is a tendency for the non-parametric median predictions to jump suddenly upward

in response to the occurrence of particularly large inter-failure times and then to return rapidly to lower values as subsequent, less extreme, data is taken into account.

From table B.4-2.1 we can see that, as for the parametric models, the non-parametric models also have highly significant *u*-plots, although the *y*-plots are all insignificant at the 5% level, whereas for the parametric models they were all highly significant. Figure B.4-8 indicates that these models are giving optimistic predictions except for large inter-failure times, particularly *CM1*.

In spite of this, according to the *PLR* analysis (see figure B.4-9) the *CM1* model is comparable in accuracy with the other non-parametric models; there is no one non-parametric model which is performing steadily better than any other on this data set, although there is some local fluctuation. Comparing this with the *log(PLR)* plots in figure B.4-6 we can see the non-parametric models are much closer in predictive accuracy than the parametric models (notice the difference in the scale of these two plots). Comparison of figures B.4-10 and B.4-6 shows that the non-parametric models are much better than some of the parametric models, although not as good as the best of the parametric models subject to some local fluctuation; there are regions where they giving more accurate predictions than *KL*, although over most of the data they are worse. The best single raw predictor is thus generally *KL*, for this data set, but the highly significant *u*-plots indicate that all these models are giving inaccurate predictions and so, for this data set, we are clearly interested in any improvement to be gained by recalibration.

Comparison of figures B.4-11 and B.4-4 shows that the median predictions after recalibration are in closer agreement although they still disagree a great deal after the 260^{th} failure. Some of the raw median predictions (e.g., *JM* and *GO*) have been adjusted for optimism while other (e.g., *LV* and *KL*) have been adjusted for pessimism; notice how the *DU* median predictions have been virtually unaltered by recalibration. Table B.4-2.1 shows that most of the *u*-plots for the recalibrated parametric models are still significant at the 5% level, although comparison of figures B.4-12 with B.4-5 indicates that recalibration has drastically reduced the bias originally present in all the raw parametric model predictions. A more detailed examination of these *u*-plots suggests that recalibration has generally, on average, made an insufficient adjustment for the bias originally present in the raw predictions.

Figure B.4-13 shows how the recalibration procedure gives fairly steady improvement in predictive accuracy for all the parametric models. For the *DU* model this improvement is particularly dramatic: steady increases in the *log(PLR)* for *DUS* versus *DU* can be seen and there are also large jumps upward in this plot, which are coincident with particularly large inter-failure times in the raw data (see Chapter 7). Figure B.4-14

indicates that the best recalibrated parametric prediction system is *DUS*; *MOS* also seems to be performing quite well apart from the large drops at various stages which again coincide with particularly large inter-failure times. It is interesting to note that recalibration gave improvement with respect to these extreme data values for the *DU* model but that no such improvement was seen for the *MO* model. Comparison of this figure with figure B.4-6 shows that recalibration has resulted in closer predictive accuracy from the different models, although there is still quite a lot of disagreement. We can also see how the model which was initially much worse than the others, i.e., *DU*, after recalibration now gives the best predictions.

It is also interesting to observe that recalibration would appear to be making some quite sophisticated adjustments in order to eliminate errors in the *DU* and *MO* predictions. The medians are virtually unchanged by recalibration while at either ends of the scale (i.e., for large and small inter-failure times) the adjustment being made is, according to figure B.4-5, in opposite directions. Actually this is probably an over-simplification of the situation, since the presence of significant *u*-plots after recalibration for most of the parametric models indicates that there is non-stationarity in the errors in the raw model predictions. Figure B.4-15 confirms this, and here we see how a *u*-plot based on all previous predictions may not give the right adjustment via recalibration at many stages. For example, for *JM* and *GO*, most of the optimism in the raw predictions appears to be towards the end of the data set, while the *u*-plot used for recalibrating these predictions will include earlier, less biased, predictions.

For the non-parametric models comparison of figures B.4-16 and B.4-7 shows how recalibration has made adjustments for optimism in the raw median predictions. Similar growth and noise in seen in the medians after recalibration to before and the recalibrated predictions are in marginally closer agreement. From table B.4-2.1 we can see that most of the *u*-plots after recalibration of the non-parametric models are insignificant at the 5% level and comparison of figures B.4-17 and B.4-8 suggests that the optimism in the original raw predictions has been drastically reduced.

Figure B.4-18 shows that recalibration has given fairly steady improvement for these non-parametric models throughout the data set. Comparison of figure B.4-19 with B.4-9 shows that the non-parametric predictors are marginally closer in accuracy after recalibration than before and that there is no one recalibrated non-parametric predictor that is steadily better than the others, although there is some local fluctuation. Comparison of figures B.4-19 and B.4-14 shows that the recalibrated non-parametric predictors are much closer in accuracy than the recalibrated parametric predictors. Figure B.4-20 shows that, in spite of the fact that many of the recalibrated non-parametric predictors have marginally better *u*-plots, the best recalibrated parametric predictor, *DUS*, is better than

these predictors over much of the data set suggesting that the noise in the non-parametric model predictions may be causing problems with predictive accuracy. Comparison with figure B.4-14, though, suggests that some of the recalibrated non-parametric predictors are as accurate as the other *7* recalibrated parametric predictors.

Figure B.4-21 shows how there are signs of non-stationarity in the raw non-parametric prediction errors with optimism increasing towards the end of the data set but this non-stationarity in prediction errors is not as pronounced as it is for some of the raw parametric predictors (compare with figure B.4-15).

From table B.4-2.1 we can see that the *u*-plots for *all* the parametric and non-parametric models recalibrated with *all* window sizes are insignificant at the 20% level and many of the *y*-plots have also improved. For the parametric models the median predictions for these predictors are much closer in agreement than was the case with recalibration without windows (see, for example, figures B.4-22 and B.4-23 and compare these with figure B.4-11) although slightly more noisy. For some of the parametric models the application of recalibration with windows gives large improvements in predictive accuracy over recalibration without windows in some regions of the data. For example, from figures B.4-24 and B.4-25 we see large improvement for *JM* and *GO* at the end of the data set, particularly for window size *50*. Comparing, again, the median predictions suggests that this improvement is due to recalibration with windows making sufficient adjustments for optimism in the raw predictions in this region, while for recalibration without windows this adjustment was insufficient. Returning to the *PLR* analysis we can see jumps upwards in some of these plots again coincident with the particularly large inter-failure times just after the 260^{th} failure; it seems that the application of recalibration with windows can sometimes give more improvement at extreme values than recalibration without windows.

From figure B.4-24 and B.4-25 we can also see some regions where there are steady decreases which indicates that any decrease in bias which may have been bought by using windows has been out-weighed by an increase in noise, and this problem is worsens as the window size decreases. This is also the case for all the non-parametric models recalibrated with all the window sizes; the median predictions are marginally less optimistic than those for recalibration without windows (compare, for example, figure B.4-26 with B.4-16) but steady decreases can be seen in the *log(PLR)* which become more pronounced as the window size decreases (see, for example, B.4-27 and B.4-28).

Table B.4-2.2 shows that the *u*- and *y*-plots for the meta-predictors are all insignificant at the 5% level, while for the larger window sizes, *20*, *30*, *40* and *50*, they are all insignificant at the 20% level. Figure B.4-29 shows how median predictions for

the smaller window sizes tend to be more noisy than for larger window sizes. The *log(PLR)* plot in figure B.4-30 shows that there is quite a marked variation in the predictive accuracy according to the window size used for the meta-predictor. The meta-predictors with smaller window sizes, *M1*, *M2* and *M5* are steadily worse than the others and *M30*, *M40* and *M50* are steadily better in the second half of the data.

Comparison of figures B.4-31 with figures B.4-13, B.4-14, B.4-18 and B.4-20 show that, apart from the jumps downward seen in figure B.4-31[11], these two meta-predictors are about the same in predictive accuracy as the best of the original predictors.

8.3.1 Summary for *USBAR*

For this data set, as for *CISI2*, *all* of the raw models gave predictions which were on average *grossly in error*. The nature of the bias in the predictions again varied depending on the model, with all except the *MO*, *DU*, *LV* and *KL* models giving on average very optimistic predictions, particularly *JM*, *GO* and *CM1*. In general the bias in most of the raw predictors tended to increase as the data evolved. The non-parametric models, which were again in closer agreement, although they exhibited more noise gave predictions which were better than many of the parametric models, although not as good as the best of the parametric models, subject to some local variation. Again, it is interesting to note that the simple non-parametric predictor *OTL*, *OTL20* and *OTL50*, performed as well as all of the other non-parametric predictors.

Steady improvement (reduction in the bias) for all the raw predictors could be achieved by the application of the recalibration technique with more improvement shown for those raw predictors which were initially worst. In fact, the predictor which was initially the worst before recalibration, *DU*, turned out to be the best after recalibration. The best of the recalibrated non-parametric predictors were as good as many of the recalibrated parametric predictors, but not as good as the best of the recalibrated parametric predictors, again subject to some local variation.

For some of the predictors, in particular for *JMS* and *GOS*, bias was still present even after recalibration. On further investigation it was found that there was some non-stationarity in the prediction errors for these raw models. Application of recalibration with windows was shown to give improvement (sometimes dramatic) in regions of the data where gross bias was still present after applying recalibration without windows. In other

---

11 Here the meta-predictors are switching to a predictor which is marginally worse than *DUS* at the large inter-failure time, t_{340}.

regions of the data and for most of the other models applying windows seemed to make things marginally worse than recalibration without windows, and often unwanted noise was added to the predictions, which tended to increase as the window size used for recalibration decreased. For both recalibration with and without windows improvement with respect to problems encountered by the raw predictors due to the occurrence of particularly large inter-failure time data could be seen, indicated by large jumps in the *log(PLR)* plots coincident with these extreme data points.

For this data set, performance of the meta-predictor varied more with the window size with which it was applied than for the previous two data sets. Also, in contrast to the previous 2 data sets, the predictors with smaller window sizes, which resulted in more noisy median predictions, performed worse than those with larger window sizes. The best of the meta-predictors gave predictions which were as good as the "best" (on average) of the initial predictors.

8.4 Data set USROFF

The *JM*, *GO*, *MO*, *LM* and *LNHPP* models tend to the limiting case of an *HPP* very rarely (only within the region $i \sim 20\text{-}34$). The *LM* model often goes to the limiting case of the *JM* model previous to $i \sim 104$, while the *LNHPP* model fluctuates between *MO* and *GO*, but mostly *MO*. The median predictions (see figure B.5-4) from these models generally exhibit significant growth after the 85^{th} failure and from this point there is disagreement in the median predictions which generally becomes greater as the data evolves. The point at which the medians start to disagree coincides with the previously identified (see Chapter 7) change-point in the data just after the 85^{th} failure and this disagreement becomes even more pronounced from the 100^{th} failure where there are some particularly large inter-failure times, t_{100} and t_{101}. In fact, in response to these large inter-failure times the *JM* and *LM* models both predict that there are no bugs left in the program at this point and hence give zero rates and infinite medians (observe the peak at $i = 102$ in figure B.5-4 for these two models). From table B.5-2.1 we see that, with the exception of the *MO* model, all the *u*-plots for the parametric models are significant at the 5% level and figure B.5-5 suggests that the *JM* and *GO* predictions are very optimistic and to a lesser extent this is also the case for *LM* and *LNHPP*, while the *LV* and *KL* predictions are very pessimistic and *DU* is pessimistic for large inter-failure times.

The *log(PLR)* plot for the raw parametric models (see figure B.5-6) indicates that, apart from the large drop downwards for LM^{12}, the *MO*, *LM* and *LNHPP* models outperform the other models for this data set (although the difference between these and some of the others is marginal). It is interesting to note that the *MO* model is not significantly better than the other raw predictors according to this *PLR* analysis, in spite of its' having the best *u*-plot. Notice how the *JM* and *GO* models perform particularly badly in the second half of the data set where the growth (considered globally) is non-linear. The *LV* and *KL* models perform badly right at the end of the data set. As we shall see later, it is in these regions that the raw predictions from these two groups of models are particularly biased. The bad performance of the *DU* model is mainly due to some jumps in this plot coincident with extreme data values relatively large inter-failure times.

The non-parametric models fitted with windows of size 20 fluctuate frequently between *HPP* and non-*HPP* predictions throughout the data set, whereas for window size *50* there a very few *HPP* predictions and for *OTL HPP* predictions coincide with those from the parametric models (i.e., only within the region $i \sim 20\text{-}34$). Comparing the medians from these models (see figure B.5-7) with those from the parametric models (see figure B.5-4) we can see that, for the non-parametric models, these predictions are in much closer agreement and generally a little more noisy, than for the parametric models. Growth is also exhibited by these predictions after the 85^{th} failure. The noise in the median predictions for the non-parametric models is mainly due to sudden increases in these predictions in response to the occurrence of large inter-failure time data (e.g., t_{88}, t_{100}, t_{101}); as was observed for *USBAR*, the predictions tend to return more quickly than some of the parametric median predictions, to lower values as subsequent, less extreme, data is taken into account. From table B.5-2.1 it can be seen that the *u*-plots for some of the raw non-parametric models are insignificant at the 5% level and the *y*-plots are all insignificant at the 20% level whereas the *y*-plots for many of the parametric models are significant. Figure B.5-8 indicates that the *CM1* model, in particular, is giving optimistic predictions (although, from figure B.5-5, not as optimistic as *JM* and *GO* nor as biased as *LV* and *KL*), while for the other non-parametric models it would appear that predictions of smaller inter-failure times, are optimistic.

From figure B.5-9 we can see that the non-parametric models with significant *u*-plots, *CM1*, *OTL*, *OTL50* and *OTY50*, are marginally worse than the other non-parametric models but comparing with figure B.5-6 we can see how the predictions from

---

12 The large jumps downward for the *JM* and *LM* models are due to the zero rate predictions for T_{102} from these models.

the non-parametric models are much closer in accuracy than the predictions from the parametric models. From figure B.5-10 we see that the raw non-parametric models are marginally worse than the better of the raw parametric models after the 85^{th} failure; the jumps are again coincident with extreme data values.

Comparing figure B.5-11 with figure B.5-4 we can see that the recalibrated medians for the parametric models are in closer agreement than the raw but there is still disagreement in the predictions after the 85^{th} failure with *JMS*, *GOS*, *LVS* and *KLS* some distance from the other median predictions. The predictions from *JM* and *GO*, and to a lesser extent from *LM* and *LNHPP*, have been adjusted for optimism and those from *DU*, *LV* and *KL* have been adjusted for pessimism; for *MO* the predictions are altered very little by recalibration. Table B.5-2.1 shows that the *u*-plots for *MOS*, *DUS*, *LMS* and *LNHPPS* are insignificant at the 20% level while the other four predictors still have significant *u*-plots after recalibration. Comparison of figure B.5-12 with figure B.5-5 shows that although there is improvement in the *u*-plots after recalibration *JMS* and *GOS* are still optimistic while *LVS* and *KLS* are still pessimistic. Notice that the raw *y*-plots here are not a good indication of the efficiency of recalibration since from these one would expect *LVS* and *KLS* to have the best *u*-plots and not expect good *u*-plots for *LMS* and *LNHPPS*.

Figure B.5-13 indicates improvement via recalibration for most of the models after the 85^{th} failure, while early on in the data recalibration makes things marginally worse. The *PLR* analysis in figure B.5-14[13] indicates that *JMS* and *GOS* are steadily worse than the recalibrated versions of the other models after the 85^{th} failure and *LVS* and *KLS* are also worse in a region at the end of the data. For the other models performance of the recalibrated predictors is about the same - *MOS* is marginally better than *DUS*, *LMS* and *LNHPPS*; *DUS* is worse again mainly due to jumps coincident with extreme data although recalibration has gone some way in reducing these jumps. Comparison with figure B.5-6 shows that the models are closer in predictive accuracy after recalibration than they were before recalibration.

The presence of significant *u*-plots after recalibration indicates non-stationarity in the raw prediction errors for *JM*, *GO*, *LV* and *KL* and figure B.5-15 confirms this. We can see how insufficient adjustments will be made by recalibration for these 4 models since the extreme bias in the raw predictions given by these models is only present in the later data; for the *JM* and *GO* models the optimism is in the second half of the data while

---

13 The large jumps downwards for *JMS* and *LMS* are again due to the zero rate predictions which cannot be eliminated by recalibration.

for *LV* and *KL* the pessimism is just at the end of the data. Comparison of figure B.5-15 with figures B.5-13 and B.5-14 shows how there is improvement via recalibration, but not enough, precisely in those regions where bias is present for these four models. We can also see that recalibration makes things slightly worse when there is no bias in the raw model predictions (at the beginning of the data set for all the parametric models and at the end for *MOS*, *LMS* and *LNHPPS*); in such circumstances recalibration will either just add unwanted noise, or make an inappropriate adjustment due to bias in earlier predictions.

When recalibration is applied to the non-parametric models the resulting median predictions (see figure B.5-16) are marginally closer than before (compare with figure B.5-7) with slight adjustments for optimism throughout the data and the noise in the original predictions is retained in the recalibrated predictions. From table B.5-2.1 it can be seen that all the *u*-plots for the recalibrated non-parametric predictors are insignificant at the 20% level. In this case this *does* fit in with the significance levels for the *y*-plots for the raw models; these are all insignificant at the 20% level suggesting that recalibration is likely to be efficient. Figure B.5-17 shows that over much of the data there is improvement via recalibration for all the non-parametric models except at the beginning and at the end of the data set where there appears to be a degradation in predictive performance. From figure B.5-18 we can see that after recalibration these models are close in predictive accuracy, as were the raw non-parametric models (compare with figure B.5-9) and closer in accuracy than the recalibrated parametric predictors (compare with figure B.5-14). From figure B.5-19 we can see that they are also close, but marginally worse, than the best of the recalibrated parametric predictors, *MOS*.

From figure B.5-20 we can see that there is non-stationarity in the error in the raw predictions for the non-parametric models but comparison with figure B.5-15 shows that this is not so pronounced as for some of the raw parametric models. Comparison of figure B.5-20 with figure B.5-17 suggests that recalibration make things worse in regions where the raw non-parametric models are not biased; at the end of the data set it is likely that an adjustment will be made for optimism when the raw models have ceased to be optimistic. Notice also from figure B.5-20, how the *CM1* model is fairly consistently more optimistic than the others throughout the data set.

From table B.5-2.1 it can be seen that when recalibration is applied with moving windows, for the parametric models most of the resulting *u*-plots are insignificant at the 20% level while for the non-parametric models all the resulting *u*-plots are insignificant at the 20% level. For those parametric models for which the *u*-plots were significant after recalibration without windows there is a tendency for the *u*-plots to become worse as the window size for recalibration increases, indicating that recalibration with the larger

window sizes does not make a sufficient adjustment for the bias in the raw models. The resulting median predictions are in closer agreement than those from recalibration without windows and become even closer, but marginally more noisy, as the window size used for recalibration decreases (see, for example, figures B.5-21 and B.5-22 and compare with figure B.5-11). The predictive accuracy also becomes closer as the window size is decreased (see for example figures B.5-23 and B.5-24[14] and compare with figure B.5-14).

From figures B.5-25 and B.5-26 it can be seen that, in those regions where it was identified that the recalibration without windows made an insufficient adjustment for bias in the *JM*, *GO*, *LV* and *KL* models, a marked improvement can be seen via the application of windowing. Otherwise the application of windowing has made things marginally (the scale is small) worse, and more so for smaller window sizes, than recalibration without windows, presumably due to increased noise in the predictions. A similar pattern is seen for the non-parametric models for which recalibration without windows had already effectively eliminated the bias in the raw predictions. For example, from figures B.5-27 and B.5-28, apart from jumps upwards for window size *20*[15] for some of these models, the application of windows results in marginally worse predictive accuracy and more so for smaller window sizes.

From table B.5-2.2 it can be seen that the meta-predictor gives good *y*- and *u*-plots for all window sizes, in fact the *u*-plots are all insignificant at the 20% level. The median predictions in figure B.5-29 are quite noisy for the smaller window sizes and also disagree at various points. According to the *PLR* analysis in figure B.5-30, after the 100^{th} failure *M1* and *M2* are marginally worse than the meta-predictors with bigger window sizes, but, apart from the jump downwards for *M10*[16], in general they are fairly close in accuracy.

---

14 The large jumps downwards for the various recalibrated versions of *JM* and *LM* are again due to the zero rate predictions which cannot be eliminated by recalibration with or without windows. The large jumps downwards for window size *20* indicate that application of recalibration with this window size has improved things for *DU* with respect to the large inter-failure time t_{88}.

15 Again, these jumps are due to improvement with respect to the large inter-failure time, t_{88}.

16 This jump downwards for *M10* is due to the fact that it has switched to the *JM* or *LM* models (or one of their recalibrated versions) coinciding with the stage at which they give a zero rate prediction.

Comparison of figure B.5-31 with figures B.5-13, B.5-14, B.5-19 and B.5-17 show that, according to the *PLR* analyses, these meta-predictors are marginally worse than the best of the original predictors but as good as or better than many of the original predictors.

8.4.1 Summary for *USROFF*

For this data set all the raw parametric models except *MO*, gave predictions which were on average in error. For the non-parametric models the OTL models and CM1 also gave predictions which were on avergae in error. The nature and extent of the bias in these predictions again varied depending on the model. All except *MO*, *LV* and *KL* gave optimistic predictions, particularly *JM* and *GO* and *LV* and *KL* were equally pessimistic. *CM1* gave predictions which were more optimistic than the other non-parametric models but not as optimistic as *JM* and *GO*. In general disagreement between the raw parametric predictors tended to increase as the data evolved and growth became more rapid. The non-parametric models, which were in closer agreement, although they exhibited more noise gave predictions which were comparable with some of the parametric models, although not as good as the best of the parametric models. Again, the simple non-parametric predictor *OTL*, *OTL20* and *OTL50*, performed as well as the other non-parametric predictors.

On average improvement (reduction in the bias) for all the raw predictors could be achieved by the application of the recalibration technique with more improvement shown for those raw predictors which were initially worst, although there were some regions where recalibration resulted in less accurate predictions. The best of the recalibrated non-parametric predictors were as good as some of the recalibrated parametric predictors, but not as good as the best of the recalibrated parametric predictors.

For some of the predictors, *JMS*, *GOS*, *LVS* and *KLS*, bias was still present even after recalibration. On further investigation it was found that there was non-stationarity in the prediction errors for these raw models with most of the bias in the predictions toward the end of the data. Application of recalibration with windows was shown to give improvement in regions of the data where gross bias was still present after applying recalibration without windows. In other regions of the data and for most of the other models applying windows seemed to make things marginally worse than recalibration without windows, and often unwanted noise was added to the predictions, which tended to increase as the window size used for recalibration decreased.

For this data set, performance of the meta-predictor varied marginally with window size, with predictions from *M1* and *M2*, which gave more noisy median

predictions, slightly worse than those from the larger window sizes. These meta-predictors gave predictions which were better than many of the initial predictors but marginally worse than the "best" (on average) of the initial predictors.

8.5 Data set USPSCL

As for the data set *USROFF* for *USPSCL* the parametric models rarely tend to the limiting case of an *HPP*. In the region $i \sim 20\text{-}65$ the *LM* model tends to the limiting case of the *JM*, while the *LNHPP* fluctuates between *MO* and *GO*. Later on in the data set ($i \sim 94\text{-}104$ for the *LM* and $i \sim 80\text{-}84,90\text{-}104$ for the *LNHPP*) both these models tend to the limiting case of the *MO*. The raw median predictions from the parametric models in figure B.6-4 exhibit slight reliability growth and disagreement throughout the data with the more optimistic predictions more noisy than for the others. From table B.6-2.1 it can be seen that all the parametric models except *LV* and *KL* have significant *u*-plots. Figure B.6-5 suggests that the *JM*, *GO*, *MO*, *LM* and *LNHPP* models are optimistic, except for large inter-failure times and the *DU* model is also optimistic but to a lesser extent than the others. Figure B.6-6 suggests that the most accurate predictions are indeed from the *LV* and *KL* models, the *JM* and *GO* models are particularly inaccurate from failure *85* onwards and the other four models are about the same in predictive accuracy.

For the raw non-parametric models applied with window size *20* we see rapid fluctuations between *HPP* and non-*HPP* solutions throughout most of the data. For window size *50* there are no *HPP* solutions and *OTL* only results in *HPP* solutions very rarely. From figure B.6-7 we can see that *CM1* is generally giving more optimistic median predictions than the other non-parametric models. The predictions from the remaining non-parametric models are generally very close and much closer than those from the raw parametric models (compare with figure B.6-4). As with the parametric models there is slight growth in these predictions. Some of the these models (*CM1* and *OTL*) are more noisy than the parametric models. This is again due to jumps upwards in response to large inter-failure times (for example t_{84}). The predictions again return more quickly to lower values, as subsequent less extreme data occurs, than the parametric medians which jump up similarly. The *u*-plots for the raw non-parametric models are all significant at the 1% level (see table B.6-2.1) and figure B.6-8 suggests that these models, particularly *CM1*, are giving optimistic predictions except for large inter-failure times; comparison with figure B.6-5 suggests that these predictions are marginally less optimistic than the parametric models with the most bias.

Figure B.6-9 suggests that *CM2*, *CM3*, *OTY20*, and *OTY50* are about the same in predictive accuracy. *OTL*, *OTL20* and *OTL50* are worse mainly due to a jumps downwards just after t_{84}, while *CM1* is steadily worse than the others. Comparison with

figure B.6-6 shows that the raw non-parametric models are closer in accuracy than the raw parametric models (note the change in the scale). Figure B.6-10 suggests that the non-parametric models are generally worse than *KL*, the best of the parametric models, except towards the end of the data, where there is little to choose. Comparison with figure B.6-6 suggests that the raw non-parametric models are more accurate than the worst of the parametric models.

Figure B.6-11 shows that the recalibrated parametric median predictions are in closer agreement than the raw parametric median predictions (compare with figure B.6-4). Large adjustments for optimism have been made by recalibration of *JM* and *GO* and to a lesser extent for *MO*, *LM* and *LNHPP*, while for *DU*, *LV* and *KL* only very marginal adjustments for optimism have been made. Note that there is no longer growth in these predictions, which is not surprising, since, as noted earlier in Chapter 7, although there is growth over the whole data set for the region over which this analysis is conducted there is no growth. From table B.6-2.1 it can be seen that the *u*-plots for the recalibrated non-parametric predictors are all insignificant at the 5% level with the exception of *JMS* and *GOS* and figure B.6-12 suggests that these two models are still slightly optimistic after recalibration. These plots suggest that the *LVS* and *KLS* give the most accurate median predictions.

Figure B.6-13 suggests that recalibration generally improves the predictions for most of the parametric models although for *LVS* and *KLS* the recalibrated predictions are slightly worse at the end of the data set. Comparing figure B.6-14 with B.6-6 we can see that recalibration has resulted in predictors which are closer in accuracy than were the raw. *DUS*, *LVS* and *KLS* are best although *MOS*, *LMS* and *LNHPPS* are only marginally worse than these; *JMS* and *GOS* are generally worse than the others due, presumably, to the optimism still present in these recalibrated predictions.

The significant *u*-plots for *JMS* and *GOS* indicate non-stationarity in the prediction errors of *JM* and *GO* and examination of figure B.6-15 confirms this. Here we see that the raw predictions in these two models are initially unbiased and then become more and more optimistic as the data evolves, which explains why the *u*-plots are still optimistic after recalibration since in the later predictions insufficient adjustment will be made for optimism. Similar non-stationarity is also present to a lesser extent for the *MO*, *LM* and *LNHPP* models and for the *LV* and *KL* models recalibration may be marginally inefficient at the end of the data set since the raw models have become unbiased while earlier there was pessimism in the predictions.

The median predictions for the recalibrated non-parametric predictors (see figure B.6-16) are closer in agreement and less optimistic than the raw (compare with figure

B.6-7). For the models for which the raw predictions were noisy this noise is carried through into the recalibrated predictions. These predictions are also in closer agreement than the recalibrated parametric predictions (compare with B.6-11) and also no longer exhibit growth. From table B.6-2.1 it can be seen that for the non-parametric models all the *u*-plots are insignificant at the 20% level after recalibration.

Figure B.6-17 shows that there is generally improvement to be gained via recalibration for all the non-parametric models. Comparing figure B.6-18 with B.6-9 we can see that the recalibrated predictions are marginally closer in accuracy than the raw predictions. *CM2S*, *CM3S*, *OTY20S* and *OTY50S* give the best predictions; *OTLS*, *OTL20S* and *OTL50S* are worse in the first half of the data and *CM1S* is steadily worse than the others throughout the data set. Comparing B.6-18 with B.6-14 we can also see that the recalibrated non-parametric predictors are closer in accuracy than the recalibrated parametric predictors. Figure B.6-19 shows that the recalibrated non-parametric models are marginally worse than the best of the recalibrated parametric predictors but for the better of the recalibrated non-parametric models this is mainly due to a single jump downwards coincident with the large inter-failure time, t_{84}. Figure B.6-20 suggests that the raw models become more optimistic as the data evolves, with *CM1* consistently more optimistic than the others, but it is clear than any non-stationarity present in the prediction errors for the raw non-parametric models is not enough to result in significantly biased predictions after recalibration.

After the application of recalibration with windows it can be seen from table B.6-2.1 that all the *y*- and *u*-plots for all window sizes and all models are insignificant at the 20% level. There is again a tendency for much closer but more noisy median predictions for the parametric models as the recalibration window size decreases (see, for example, figures B.6-21 and B.6-22) when compared to recalibration without windows (figure B.6-11). The *PLR* analysis suggests that the predictions are closer in accuracy, and more so the smaller the window size, than for recalibration without windows (see, for example, figures B.6-23 and B.6-24 and compare with figure B.6-14). However, there is little improvement over recalibration without windows (see, for example, figure B.6-25 and B.6-26) not even over *JMS* and *GOS* which had significant *u*-plots. This is probably because the bias in *JMS* and *GOS* was fairly slight. For some of the parametric models performance is generally worse, particularly for the smaller windows sizes, and in such cases it seems that we have just the added noise to the predictions but not bought anything with respect to bias. From the equivalent plots for the non-parametric models a similar pattern is seen and no improvements by the application of recalibration with windows is gained over recalibration without (see, for example, B.6-27 and B.6-28).

From table B.6-2.2 we can see that for the meta-predictors all the *y*- and *u*-plots are insignificant at the 20% level for all window sizes. The median predictions (see figure B.6-29) again tend to be noisier for the smaller window sizes. From figure B.6-30 we can see that the predictive accuracy resulting from the different window sizes is fairly close; *M1* and *M2* are marginally worse than the others; *M40* and *M50* are marginally better but this is mainly due to jumps in the plots coincident with extreme data.

From figures B.6-31[17], B.6-14, B.6-13, B.6-19 and B.6-17 we can see that the better of these meta-predictors are comparable with the best initial prediction system, and that they are better than many of the initial prediction systems.

8.5.1 Summary for *USPSCL*

For this data set all the raw models except *LV* and *KL*, gave predictions which were on average in error. All of the biased raw predictors resulted in optimistic predictions, but the extent of the optimism varied depending on the model. *JM* and *GO* were the most optimistic of the parametric models and *CM1* was more optimistic than the other non-parametric models but not as optimistic as *JM* and *GO*. There was disagreement between the raw parametric predictions throughout the data while the non-parametric predictions were in close agreement but more noisy than the parametric. They were generally more inaccurate than the best of the parametric predictors but not as inaccurate as *JM* and *GO*.

On average improvement (reduction in the bias) for all the raw predictors which were initially in error could be achieved by the application of the recalibration technique, while for the best of the raw predictors there was little to choose between the recalibrated and the raw. The best of the recalibrated non-parametric predictors were marginally more inaccurate than the best of the recalibrated parametric predictors.

For some of the predictors, *JMS* and *GOS*, bias was still present even after recalibration. On further investigation it was found that there was non-stationarity in the prediction errors for these raw models with most of the bias in the predictions toward the end of the data. However, due to the fact that the bias remaining was fairly slight, application of recalibration with windows for these predictors gave little improvement. In many cases applying windows seemed to make things marginally worse than recalibration

---

17 The horizontal lines in this plot are where the meta-predictors are actually switching to the *DUS* model.

without windows, since unwanted noise was added to the predictions, which tended to increase as the window size used for recalibration decreased.

For this data set, performance of the meta-predictor varied slightly with window size, with predictions from *M1* and *M2*, which gave more noisy median predictions, slightly worse than those from the larger window sizes. These meta-predictors gave predictions which were better than many of the initial predictors and comparable with the "best" (on average) of the initial predictors.

8.6 Data set TSW

For this data set the *JM, GO, MO, LM* and *LNHPP* models all tend to the limiting case of an *HPP* for $i \sim 20\text{-}66$ coinciding with the region, as previously discussed in Chapter 7, where there is no growth in the data set. Then up to $i \sim 90$ the *LM* and *LNHPP* models tend to the limiting case of the *JM* and *GO* models respectively. The raw parametric median predictions (see figure B.12-4) for the *JM*, *GO*, *LM* and *LNHPP* models are more optimistic than the median predictions from the other raw parametric models and the *LV* and *KL* medians are the most pessimistic. The more optimistic models are giving noisier median predictions than the other models. This is partly due to their response to the extreme data values mentioned in Chapter 7, t_{79}, t_{86} and t_{105}; in response to these large times the medians jump upwards suddenly, taking some time to return to lower values as subsequent, less extreme data, is taken into account. After the first of these extreme values, t_{79}, the median predictions from the various models all generally exhibit growth but they disagree.

From table B.12-2.1 we can see that the *u*-plots for the raw parametric models are all significant, with most of them significant at the 1% level. Figure B.12-5 shows that the *u*-plots have a tendency to be *S*-shaped; these plots suggest that the *JM*, *GO*, *LM* and *LNHPP* predictions are optimistic, except for large times, the *LV* and *KL* predictions are pessimistic, except for small times while the *MO* and *DU* models are optimistic for small times and pessimistic for large times. These *u*-plots suggest that the most accurate median predictions are likely to be from *MO* or *DU*. The *log(PLR)* analysis in figure B.12-6 is swamped by the first extreme data value, t_{79}, where the *DU* model copes very badly (indicating a small right hand tail in the density for this model at this stage), while the *LV* and *KL* models cope the best. Even not taking account of this jump the *LV* and *KL* models are on average giving more accurate predictions than the other raw predictors, although the others, except for maybe *DU*, are as accurate towards the end of the data set.

The *OTL* model, and the raw non-parametric models applied with window size *50* gave *HPP* solutions, as for the parametric models, in the first half of the data set. For

window size *20* solutions fluctuated between *HPP* and non-*HPP* throughout most of the data set. Figure B.12-7 shows that, as for some of the raw parametric models, the median predictions tend to jump upwards in response to the large inter-failure times, t_{79}, t_{86} and t_{105}, although comparison with figure B.12-4 shows that the medians for the raw non-parametric models tend to return more quickly to lower values, as subsequent less extreme data is taken into account. This quickness in response for the non-parametric models results in the median predictions being generally more noisy than those from the parametric models. The medians are in closer agreement than those from the parametric models and growth is generally present in the non-parametric median predictions.

From table B.12-2.1 we can see that the non-parametric models all result in *u*-plots which are significant at the 1% level. According to the *u*-plots in figure B.12-8 these models are all optimistic except for larger inter-failure times, with the predictions from *CM1* marginally more optimistic than the others. Comparison with figure B.12-5 suggests that the errors in the predictions for these models are not as optimistic as those from *JM* and *GO*, and about the same as *LM* and *LNHPP*. Figure B.12-9 shows that, as for the raw parametric models (see figure B.12-6), comparative performance according to the *log(PLR)* analysis, is swamped by the inter-failure time, t_{79}, although the differences between the performance of the raw non-parametric models at this point is not as great as the difference between the parametric. Other differences in accuracy between the raw non-parametric models is fairly marginal but changeable over different regions of the data and the raw non-parametric models are closer in predictive accuracy than the raw parametric. Figure B.12-10 indicates that the non-parametric models are generally worse than the better of the parametric models, except toward the end of the data, but comparison with figure B.12-6 suggests that they are better than some of the parametric.

For this data set, according to the raw *u*-plots, all the raw model predictions are significantly in error. Comparing the median predictions in figures B.12-11 and B.12-4, we can see how recalibration has bought the median predictions for the parametric models much closer together, although they still disagree in some regions of the data. The *JM*, *GO*, *MO*, *LM* and *LNHPP* median predictions have been adjusted for optimism, with *MO* only adjusted very slightly and the *LV* and *KL* models have been adjusted for pessimism. Growth is still present in the recalibrated parametric median predictions and noise in the raw median predictions is also exhibited by the equivalent recalibrated medians. Table B.12-2.1, shows that, after recalibration, the *u*-plots for the parametric models have improved; they are now mostly insignificant at the 20% level.

Figure B.12-12 suggests that there is generally improvement to be gained by recalibration for all the parametric models, particularly for *DU* and for those regions where there is not improvement the recalibrated predictions are certainly no worse than

the raw. Comparison of figure B.12-13 with B.12-6 shows again that the *PLR* analysis is highly influenced by the behaviour of the predictions at the extreme value, t_{79}. The differences in performance at t_{79}, between the various recalibrated parametric predictors, are about the same as the differences initially present in the raw predictors. Apart from the differences at this single point, the recalibrated predictors are marginally closer in accuracy than the raw. Figure B.12-14 suggests that there is non-stationarity in the prediction errors, but that this is clearly not very significant (or at least, not enough to cause the recalibrated *u*-plots to be significant).

For the non-parametric models, the medians have been brought into slightly closer agreement via recalibration (compare figure B.12-15 with figure B.12-7). There has been adjustment for optimism in the raw non-parametric medians. The jumps upwards for the raw median predictions are carried through into the recalibrated medians and so these, too, are more noisy than the recalibrated parametric medians (see figure B.12-11). From table B.12-2.1 we can see that, for the non-parametric models, all the *u*-plots are insignificant at the 20% level after recalibration.

Figure B.12-16 shows that, generally, there is improvement to be gained by recalibration of the non-parametric models, particular in a region in the middle of the data set. Comparison of figure B.12-17 with B.12-9 shows that, after recalibration, the predictions are *not* any closer in accuracy than before, although they are marginally closer in accuracy to *LVS* (see figure B.12-18) than they were to *LV* before recalibration (see figure B.12-10). In fact, apart from the jumps, the recalibrated non-parametric models are as good as *LVS*. From figure B.12-19 we can again see that there is evidence of non-stationarity in the errors in predictive accuracy of the raw non-parametric models, but, again, this non-stationarity is not so pronounced that it results in bad *u*-plots after recalibration.

From table B.12-2.1 we can see that the *u*-plots for the only two recalibrated predictors which remained significant at the 20% level can be made insignificant by the application of recalibration with windows. However, according to the *PLR* analyses the only improvement gained for this technique for any of the 16 predictors is with respect to the particularly large inter-failure times (see, for example, figures B.12-20, B.12-21, B.12-22 and B.12-23). Apart from this the use of windows adds noise to the predictions resulting in steady decreases in the *log(PLR)* of the recalibrated with windows versus the recalibrated, and this problem becomes worse as the window size is decreased. It is not surprising that the application of windowing does not buy us any improvement over recalibration without windowing for this data set, since the recalibrated predictions already result in very good *u*-plots, indicating that there is no room for improvement with respect to bias in the predictions.

From table B.12-2.2 we can see that for the meta-predictor applied with varying window sizes, all the *u*-plots are insignificant at the 5% level (with some insignificant at the 20% level). Figure B.12-24 shows that the resulting median predictions are fairly noisy, particularly for small window sizes, and that sometimes the predictions resulting from the different window sizes, disagree quite a lot. Figure B.12-25 is again swamped by the behaviour at the extreme data point, t_{79}. Here some of these predictors (e.g. *M50*) have switched to initial prediction systems which behave favourably at this stage while others (e.g. *M30*, *M40*) have not. Apart from this point, *M30*, *M40* and *M50*, are marginally more accurate than the others. Again, *M1*, predicts surprisingly well, considering each prediction for this meta-predictor is only based on the single previous initial predictions.

Comparing figure B.12-26 with figures B.12-12, B.12-13, B.12-16 and B.12-18, shows that these meta-predictors are better than the majority of the original prediction systems and that some are as good as the best of the original prediction systems, although for those meta-predictors which are worse this is only due to their behaviour at t_{79}.

8.6.1 Summary for *TSW*

For this data set, *all* of the raw models gave predictions which were on average *in error*. The nature and extent of the bias in the predictions again varied depending on the model, with all except the *MO*, *DU*, *LV* and *KL* models giving on average generally optimistic predictions, particularly *JM* and *GO*, and *CM1* was again giving more optimistic predictions than the other non-parametric models but not as optimistic as those from *JM* and *GO*. *LV* and *KL* were giving on average generally pessimistic predictions, while the predictions from *MO* and *DU* were generally optimistic for small times and pessimistic for large times. The non-parametric models, which were again in closer agreement and accuracy, although they exhibited more noise gave predictions which were better than some of the parametric models, although not as good as the best of the parametric models, subject to some local variation. Again, it is interesting to note that the simple non-parametric predictor *OTL*, *OTL20* and *OTL50*, performed as well as all of the other non-parametric predictors.

Steady improvement (reduction in the bias) for all the raw predictors could be achieved by the application of the recalibration technique with marginally more improvement shown for those raw predictors which were initially worst. In this case non-stationarity in the raw prediction errors was not significant and recalibration thus effectively eliminated the bias in all the raw predictors. The recalibrated non-parametric predictors were as good as some of the recalibrated parametric predictors, but not as good

as the best of the recalibrated parametric predictors, although this was largely due to jumps downward coincident with extreme data values, rather than steady decreases.

In this case, since recalibration had already eliminated bias in the raw predictions there was little to be gained by the application of recalibration with windows. In fact, this technique generally made things steadily worse, and more so for smaller window sizes, due to increased noise in the predictions. However, for the application of recalibration with windows improvement with respect to problems encountered by the raw predictors due to the occurrence of particularly large inter-failure time data could be seen, indicated by large jumps in the *log(PLR)* plots coincident with these extreme data points.

For this data set, performance of the meta-predictor varied only marginally with window size, except with respect to extreme data points. The best of the meta-predictors gave predictions which were as good as the "best" (on average) of the initial predictors, and for those which were worse this was again only at extreme data points.

8.7 Data set TUSAB

The *JM, GO, MO, LM* and *LNHPP* models went to the limiting case of an *HPP* only for $i \sim 21\text{-}45$. Then, up to $i \sim 100$ the *LM* and *LNHPP* models went to the limiting cases of *JM* and *GO* respectively. From figure B.16-4 we see that after the second point of change in the raw data (previously discussed in Chapter 7) the raw parametric median predictions start to disagree; the *JM* and *GO* models give the most optimistic median predictions, with the *LM* and *LNHPP* medians also more optimistic than the others, while the *LV* and *KL* models give the most pessimistic median predictions. Most reliability growth is apparent in these predictions after this point although there is, on average, also marginal growth in the earlier predictions. The *JM* and *GO* models (and to a lesser extent, *LM* and *LNHPP*) tend to jump suddenly upwards in response to particularly large inter-failure times (e.g. t_{115}) and the median predictions and the median predictions from these models are more noisy than those from the other raw parametric models.

From table B.16-2.1 we see that the *MO* model results in the only insignificant *u*-plot for the raw parametric models on this data set and this *u*-plot is good, being insignificant at the 20% level. The *LNHPP* model gives a *u*-plot which is significant at the 5% level, while the *u*-plots for the rest of these models are significant at the 20% level. According to figure B.16-5 the *JM* and *GO* models are giving very optimistic predictions and so are the *LM* and *LNHPP* but to a lesser extent. These *u*-plots also suggest that the *DU, LV* and *KL* models are giving pessimistic predictions except for small inter-failure time. These plots suggest that the best median predictions probably come from the *MO* model. The *log(PLR)* plot (see figure B.16-6) is highly influenced by

extreme data values and we can see from the jumps upwards in these plots that the *DU* model performs particularly badly at these values. There is some variation in which we might choose as giving the most accurate predictions as the data evolves. From the 115^{th} failure the *JM* and *GO* models perform particularly badly compared to the others, while the difference between the others is fairly marginal.

The *OTL* model again gives *HPP* solutions in the same region as the parametric models (in the region $i \sim 21\text{-}45$) whereas for the non-parametric models applied with window size *50* there are very few *HPP* predictions, only within the region $i \sim 70\text{-}80$. For the non-parametric models applied with window size *20* the solutions fluctuate between *HPP* and non-*HPP* solutions throughout most of the data set. The median predictions from the raw non-parametric models, particularly those from *CM1* and *OTL* (see figure B.16-7), tend to be fairly noisy. They show a tendency to jump suddenly upwards in response to large inter-failure time data, but return more quickly to lower values as subsequent, less extreme, data is taken into account, than those from the parametric models which jump up similarly (compare with figure B.16-4). Apart from these jumps the raw non-parametric median predictions are closer in agreement than those from the parametric models and, as for the parametric medians, they generally exhibit growth, particularly after the 115^{th} failure.

Table B.16-2.1 shows that, for *CM2* and *OTY50*, the *u*-plots are insignificant at the 10% level, while the *u*-plots for the rest of the raw non-parametric models are significant at the 5% level, with those for *CM1* and all the *OTL* models significant at the 1% level. Figure B.16-8 suggests that the *CM1* model in particular is giving optimistic predictions (except for very large inter-failure times) and that the *OTL* models are also giving optimistic predictions (except for large inter-failure times). The rest of the *u*-plots indicate marginal optimism for small inter-failure times only. Comparison with figure B.16-5 suggests that the raw non-parametric predictors are not as biased as the worst of the raw parametric predictors. The *PLR* analysis in figure B.16-9 indicates that raw non-parametric predictors are fairly close in predictive accuracy and much closer in accuracy than the raw parametric predictors (compare the scale of this plot with the scale in figure B.16-6). Figure B.16-10 suggests that these models are comparable in accuracy with the *KL* model, which is among the best of the raw parametric models (see figure B.16-6). The jumps in the *log(PLR)* plots again coincide with extreme data values.

Comparing figure B.16-11 with B.16-4 we can see that recalibration has brought the median predictions from the parametric models into closer agreement but that there is still quite a lot of disagreement after the 115^{th} failure. The recalibrated parametric medians still exhibit most growth after this point, and *JMS*, *GOS*, *LMS* and *LNHPPS* still give noisier median predictions than the other recalibrated parametric predictors. The raw *JM*,

GO, *LM* and *LNHPP* median predictions have been adjusted for optimism while the *MO*, *DU*, *LV* and *KL* medians have been adjusted for pessimism, with the adjustments for *MO*, *LM* and *LNHPP* being fairly small. After recalibration of the parametric models some of the *u*-plots improve (see table B.16-2.1) but for the *u*-plots for *JMS* and *GOS* are still significant at the 1% level, while those for *LMS* and *LNHPPS* are still significant at the 5% level. It is interesting to note that, in this case, the evidence from the *y*-plots for the raw predictions coincide with the results of recalibration, since those with worse *y*-plots before recalibration result in worse *u*-plots after recalibration. Comparing figure B.16-12 with B.16-5 we can see that recalibration has indeed improved the *u*-plots, but in particular, those for *JMS* and *GOS*, and to a lesser extent for *LMS* and *LNHPPS*, are still optimistic, indicating that there has been an insufficient adjustment for the initial optimism in these raw models.

Figure B.16-13 shows that all raw parametric predictors are improved by recalibration in the second half of the data set, particularly *DUS*, *LVS* and *KLS* after the 110^{th} failure and *JMS* and *GOS* after the 120^{th} failure. In some regions of the earlier data recalibration seems to have made things marginally worse. Comparing the *PLR* analyses in figures B.16-14 and B.16-6 we can see that recalibration has made the predictions closer in accuracy but there are still differences in accuracy in the second half of the data; here *JMS* and *GOS* are still steadily worse than all the other prediction systems and in the region $i = 115\text{-}130$ the *LMS* and *LNHPPS* predictions are also bad; *MOS* and *DUS* are probably about the most accurate, with *LVS* and *KLS* also fairly good. *DUS* still performs particularly badly on the occurrence of large inter-failure times in the first half of the data set.

The presence of highly significant *u*-plots for *JMS* and *GOS* indicates non-stationarity in the raw prediction errors and figure B.16-15 confirms this; most of the optimism for these two raw predictors is at the end of the data set and we can see how over these regions insufficient adjustment will be made via recalibration which utilises all previous data. Non-stationarity also appears to be present for the *LV* and *KL* models, but this is clearly not significant enough to result in bad *u*-plots after recalibration. For *LM* and *LNHPP* non-stationarity is not so apparent from this figure, but it should be noted that although the *u*-plots for these two predictors were significant after recalibration they were not nearly as bad as those for *JMS* and *GOS*. Comparison with figure B.16-13 shows that improvement via recalibration generally coincides with regions where there is bias in the initial raw predictions and where there is no bias in the raw predictions it sometimes makes things marginally worse.

The recalibrated non-parametric median predictions are not closer than the raw (compare figure B.16-16 with B.16-7) except, sometimes, at the peaks, but these were

already very close before recalibration. Recalibration has generally made a small adjustment for optimism in the raw median predictions, and the adjustment being slightly larger for *CM1* than for the other non-parametric models. The jumps and the noise in the raw predictions are carried through into the recalibrated predictions, and the recalibrated median predictions still exhibit most growth in the second half of the data set. Comparison with figure B.16-11 shows that the recalibrated non-parametric medians are closer in agreement than the recalibrated parametric medians, but that they are generally more noisy than the parametric medians. For the non-parametric models all the *u*-plots are insignificant at the 20% level after recalibration (see table B.16-2.1) and evidence from the *y*-plots for the raw non-parametric predictors suggest that we should expect good *u*-plots after recalibration.

Figure B.16-17 indicates steady improvement in the accuracy of the predictions via recalibration for the non-parametric models in the second half of the data, while in the first half of the data recalibration seems to have made things marginally worse for all the non-parametric predictors except *CM1*. Comparison of figures B.16-18 and B.16-9 indicates that the recalibrated predictions are not closer in accuracy than the raw. The *CMS* models and *OTY50S* are marginally more accurate than the other recalibrated non-parametric models according to the *PLR* analysis, but there is some variation to this over different regions of the data. Figure B.16-19 indicates that the recalibrated non-parametric predictors are marginally less accurate than the best of the recalibrated parametric predictors but comparison with figure B.16-14 shows that they are as accurate as many of the recalibrated parametric predictors.

Figure B.16-20 suggests that, except for *CM1* , which is marginally more optimistic than the other raw non-parametric predictors, there is little bias in these raw predictors, and so we would not really expect to get large improvement via recalibration. Any non-stationarity present in the raw non-parametric prediction errors is clearly not so pronounced that it causes bad *u*-plots after recalibration.

Table B.16-2.1 suggests that for the parametric models improvement in the *u*-plots can be achieved by the application of recalibration with windows. In fact, with the exception of the *JM* and *GO* models, all the *u*-plots are now insignificant at the 20% level and for *JM* and *GO* the *u*-plots also become very good as the window size is decreased. For the non-parametric models all the *u*-plots are insignificant at the 20% level for recalibration with and without windows. As the window size decreases the recalibrated medians for the parametric models become closer and closer (see for example, figures B.16-21 and B.16-22 and compare with figure B.16-11) and also show a tendency to become more noisy. From figures B.16-23 and B.16-24 we can see that application of this method has generally resulted in predictions which are closer in accuracy than for

recalibration without windows (compare with figure B.16-14) but that there is still quite a bit of variation in performance as we move through the data. *JMS* and *GOS* are still worse than the other prediction systems in the second half of the data set, but from figures B.16-25 and B.16-26, we can see that this method has resulted in better predictive accuracy for these two models in this region than recalibration without windows. For the rest of the parametric models it would seem that, apart from some jumps for *DUS*, which are coincident with extreme data values, we have generally just added noise at the expense of little or no decrease in the bias and this problem becomes worse as the window size is decreased.

For the non-parametric models, we do not see the median predictions getting closer via the application of recalibration with windows (compare figures B.16-27 and B.16-28 with figure B.16-16), but this is perhaps not surprising since these predictions were very close already. In spite of this the *PLR* analyses (see figures B.16-29 and B.16-30), suggest that, apart from some jumps coincident with extreme data values, marginally closer accuracy results from using windows compared with recalibration without windows (compare with figure B.16-18). Figures B.16-31 and B.16-32 suggest that, as for most of the parametric models, we have gained little by the application of this method by just adding more noise with little or no decrease in the bias and this problem becomes worse as the window size is decreased.

From table B.16-2.2 we can see that the *u*-plot for *M1* is insignificant at the 10% level, for *M2* is insignificant at the 5% level and for the remaining meta-predictors the *u*-plots are insignificant at the 20% level. The median predictions for these predictors with small window sizes tend to be noisy (see figure B.16-33) and generally all show growth, particularly after the 115^{th} failure, although there are some regions of local decay. According to the *PLR* analysis (see figure B.16-34) for window sizes *10* up to *50*, these meta-predictors are pretty close in accuracy, while for the smaller window sizes, *M1*, *M2* and *M5*, performance is fairly steadily worse.

Comparison of figure B.16-35 with figures B.16-14, B.16-13, B.16-19 and B.16-17, suggests that the better of these meta-predictors are comparable in performance with the best of the initial prediction systems and better than many of the initial prediction systems.

8.7.1 Summary for *TUSAB*

For this data set, all of the raw models, except *MO*, *CM2* and *OTY50*, gave predictions which were on average *in error*. The nature and extent of the bias in the predictions again varied depending on the model, with all except the *MO*, *CM2*, *OTY50*,

DU, *LV* and *KL* models giving on average generally optimistic predictions, particularly *JM* and *GO*, and *CM1* was again giving more optimistic predictions than the other non-parametric models but not as optimistic as those from *JM* and *GO*. The non-parametric models, which were again in closer agreement and accuracy, although they exhibited more noise gave predictions which were better than some of the parametric models, and as good as the best of the parametric models, subject to some local variation. Again, it is interesting to note that the simple non-parametric predictor *OTL*, *OTL20* and *OTL50*, performed as well as all of the other non-parametric predictors.

Generally, steady improvement (reduction in the bias) for all the raw predictors which were initially in error could be achieved by the application of the recalibration technique in the second half of the data set, although in some regions of the data, where there was not significant bias in the raw predictions, recalibration made the predictions marginally worse. The recalibrated non-parametric predictors were as good as some of the recalibrated parametric predictors, but marginally worse than the best of the recalibrated parametric predictors.

For some of the predictors, in particular *JMS* and *GOS* bias was still present even after recalibration. On further investigation it was found that there was non-stationarity in the prediction errors for these raw models with most of the optimism in the predictions toward the end of the data. Application of recalibration with windows was shown to give improvement in regions of the data where optimism was still present after applying recalibration without windows. In other regions of the data and for most of the other models, apart from some improvements with respect to extreme data values, applying windows seemed to make things marginally worse than recalibration without windows, and often unwanted noise was added to the predictions, which tended to increase as the window size used for recalibration decreased.

For this data set, performance of the meta-predictor varied with window size, the larger window sizes generally giving more accurate predictions than the smaller window sizes, which tended to result in noisier predictions. The best of the meta-predictors gave predictions which were as good as the "best" (on average) of the initial predictors, and better than many of the initial predictors.

8.8 General Comments on Data Analyses

Tables 8.8-1 to 8.8-7 give summaries of the bias and the relative predictive quality of the different prediction systems for each of the 7 data sets. In each case the prediction systems arising from recalibration with windows are only included if the raw predictions were, on average, significantly in error and recalibration without windows failed to

eliminate this bias in the raw predictions. This is because there is little to be gained by applying recalibration with windows in such circumstances.

It is important to note that the rankings given in these tables represent an average taken over the whole range of predictions investigated for the data set in question. However, it has been seen that the predictive quality (both with respect to bias alone and/or according to the *PLR* analyses) of these various prediction systems frequently changes significantly *within* this range. Thus, the fact that a particular prediction system receives a high ranking does not exclude the possibility that it may have relatively good predictive quality in some subset of the data in question. Further, the rankings given here for the *PLR* analyses are fairly informal and are an attempt at representing the general average trend in the *PLR* plots: jumps in the *log(PLR)* plots which coincide with single extreme data values, for example, are ignored in cases where these single jumps are so large that they swamp all the results in other regions of the data. In practice these problems can be overcome by changing the choice of the particular prediction system to use for future predictions as the data evolves; risks associated with the possibility that some prediction systems may perform particularly badly at single data values can be taken into consideration in making such a choice. Thus, these issues are mentioned merely because they imply that the summaries in these tables should be interpreted with caution; in most cases they represent an over-simplification of the situation.

The meta-predictor presented here is a crude method which formalises a process of dynamic selection of different prediction systems as the data evolves. We would thus expect it to be more robust against local fluctuations in performance than the initial predictors. This is confirmed by the consistently low rankings seen for the meta-predictors in all the tables. However, even for this predictor there can be fluctuation in performance dependant on the window size being applied and so again, choosing dynamically between meta-predictors with different window sizes as the data evolves is advisable.

A final point that should be made about these tables is that, usually, the difference between the various better prediction systems is fairly marginal when compared with the difference seen between initially inaccurate raw predictions, or recalibrated predictions when recalibration is not effective in removing bias, and improved predictions. Thus the lower numbers tend to represent predictors which are fairly close in accuracy while the high numbers represent predictors which are dramatically worse.

	RAW			*S*			*S20*		*S30*		*S40*			*S50*					
	B	R^u	R^p	B	R^u	R^p	R^u	R^p	R^u	R^p	B	R^u	R^p	B	R^u	R^p		R^u	R^p
JM	=	1	1	=	1	1											***M1***	1	1
GO	=	1	1	=	1	1											***M2***	1	3
MO	=	1	1	PP	3	1											***M5***	1	4
DU	PPP	4	7	PPP	3	5	1	4	1	4	=	1	1	=	1	1	***M10***	1	4
LM	=	1	1	=	1	1											***M20***	1	1
LNHPP	=	1	1	=	1	1											***M30***	1	1
LV	PPP	5	8	PPP	4	6	1	1	1	1	=	1	1	=	1	1	***M40***	1	4
KL	PPP	5	8	PPP	4	6	1	1	1	1	=	1	1	P	2	2	***M50***	1	4
CM1	OOO	4	2	=	1	1													
CM2	=	1	4	PP	3	4													
CM3	=	1	4	=	1	4													
OTY20	=	1	1	=	1	1													
OTY50	=	1	1	=	1	1													
OTL	=	1	3	=	1	4													
OTL20	=	1	1	=	1	1													
OTL50	=	1	3	=	1	4													

B - bias, according to the *u*-plot	biased	generally optimistic	O - 2-5%; OO - 1-2%; OOO - significant at 1% level
		generally pessimistic	P - 2-5%; PP - 1-2%; PPP - significant at 1% level
		S-shaped *u*-plot	S - 2-5%; SS - 1-2%; SSS - significant at 1% level
	unbiased		= - insignificant at 5% level
R^u - ranking, according to the *u*-plots (i.e. bias)			1 is best
R^p - ranking, according to the *PLR* analyses (i.e. global accuracy)			1 is best

Notes: 1. All the *u*-plots for *S20*, *S30* and the meta-predictors, are unbiased for all 7 data sets.
2. Prediction systems arising from recalibration with windows are only included if the equivalent raw and recalibrated (without windows) predictions are biased.

Table 8.8-1. Summary of the bias and relative predictive quality of the different prediction systems for data set *CISI1*.

	RAW			*S*			*S20*		*S30*		*S40*			*S50*					
	B	R^u	R^p	B	R^u	R^p	R^u	R^p	R^u	R^p	B	R^u	R^p	B	R^u	R^p		R^u	R^p
JM	OOO	7	7	OOO	6	6	1	3	2	3	=	2	4	=	2	4	***M1***	1	1
GO	OOO	7	7	OOO	6	6	1	3	2	3	=	2	4	=	2	4	***M2***	2	1
MO	OOO	4	6	OOO	4	4	1	3	1	3	=	2	4	=	2	4	***M5***	1	2
DU	SSS	4	5	S	3	4	1	2	1	3	=	1	3	=	1	4	***M10***	1	2
LM	OOO	6	6	OOO	5	4	1	3	2	3	=	2	4	=	1	4	***M20***	2	2
LNHPP	OOO	6	6	OOO	5	4	1	3	2	3	=	2	4	=	1	4	***M30***	2	3
LV	PPP	4	3	=	2	3											***M40***	1	2
KL	PPP	4	3	=	2	3											***M50***	2	3
CM1	OOO	7	4	OOO	4	2	1	2	1	1	=	1	2	=	1	2			
CM2	OOO	5	4	OOO	4	3	1	3	1	3	=	1	4	=	1	4			
CM3	OOO	5	4	OOO	4	3	1	3	1	3	=	1	4	=	1	4			
OTY20	OOO	4	2	=	2	1													
OTY50	OOO	5	3	O	3	2	1	3	1	3	=	1	3	=	1	3			
OTL	OOO	5	4	OOO	4	3	1	3	1	3	=	1	3	=	1	4			
OTL20	OOO	4	2	O	3	1	1	2	1	2	=	1	1	=	1	2			
OTL50	OOO	5	3	O	3	2	1	3	1	2	=	1	3	=	1	3			

B - bias, according to the *u*-plot	biased	generally optimistic	O - 2-5%; OO - 1-2%; OOO - significant at 1% level
		generally pessimistic	P - 2-5%; PP - 1-2%; PPP - significant at 1% level
		S-shaped *u*-plot	S - 2-5%; SS - 1-2%; SSS - significant at 1% level
	unbiased		= - insignificant at 5% level
R^u - ranking, according to the *u*-plots (i.e. bias)			1 is best
R^p - ranking, according to the *PLR* analyses (i.e. global accuracy)			1 is best

Notes: 1. All the *u*-plots for *S20*, *S30* and the meta-predictors, are unbiased for all 7 data sets.
2. Prediction systems arising from recalibration with windows are only included if the equivalent raw and recalibrated (without windows) predictions are biased.

Table 8.8-2. Summary of the bias and relative predictive quality of the different prediction systems for data set *CISI2*.

	RAW			*S*			*S20*		*S30*		*S40*			*S50*					
	B	R^u	R^p	B	R^u	R^p	R^u	R^p	R^u	R^p	B	R^u	R^p	B	R^u	R^p		R^u	R^p
JM	OOO	6	6	OOO	4	4	1	2	1	2	=	1	2	=	1	2	***M1***	2	3
GO	OOO	6	6	OOO	4	4	1	2	1	2	=	1	2	=	1	2	***M2***	2	3
MO	SSS	4	5	S	3	2	1	2	1	1	=	1	1	=	1	1	***M5***	3	3
DU	SSS	6	7	S	2	1	1	2	1	1	=	1	1	=	1	1	***M10***	2	2
LM	OOO	5	5	OO	3	2	1	2	1	2	=	1	1	=	1	2	***M20***	1	2
LNHPP	OOO	5	5	O	3	2	1	2	1	2	=	1	1	=	1	2	***M30***	1	1
LV	PPP	6	5	P	3	1	1	3	1	3	=	1	2	=	1	2	***M40***	1	1
KL	PPP	6	4	=	1	2											***M50***	1	1
CM1	OOO	7	5	=	2	2													
CM2	OOO	5	4	=	1	2													
CM3	OOO	5	4	=	1	2													
OTY20	OOO	5	4	=	2	2													
OTY50	OOO	5	4	=	1	2													
OTL	OOO	5	4	=	1	1													
OTL20	OOO	6	5	O	3	2	1	3	1		=	1	3	=	1	3			
OTL50	OOO	6	5	=	1	2													

B - bias, according to the *u*-plot	biased	generally optimistic	O - 2-5%; OO - 1-2%; OOO - significant at 1% level
		generally pessimistic	P - 2-5%; PP - 1-2%; PPP - significant at 1% level
		S-shaped *u*-plot	S - 2-5%; SS - 1-2%; SSS - significant at 1% level
	unbiased		= - insignificant at 5% level
R^u - ranking, according to the *u*-plots (i.e. bias)			1 is best
R^p - ranking, according to the *PLR* analyses (i.e. global accuracy)			1 is best

Notes: 1. All the *u*-plots for *S20*, *S30* and the meta-predictors, are unbiased for all 7 data sets.
2. Prediction systems arising from recalibration with windows are only included if the equivalent raw and recalibrated (without windows) predictions are biased.

Table 8.8-3. Summary of the bias and relative predictive quality of the different prediction systems for data set *USBAR*.

	RAW			*S*			*S20*		*S30*		*S40*			*S50*				
	B	R^u	R^p	B	R^u	R^p	R^u	R^p	R^u	R^p	B	R^u	R^p	B	R^u	R^p		R^u R^p
JM	OOO	6	4	OOO	5	3	1	2	1	2	=	2	2	=	2	3	***M1***	2 3
GO	OOO	6	4	OOO	5	3	1	3	1	2	=	2	3	=	2	3	***M2***	1 2
MO	=	2	1	=	1	1											***M5***	1 2
DU	PPP	5	3	=	2	1											***M10***	2 1
LM	OOO	4	2	=	1	1											***M20***	1 1
LNHPP	O	4	2	=	1	1											***M30***	1 2
LV	PPP	6	3	PP	4	2	2	2	3	1	=	3	2	P	3	2	***M40***	1 2
KL	PPP	6	3	P	4	2	2	1	3	1	=	3	2	=	3	2	***M50***	1 2
CM1	OOO	5	2	=	1	1												
CM2	=	3	2	=	1	1												
CM3	=	3	2	=	1	1												
OTY20	=	3	2	=	1	1												
OTY50	=	3	2	=	1	2												
OTL	O	3	2	=	1	2												
OTL20	O	3	2	=	1	1												
OTL50	O	3	2	=	1	2												

B - bias, according to the *u*-plot	biased	generally optimistic	O - 2-5%; OO - 1-2%; OOO - significant at 1% level
		generally pessimistic	P - 2-5%; PP - 1-2%; PPP - significant at 1% level
		S-shaped *u*-plot	S - 2-5%; SS - 1-2%; SSS - significant at 1% level
	unbiased		= - insignificant at 5% level
R^u - ranking, according to the *u*-plots (i.e. bias)			1 is best
R^p - ranking, according to the *PLR* analyses (i.e. global accuracy)			1 is best

Notes: 1. All the *u*-plots for *S20*, *S30* and the meta-predictors, are unbiased for all 7 data sets.
2. Prediction systems arising from recalibration with windows are only included if the equivalent raw and recalibrated (without windows) predictions are biased.

Table 8.8-4. Summary of the bias and relative predictive quality of the different prediction systems for data set *USROFF*.

	RAW			*S*			*S20*		*S30*		*S40*			*S50*					
	B	R^u	R^p	B	R^u	R^p	R^u	R^p	R^u	R^p	B	R^u	R^p	B	R^u	R^p		R^u	R^p
JM	OOO	6	4	O	3	2	1	2	1	2	=	2	1	=	2	2	***M1***	1	3
GO	OOO	6	4	O	3	2	1	2	1	2	=	2	1	=	2	2	***M2***	1	3
MO	OOO	5	3	=	2	1											***M5***	1	2
DU	OO	3	3	=	2	1											***M10***	1	2
LM	OOO	5	3	=	2	1											***M20***	1	2
LNHPP	OOO	5	3	=	2	1											***M30***	1	2
LV	=	2	1	=	1	1											***M40***	1	1
KL	=	2	1	=	1	1											***M50***	1	1
CM1	OOO	5	4	=	1	2													
CM2	OOO	4	3	=	1	1													
CM3	OOO	4	3	=	1	1													
OTY20	OOO	4	3	=	2	1													
OTY50	OOO	4	3	=	1	1													
OTL	OOO	4	3	=	2	1													
OTL20	OOO	4	3	=	2	1													
OTL50	OOO	4	3	=	1	2													

B - bias, according to the *u*-plot	biased	generally optimistic	O - 2-5%; OO - 1-2%; OOO - significant at 1% level
		generally pessimistic	P - 2-5%; PP - 1-2%; PPP - significant at 1% level
		S-shaped *u*-plot	S - 2-5%; SS - 1-2%; SSS - significant at 1% level
	unbiased		= - insignificant at 5% level
R^u - ranking, according to the *u*-plots (i.e. bias)			1 is best
R^p - ranking, according to the *PLR* analyses (i.e. global accuracy)			1 is best

Notes: 1. All the *u*-plots for *S20*, *S30* and the meta-predictors, are unbiased for all 7 data sets.
2. Prediction systems arising from recalibration with windows are only included if the equivalent raw and recalibrated (without windows) predictions are biased.

Table 8.8-5. Summary of the bias and relative predictive quality of the different prediction systems for data set *USPSCL*.

	RAW			*S*			*S20*		*S30*		*S40*			*S50*					
	B	R^u	R^p	B	R^u	R^p	R^u	R^p	R^u	R^p	B	R^u	R^p	B	R^u	R^p		R^u	R^p
JM	OOO	5	4	=	2	2											***M1***	2	2
GO	OOO	5	4	=	2	2											***M2***	2	3
MO	SS	3	3	=	1	2											***M5***	2	4
DU	SSS	4	4	=	1	1											***M10***	1	3
LM	OOO	5	4	=	2	2											***M20***	1	3
LNHPP	OOO	5	4	=	2	2											***M30***	1	2
LV	PPP	5	2	=	2	1											***M40***	1	2
KL	PPP	5	2	=	1	1											***M50***	3	2
CM1	OOO	5	4	=	2	1													
CM2	OOO	4	4	=	2	1													
CM3	OOO	5	4	=	2	2													
OTY20	OOO	4	4	=	1	1													
OTY50	OOO	4	4	=	1	1													
OTL	OOO	4	4	=	1	2													
OTL20	OOO	5	4	=	1	1													
OTL50	OOO	4	4	=	1	2													

B - bias, according to the *u*-plot	biased	generally optimistic	O - 2-5%; OO - 1-2%; OOO - significant at 1% level
		generally pessimistic	P - 2-5%; PP - 1-2%; PPP - significant at 1% level
		S-shaped *u*-plot	S - 2-5%; SS - 1-2%; SSS - significant at 1% level
	unbiased		= - insignificant at 5% level
R^u - ranking, according to the *u*-plots (i.e. bias)			1 is best
R^p - ranking, according to the *PLR* analyses (i.e. global accuracy)			1 is best

Notes: 1. All the *u*-plots for *S20*, *S30* and the meta-predictors, are unbiased for all 7 data sets.
2. Prediction systems arising from recalibration with windows are only included if the equivalent raw and recalibrated (without windows) predictions are biased.

Table 8.8-6. Summary of the bias and relative predictive quality of the different prediction systems for data set *TSW*.

	RAW			*S*			*S20*		*S30*		*S40*			*S50*					
	B	R^u	R^p	B	R^u	R^p	R^u	R^p	R^u	R^p	B	R^u	R^p	B	R^u	R^p		R^u	R^p
JM	OOO	5	6	OOO	4	4	1	3	2	4	O	3	3	OOO	3	4	***M1***	2	4
GO	OOO	5	5	OOO	4	4	1	3	2	3	=	2	3	O	3	3	***M2***	2	3
MO	=	2	1	=	1	1											***M5***	1	4
DU	PPP	4	4	=	2	1											***M10***	1	2
LM	OOO	3	2	O	3	2	1	3	1	3	=	1	2	=	2	2	***M20***	1	2
LNHPP	O	3	2	O	3	2	1	3	1	3	=	1	2	=	1	2	***M30***	2	2
LV	PPP	5	3	=	2	1											***M40***	2	2
KL	PPP	5	2	=	2	1											***M50***	2	1
CM1	OOO	4	3	=	1	1													
CM2	=	2	2	=	1	2													
CM3	OO	3	2	=	1	1													
OTY20	O	3	2	=	1	2													
OTY50	=	2	2	=	1	1													
OTL	OOO	3	3	=	1	2													
OTL20	OOO	3	3	=	1	2													
OTL50	OOO	3	2	=	1	2													

B - bias, according to the *u*-plot	biased	generally optimistic	O - 2-5%; OO - 1-2%; OOO - significant at 1% level
		generally pessimistic	P - 2-5%; PP - 1-2%; PPP - significant at 1% level
		S-shaped *u*-plot	S - 2-5%; SS - 1-2%; SSS - significant at 1% level
	unbiased		= - insignificant at 5% level
R^u - ranking, according to the *u*-plots (i.e. bias)			1 is best
R^p - ranking, according to the *PLR* analyses (i.e. global accuracy)			1 is best

Notes: 1. All the *u*-plots for *S20*, *S30* and the meta-predictors, are unbiased for all 7 data sets.
2. Prediction systems arising from recalibration with windows are only included if the equivalent raw and recalibrated (without windows) predictions are biased.

Table 8.8-7. Summary of the bias and relative predictive quality of the different prediction systems for data set *TUSAB*.

From the analysis of the seven data sets in this chapter quite a lot of variation in the predictions and the predictive quality is seen for the different raw predictors. For example, for data set *CISI1* the *JM* and *GO* models are amongst the best raw predictors and *LV* and *KL* the worst, while for other data sets (e.g. *USBAR*, *USPSCL* and *TSW*) this is reversed, with *LV* and *KL* amongst the best and *JM* and *GO* amongst the worst. Further, such variation is seen not only over different data sets, but often over different

regions within the same data set. In general it could not be said that there is any one consistently best raw model and so applying a *group* of raw models seems to be a necessary approach, particularly if selection is to be made from amongst the raw predictors only and no further techniques for improving these predictions are going to be used.

Some of the raw models always gave very similar predictions. For the parametric models, pairs *JM* and *GO*, *LM* and *LNHPP* and *LV* and *KL* usually gave very similar predictions. This implies there is little benefit to be gained by applying more than one of each of these pairs of parametric models on each data set. In application of many of the parametric models acquiring estimates of the model parameters at each stage is computationally intensive. More importantly a certain amount of expertise is involved in choosing the control parameters in applying these models and so it would not be practical for a user to apply many of these models to each data set.

Generally all the raw non-parametric median predictions were quite similar to each other and according to the *PLR* analyses, these models are generally very similar in predictive accuracy, although *CM1* is frequently marginally worse. Surprisingly, the very simple analytical model, *OTL*, is frequently comparable in accuracy with the others. The raw non-parametric predictors performed about as well as the best of the raw parametric predictors (and better than many of the parametric predictors) on 5 of the data sets but for *USPSCL* and *TSW* they are worse than the best of the parametric models and for *CISI2* they are better. However, in this case where they are better we cannot exclude the possibility that this is due to the application of these models with moving windows rather than to the underlying assumptions of the models. Since the effort involved in applying these models is much less and, more importantly, acquiring the predictions just involves running the programs automatically, it is probably worth including a number of raw non-parametric models. In particular, the *OTL* model should be used, since the effort involved in using this is negligible, when compared to all of the others. Further, it may be possible to apply simple techniques which improve on the accuracy of these raw predictors to the extent that we do not have to apply any of the parametric models.

Some of the non-parametric median predictions tend to jump suddenly upwards in response to large inter-failure time data, and then, in contrast to the parametric models which also exhibit this behaviour (*JM*, *GO* and to a lesser extent *LM* and *LNHPP*), return more quickly to lower values as more typical data is taken into account. This is true not only for the non-parametric models that are applied with moving windows (for which we might expect quicker response to the data) but also for those non-parametric models which are applied over all the data, particularly *CM1*. Apart from these stages at which

some of the predictions suddenly increase, as mentioned above the median predictions for the non-parametric models are very close.

JM, *GO*, *LM* and *LNHPP* result in noisier median predictions than the other raw parametric models, particularly *JM* and *GO*. The raw non-parametric models tended to give noisier median predictions than the raw parametric models, but this does not, according to the *PLR* analyses, mean that they are much worse in predictive accuracy than the parametric models although there is some suggestion that this may degrade their performance slightly. Surprisingly this noise in the medians is not limited to those non-parametric models which are applied with moving windows, although, for each of these types of non-parametric models, smaller windows tend to result in noisier predictions than larger windows (as we would expect).

Various problems with some of the raw models were noticed, where particularly inaccurate predictions resulted at single prediction instances due to particularly extreme behaviour in the raw data. For example, the raw *DU* model in particular tends to encounter problems, when compared to the other raw models, in the *PLR* analyses on the occurrence of particularly large inter-failure time data. Although this occurred very rarely in the data sets we analyse here, the *JM* and *LM* model can predict that there are no bugs remaining in the software which is a highly optimistic (and hence very undesirable) prediction. There was also one such occurrence for the non-parametric model, *OTY50*, and it is possible for all of these non-parametric models to result in such predictions. When encountering such zero rate predictions a user should perhaps reject this prediction on principle if it occurs and choose one of the other models, or prediction systems, for prediction at this stage.

For the data sets examined here, apart from regions of data where there is no reliability growth (and aside from those groups of models previously mentioned which give very similar predictions), the raw median predictions from the different models are in great disagreement, with a tendency for more disagreement as the data evolves, and, in such cases the *u*-plots and *PLR* analyses confirmed that accuracy of a much wider class of reliability predictions than just medians also varied greatly for the different raw predictors.

The *u*-plots indicated that in most cases the raw predictors were, on average, grossly in error; out of all 112 (data set, raw predictor) pairs 80% had *u*-plots which were significant at the 5% level and 73% had *u*-plots which were significant at the 1% level. The nature and extent of the error depended on the data set in question and on the model. However, when bias was present the *JM*, *GO*, *LM*, *LNHPP* and all the raw non-parametric predictors resulted in generally optimistic predictions and *LV* and *KL* gave

generally pessimistic predictions. *DU* and *MO* showed more variation in the bias over different data sets; *MO* had a tendency towards optimism but sometimes an *S*-shaped *u*-plot resulted indicating optimism for small inter-failure times and pessimism for large inter-failure times; similar *S*-shaped *u*-plots were also seen for *DU* on some data sets but on other data sets average optimism, or average pessimism, was indicated by the *u*-plots for *DU*.

For all except data set *CISI1* gross optimism was present for *JM* and *GO* and the extent of the bias in these models was the worse seen, although for 3 of these data sets the extent of the bias in *LV* and *KL* was as bad, while for 2 data sets the extent of the bias in *CM1* was as bad. The *CM1* model resulted in *u*-plots which were significant at the 1% level on all seven data sets, and this model resulted in raw predictions which were more optimistic than the other non-parametric models, but, according to the *PLR* analyses, this does not seem to result in much worse predictive accuracy than the other non-parametric models although, as mentioned above, it is frequently marginally worse. Also, for three data sets the predictions from *CM1* were not as optimistic as those from *JM* and *GO*.

For data set *CISI1*, all except four of the *u*-plots for the raw predictors were insignificant, and for *USROFF*, *USPSCL* and *TUSAB*, some of these plots were insignificant. Thus, for these four data sets it is possible to get raw predictors which are, on average, unbiased, although the particular models which resulted in these unbiased predictors varied depending on the data set. For example, for *USPSCL* only *LV* and *KL* resulted in insignificant *u*-plots while for *USROFF* and *TUSAB*, only *MO* and some of the non-parametric models resulted in insignificant *u*-plots. For the remaining three data sets all the *u*-plots were highly significant implying that even the best of these raw models (according to the *PLR* analyses) could not be trusted to give accurate predictions on these data sets. The use of recalibration is essential, therefore, in order to eliminate the bias in the raw predictors.

In those cases where the raw models are initially biased, recalibration tends to greatly improve the predictions both with respect to better *u*-plots resulting and as seen by the *PLR* analyses. After recalibration only 32% of the *u*-plots out of 112 (data set, recalibrated predictor) pairs remain significant at the 5% level and this figure for the 1% level is 16% (as compared with 80% and 73%, respectively, before recalibration). Even in those cases where the *u*-plots remain significant they are much closer to the line of unit slope than they were before recalibration, and improvement, which is often dramatic, is seen in the *log(PLR)* plots.

In general the median predictions are much closer after recalibration than before, and the resulting set of recalibrated predictions is usually very close in accuracy, although

in some cases differences are still present in the recalibrated predictors. Sometimes, the raw predictor which is initially worse results in the best recalibrated predictions, while sometimes the raw predictor which is initially best results in the best recalibrated predictions. In some cases the best of the recalibrated non-parametric predictors are marginally worse than the best of the recalibrated parametric predictors even though both have good *u*-plots, presumably due to more noise in the recalibrated non-parametric predictions. In other cases the best of the recalibrated non-parametric predictors is comparable in accuracy with the best of the recalibrated parametric predictors.

In some cases recalibration helps a model to cope with extreme data values; this is only likely to be the case where there are a number of extreme data values within a single data set. Of course, noise in initial predictions will still be present after recalibration which seems a likely explanation for the marginally worse performance for the recalibrated non-parametric predictors compared with the recalibrated parametric predictors, and zero rate predictions which occur in the raw predictors are carried through to the recalibrated versions, and so neither of these problems can be eliminated by recalibration. Frequently, when there is no significant bias in the raw predictions, recalibration results in marginally worse accuracy according to the *PLR* analysis.

For some data sets, although recalibration improved the predictions dramatically (shown by improvements in the *u*-plots and in the *PLR*), the presence of significant *u*-plots after recalibration indicated that there was still room for further improvement, with disagreement between the recalibrated predictors and bias still present. The *JM* and *GO* models had *u*-plots which were significant at the 1% level after recalibration for 4 of the data sets but this behaviour was not limited to just these models. In fact for all the models except *OTY20*, *OTY50*, *OTL20* and *OTL50*, at least one data set resulted in *u*-plots which remained significant at the 1% level after recalibration. For data set *CISI2* many of the *u*-plots remained highly significant after recalibration.

On further investigation it was found that in such cases there was non-stationarity in the raw prediction errors. In general the *y*-plots of the raw predictors were not a good indication of this non-stationarity, and thus, of whether we should expect recalibration to effectively eliminate the bias in the raw predictions.

Detailed comparison of the nature of the non-stationarity in the prediction errors generally showed how in some regions, although improvement in the predictions might be seen via recalibration, insufficient adjustment to the raw predictions would be made and so we would not expect this technique to effectively eliminate the bias in the raw models. Further, it showed regions where there was no bias and from the *PLR* analyses it could be seen that in these regions recalibration sometimes made things marginally worse.

In fact, although improvement was generally shown for recalibration of models with significant *u*-plots, according to the *PLR* analysis of the recalibrated versus the raw predictors, there was often a certain amount of local fluctuation. For example, for many of the data sets bias was not present in the early data, and most improvement was only seen in the later data.

In general the results from applying recalibration suggest that it would probably not be sensible to initially apply just one raw predictor and then to recalibrate, since we cannot exclude the possibility that we may encounter problems of either noise, or significant bias, after recalibration on any single recalibrated predictor. Thus, a number of initial parametric and non-parametric models and the recalibration technique should be applied. On the other hand since the recalibrated non-parametric predictors were not dramatically worse than the best of the recalibrated predictors, minimising effort by trying just the non-parametric predictors and *DU* (which has an analytical solution, and so needs no expertise to apply), and then recalibrating, will probably give fairly reasonable results. Also, there seems to be evidence that the *JM* and *GO* models are particularly prone to non-stationary prediction errors and so these two models should probably not be included.

In regions where the non-stationarity in the predictions errors in the raw model predictions was significant enough to result in significant bias in the recalibrated predictions, slight improvement could be seen in the *PLR* analyses by the application of recalibration with windows as opposed to recalibration without windows. This technique always resulted in closer predictions and better *u*-plots, particularly for the smaller window sizes, but the predictions became more noisy as the window size decreased. For the larger window sizes unless the bias in the recalibrated (without windows) predictor was significant little improvement was seen since any decrease in bias bought by this technique was out-weighed by an increase in noise, and this was often also the case when there was significant bias and the window size was small. Elsewhere, where recalibration is already efficient, this technique tends to result in marginally worse predictions than recalibration without windows according to the *PLR* analysis and more so for small window sizes, since unwanted noise is added to already unbiased predictions.

The implication of this is that it would only be worthwhile applying recalibration with windows in two situations. First if the initial set of raw predictors resulted in recalibrated predictors where significant bias was present after recalibration in some regions of the data for all the recalibrated (without windows) predictors. The second situation would be where the only recalibrated predictors for which this is not the case resulted in noisy predictions and so are not a desirable choice on this basis. The first situation is quite likely to arise if only a small number of initial raw models is applied.

The second situation is not unlikely since models which are more able to capture the trend in the raw data when frequent fluctuations are seen in this trend, and hence result in stationary raw prediction errors and, on average, unbiased recalibrated predictions, may also be likely to be subject to more noise.

For the meta-predictor the results were very promising, with the performance according to the *u*-plot and the *PLR* criteria as good (differences were generally marginal) as the best of the initial prediction systems which could have been chosen for prediction purposes, and better than many of the initial predictors. There is varying performance for these meta-predictors applied with different window sizes (although this is often marginal) with smaller windows resulting in noisier median predictions but in general resulting accuracy is fairly close, certainly much closer than the group of initial predictors. For some data sets (e.g. *CISI2*) the meta-predictors applied with a small window perform surprisingly well. In particular this is surprising for *M1* since it is based on a comparison of only the most recent predictions with a single data point. However for other data sets, the application of such small window sizes does result in worse predictions as we would expect. It is disappointing that the meta-predictors do not result in better predictive accuracy than the best of the initial predictors. On the other hand, since the judgement as to which initial predictor is best is made in retrospect, perhaps we have made an unfair comparison here.

The main benefit of this crude meta-predictor is that it is completely automatic, and that it allows the user to apply a number of raw models and the recalibration technique (with and without windows) and to then select from amongst them dynamically as the data evolves without the detailed examination of the *PLR* plots and *u*-plots as conducted in this chapter, with reasonable confidence that the resulting meta-predictor will be as good as the best of initial predictors.

9 Conclusions and Future Work

The general conclusion of this work is that a "multi-modelling" approach is a good one. The analyses in the previous chapter suggest that for each new data set we could apply a small number of raw models (some parametric and some non-parametric), the recalibration technique (without windows and then with windows if necessary), and finally the meta-predictor, and the resulting predictions will be about the same in accuracy as any single predictor that could have been applied.

It has been repeatedly pointed out that for each data set the decision as to which is currently the "best" predictor (parametric or non-parametric, raw, recalibrated, or recalibrated with windows) may vary over different intervals of the data set and the meta-predictor approach is to automatically change our selection over *i* of the predictor which we would use for prediction purposes. This is only because we believe that the form of the reliability growth is changing over *i*. Thus we do not rely on theoretical justification as to whether the model assumptions are realistic or not. Instead we use our analysis techniques to decide which model is currently predicting the best, dynamically, as we move through the data, and use this for future predictions. In general, as stated above, this technique results in predictions which are among the best of all the single prediction systems that could have been chosen. This is a valuable point as it saves laborious analysis of many plots.

There are quite a few choices on exactly how to use the methods presented here, which partly depend on the resources available to the user. In reality it is more than likely that the effort needed in collecting the failure data required for use of these techniques far out-weighs the computational effort required for this multi-modelling approach. However, a certain amount of expertise is also needed, in particular for application of some of the raw models, and it is important to take this into consideration when deciding how best to use these techniques. It is obviously of great benefit to be able to use these techniques in such a way that fairly accurate reliability predictions automatically result and no such expertise is needed. More detailed recommendations on how to use these techniques are given in section 9.1 followed by, in section 9.2, suggestions for future work which the analyses presented here indicates would be of benefit.

9.1 Recommendations for Use of Multi-Modelling Techniques

The best approach to take if it is important to minimise the effort needed to get reasonably accurate reliability predictions, is to apply a small number of raw predictors, and then apply recalibration and recalibration with a range of window sizes to each raw

predictor, and then to automatically select from amongst all the resulting predictors using the meta-predictor applied with a range of window sizes. The *PLR* analyses of the meta-predictors alone could then be used to select an optimum window size for predictive accuracy. To make this process entirely automatic just one window size could be applied in the case of the meta-predictor, although this may result in marginally less accuracy than if selection is made from meta-predictors arising from a range of window sizes. Alternatively, the meta-predictor could be applied again using a single window size, but with the initial predictors being those arising from application of the meta-predictor the first time with the varying window sizes.[18]

If the group of initial predictors applied is limited to selection from *DU* and the non-parametric models, then since the results for these raw prediction systems are automatically generated this saves on the expertise involved in deciding on the control parameters for fitting the other more sophisticated parametric models. However, there may be circumstances where the results of such an approach would be marginally worse than if a number of these more sophisticated models were also included. In particular the non-parametric models sometimes perform marginally worse due to them having a tendency toward more noisy predictions; such noise cannot be eliminated by application of recalibration or the meta-predictor.

If it is deemed necessary to use some of these more sophisticated models then some intelligent decision should obviously be made about which of these models to use. For example, if the parametric models available are those investigated here, only one of pairs *JM* and *GO*, *LM* and *LNHPP*, and *LV* and *KL* should be used since each of these pairs usually give very similar predictions. If recalibration with windows is not to be used then it may not be worth applying *JM* and *GO* since these seem to be particularly prone to non-stationary prediction errors.

It cannot be guaranteed for any particular data set under investigation that individual raw predictors will not give predictions which are unacceptable since they are too noisy or unacceptable in some other way that cannot be eliminated by recalibration. For example, zero-rate predictions may result as was seen here at some prediction stages. Thus, in general it is better that the group of initial raw predictors is not too small in order to increase the chances of always getting acceptable predictions from at least one of the raw prediction systems or from one of the recalibrated.

---

18 This has not been applied to the data analysed in the previous chapter, but it is an obvious approach since it is just a way of formalising and automating examination of the associated *log(PLR)* plot.

Whatever the group of initial raw predictors used, the possibility to apply recalibration to each member of this initial group is essential, since it is quite possible that all of these raw predictors for a particular data set may be in error. Use of the meta-predictor alone with a group of initially inaccurate raw predictors is not a sensible alternative, since the meta-predictor is unlikely to be any more accurate than the best of the initial raw predictors.

In some cases, particularly if the group of initial raw predictors is small, recalibration can result in predictors which are all still biased and in such cases application of recalibration with windows is necessary in order to eliminate this bias. However, a range of window sizes should be used in order to insure the optimum balance between elimination of irrelevant early raw predictions in assessment of the current predictive error versus increased noise as a result of smaller window sizes.

Recalibration both with and with windows is easy to apply and so these might as well be used as a matter of course.

Whatever the final group of predictors, the meta-predictor investigated here is a useful tool for formalising and automating the process of selection, from amongst this group, of a predictor to use for future predictions at any stage in the data.

9.2 Suggestions for Future Work

In general it seemed that some of the non-parametric models, and the recalibration technique and meta-predictor when applied with small windows, tended to be more "data-driven" and more noisy predictions resulted, particularly for small window sizes, resulting in marginally worse predictive accuracy. It might be beneficial to investigate the possibility of applying techniques to smooth this noise. If we could develop such techniques it may be possible to only apply a very small group of initial raw models for which no expertise is needed by the user (i.e., by just using the non-parametric models and *DU*) and for predictions to result which are as good as those which could have been obtained from more sophisticated models. Preliminary investigations of a simple smoother for the non-parametric models, which just consists of using the average of the most recent one-step ahead rate predictions, was tried in [Brocklehurst 1989]. Clearly, for data which exhibits reliability growth, this is likely to result in pessimistic predictions but it was felt that this could be eliminated later by recalibration. However, it seemed that this method also resulted in non-stationary raw prediction errors and so recalibration was not efficient. It may be worth investigating some more sophisticated noise smoothing techniques.

There was often a trade-off in application of the raw non-parametric models with windows where a smaller window size tends to result in raw predictions which capture the trend in the data and sometimes are less biased and hence good raw or recalibrated (without windows) predictions result, although this decrease in bias is often out-weighed by an increase in noise in the predictions. Such a trade-off was also present with application of recalibration and the meta-predictor with windows, where there was less bias in the recalibrated predictions and quicker response of the meta-predictors to significant local changes in the predictive performance of the initial predictors with smaller windows, but again, often at the expense of too much noise. It is clear that the application of fixed size windows is not the most efficient way to eliminate non-stationarity in the raw model prediction errors used for recalibration, or to eliminate earlier predictions which do not represent the current predictive accuracy in achieving the meta-predictor. Remember, what we are seeking is a window small enough to eliminate less relevant past data or predictions, but large enough so that noise is kept at a minimum.

More accuracy might be gained by trying to assess exactly where "change-points" in the data or the prediction errors are, and in this way apply a more intelligent window, the size of which is decided upon by the location of such change-points. This would result in a window size which dynamically changes as the data evolves. What would be required here is, in the case of application of the raw models with windows, techniques for assessing change-points in the raw data; here we do not only refer to change-points such as a transition from stable reliability to reliability growth in the raw data (which can easily be detected by the Laplace statistic) but changes from one type of reliability growth to another. Techniques for the latter are not yet available although it might be possible to develop such techniques by using the raw model assumptions as a null hypothesis and detecting where there is significant departure from this hypothesis. In this way we could use the largest window of recent data possible at each prediction stage subject to this window of data not showing significant departure from the model assumptions. This would result in using a window size which varies with each prediction stage and also at a single prediction stage depending on the model.

The *OTL* model applied on the data sets analysed in the previous chapter was in fact applied in a manner similar to this. The largest vector of most recent data which exhibited no growth was selected via the sign of the Laplace statistic of the data and then an *HPP* was fitted over this vector. It might be worth investigating whether better results are obtained for *OTL* by using the significance test for the Laplace statistic rather than just its sign, and thus finding the largest vector which does not exhibit significant growth (or decay).

In the case of application of recalibration we would be interested in assessing change-points in the prediction errors or, more specifically, in the sequence of *u*s generated by each raw model. Our informal examination of the moving *u*-plots of raw predictions often showed quite clearly such non-stationarity but really we need a formal test to decide when there is a significant change in the prediction errors. This would involve using the *u*-plot for recalibration constructed from the largest window of most recent past predictions for which any non-stationarity in the prediction errors for the raw model is insignificant. This window will thus change dynamically as we move through the data for single raw model, but will also be different at the same prediction stage for each raw model. One significance test which does exist for this is that based on the K distance of the *y*-plot constructed from the sequence of *u*s. Unfortunately, though, the work presented in the previous chapter, and other previous work, has shown that this test is not very effective and so it would probably better to seek a better significance test for non-stationarity in the prediction errors.

For the meta-predictor we would be interested in assessing change-points in the trend of the *PLR* plots which represent a genuine switching of performance of the initial prediction systems. Again, a formal significance test would be useful in deciding when there is a significant change in the relative predictive accuracy of these raw models. In this case we would be seeking the largest window for which there are no significant changes in the relative predictive accuracy of predictor pairs. This will result in using a different window for each paired comparison.

An alternative approach to all of these methods for detecting change points would be to generate past predictions using all window sizes, and then to use the *PLR* as a criterion for deciding which window size to use for the next prediction. Although this would be computationally intensive (particularly in the application of the raw models, with the exception of *OTL* and *DU*), it might be worth investigating since methods for detecting change-points are not yet available. Note that this approach is equivalent to forming a meta-predictor as described in Chapter 6, but using the predictors from just one of the techniques (the raw or the recalibrated or the meta-prediction themselves) with the varying window sizes, as the initial group of predictors over which selection is made.

Using such an approach for recalibration the window used for recalibration of the current prediction would be the one which results in the best last b recalibrated predictions according to the *PL*. In fact this was approximated to in the previous chapter, by using the recalibrated predictors with a number of fixed window sizes in the group of initial predictors used to construct our meta-predictor.

Precisely such an approach of combining meta-predictors is taken in [Lu and Brocklehurst 1991; Lu et al. 1993]. There a number of ways of combining a group of initial predictors, including the meta-predictor which is presented in Chapter 6 and referred to in [Lu and Brocklehurst 1991; Lu et al. 1993] as "switching", are investigated. The switching combined predictor is constructed using all possible window sizes, a, of the previous raw predictions for prediction stage j, $j = i\text{-}b, \ldots, i\text{-}1$ and then an optimum window size, a^*, is chosen as that value of a which results in the maximum PL calculated for predictions of T_j, $j = i\text{-}b, \ldots, i\text{-}1$. This optimum window size, a^*, is then used for the meta-prediction at stage i. Of course we now have the dilemma of choosing this second window size, b. But, in [Lu and Brocklehurst 1991; Lu et al. 1993] it is found that for different values of b the final meta-predictors are usually about the best of the initial meta-predictions over which selection is made with a number of fixed window sizes, a and there is not as much variability with different b as there is with different a. Thus these results are consistent with what we discovered in Chapter 8, i.e., that the meta-predictor tends to result in predictions which are about as good as the best of the initial predictions (i.e., in this case meta-predictors themselves) and are closer in accuracy than the group of initial predictors.

Further, for all these methods, the u-plot could be used instead of the PL by finding that window which minimises the K distance of the u-plot of the resulting past predictions and use this window for the next prediction. This might be particularly useful for deciding on a window for recalibration, since the primary objective of this technique is to eliminate bias.

An alternative to using windows of recent predictions would be to instead use weights, giving larger weights to more recent predictions. This approach is used for recalibration of reliability predictors for discrete failure data, that is, when the observed failure data is in the form of failure counts in fixed time intervals, in [Wright 1988; Wright 1993]. There the u-plot used for recalibration is constructed from a weighted combination of the previous us, so that the contributions from the us die away as we move further back into the prediction sequence. In [Wright 1988; Wright 1993] it was found that variability of predictive performance can result by discounting earlier predictions to different extents by varying the choice of weights for recalibration. Again, a trade-off could be seen, with vectors of weights which result in taking large account of only the most recent predictions often resulting in too much noise. Choice of an optimum vector of weights could be made using similar methods suggested for choosing optimum windows above (i.e., by using the PLR or the u-plot of previous recalibrated predictions with different vectors of weights), and the recalibrated predictions constructed via this optimum value could be used for the next prediction.

Another approach rather than using windows or weights would be to apply the techniques a second time. For example, in the case of those prediction systems which were still biased after recalibration (without windows) it might be worth investigating whether any improvement might be gained by recalibrating the recalibrated prediction systems, just as forming combined predictors from initial predictors which are themselves combined predictors was shown to be useful in [Lu and Brocklehurst 1991; Lu et al. 1993] when there is quite a lot of variation with window size in the predictive accuracy of the initial group, as discussed above.

In general each application of further techniques like recalibration and forming combined predictors requires more initial raw data and predictions. Thus, if methods *could* be found for change-point detection in these contexts, then this would be preferable to re-applying the techniques, or to searching for optimal windows or weights.

The meta-predictor we applied here was very crude, and it may be beneficial to investigate more sophisticated methods of combining a set of initial prediction systems. In particular it might be preferable to form combined predictors which continue to take account of all the initial predictors at each stage rather than just switching to a single predictor.

In [Lyu and Nikora 1992] a number of methods of combining initial predictors are investigated where the combined predictions are constructed using a linear combination of the initial predictors. In this investigation the methods used for constructing the weights are fairly simplistic. One method is just to give all the initial predictors equal weight at every prediction stage. Another, which in contrast to the previous one results in dynamically changing weights but only in cases where the initial predictors switch in their ranking of most optimistic to most pessimistic predictions, is to give those predictors which are most optimistic, or pessimistic less or zero weight than those which lie in the centre of the group of predictors. In [Lyu and Nikora 1992] the only combination method investigated which depends on past predictive accuracy is similar to *M1*, as applied in Chapter 8 (in fact it is the Bayesian version of this predictor, which is discussed below), and it is found that the other combined predictors do not perform any better than this predictor, suggesting that it is probably not worth investigating any of these more simplistic ways of constructing combinations further.

In [Lu and Brocklehurst 1991; Lu et al. 1993], in addition to "switching", another combined predictor which depends on past predictive accuracy of the initial predictions is investigated, for which the results are marginally better than those from switching. This predictor was suggested in [Edwards 1984] and discussed in [Saglietti 1989]. This combined predictor is constructed by using a linear combination of the initial predictors

with the weight, w_i^r, from each initial predictor, r, being the normalised contribution from the PL from that predictor over the last a predictions, i.e.,

$$w_i^r = \frac{PL_{i\text{-}a\ i\text{-}1}^r}{\sum_{n=1}^{N} PL_{i\text{-}a\ i\text{-}1}^n} \qquad r = 1, \ldots, N$$

for combination of N initial predictors. This is referred to as the Bayesian combined predictor since it can be derived by constructing the posterior odds ratio given the data and the prior representing initial indifference between the initial predictors to be combined. Although, as mentioned above, investigation of this Bayesian combined predictor in [Lu and Brocklehurst 1991; Lu et al. 1993] showed that the resulting predictions are better than the cruder switching combined predictor, the results generally only differ marginally, with the Bayesian combined predictors being, as with switching, about the same as the best of the initial predictors over which combination was made.

Another alternative way to form a combined predictor via a linear combination of initial predictors, is to optimise some accuracy criterion over all possible choice of weights used in the combination. Such an approach is suggested in [Littlewood 1988] and discussed in [Saglietti 1989], with the PL as the accuracy criterion on which to base this combination. In detail it is suggested that a search for the vector of weights which maximises the PL of the combined predictor taken over the last a predictions be done, and that this optimal vector of weights should be used to construct the predictions at the next stage. Of course this would be computationally intensive since for combination of N initial predictors it involves maximising a polynomial of order a, in an $N\text{-}1$ dimensional space $[0,1]\times \ldots \times[0,1]$, since the weights are constrained so that their sum is equal to one. However, some crude approximation to this could be easily implemented by only allowing the weights to take a fixed number of discrete values in [0,1] and by only combining over a small number of initial predictors.

In any combination method used it may be preferable to make a more intelligent choice of the initial group of predictors to be combined, either for the purposes of achieving more accurate combined predictions, or in order to minimise the computational effort required in achieving the combined predictor. In [Lu and Brocklehurst 1991; Lu et al. 1993] combination is made over the 8 raw parametric predictors alone, over the 8 corresponding raw and recalibrated predictors and over the 16 raw and recalibrated predictors together. Interestingly it turns out that slightly worse performance is achieved when the worst group of recalibrated or raw predictors (usually the raw) are included in the initial group, implying that it may not be the best strategy to use all available initial predictors in our combination as was done in Chapter 8, but instead to use the best subset of the initial predictors.

In [Saglietti 1989] it is suggested that single predictors might be rejected for inclusion in the initial group to be combined due to their bad (past) performance according to absolute measures of predictive accuracy, for example, *u*-plots and *y*-plots. In [Lyu and Nikora 1992] it is also suggested that predictive accuracy on past data, according to *u*- and *y*-plots, *PL* analyses, and a noise measure [Abdel-Ghaly et al. 1986], could be used to select the group of initial predictors. Some more general properties of raw models, such as computational intensity of achieving estimates of model parameters, applicability of model assumptions to application under investigation, ability of models to make predictions which are of interest in the application under investigation, and so on, are suggested in [Lyu and Nikora 1992] as additional criteria for selecting the initial group of predictors to be combined.

Another way to incorporate preferences towards particular initial predictors would be to use some scheme based on past performance within a particular data set, or from other data sets, other than indifference between the initial predictors in the prior for the Bayesian combined predictor discussed above.

In [Lu and Brocklehurst 1991; Lu et al. 1993] substantial improvement was only seen in the combined predictor over the best of the initial predictors, when all of the initial predictors being combined were giving grossly inaccurate predictions. However, the improvement seen in this case, where combination was over the raw predictors only, was not a great as that gained by recalibration. This suggests that a strategy which consists of applying raw models only, and then combining, is not as good as one which instead combines recalibrated predictions. To date, when the latter strategy is used, we have not seen any substantial improvement of the combined predictors over the initial predictors. Of course, this may be because the best of the initial recalibrated predictors are already fairly accurate, and there is not much room for more improvement. The implication is that there is not enough significant local variation in the relative accuracy of the recalibrated predictors for the combined predictor to be expected to give improvement. However, using a combined predictor is still recommended since it is simply a way of automatically choosing the best initial recalibrated predictor.

In most of our discussions we have only considered one-step-ahead predictions. Obtaining predictions further into the future is a problem for many of the raw models. Each raw model has its own particular problems for predicting further into the future. For many of the parametric models the extension to the most simple type of future prediction, that of n steps ahead, is obvious. However, obtaining other kinds of predictions, such as the time taken to achieve some target reliability, is non-trivial for many of the raw parametric models.

For the non-parametric models even obtaining *n*-step-ahead predictions is problematic. Some recommendations of how this might be approached for the *CM* model is given in [Miller and Sofer 1988]. It is suggested that *n*-step-ahead rates can be derived by extrapolating from the estimated set of rates at any one prediction stage. As yet this method has only been tested on simulated data and it may be worth investigating it for real data.

Unfortunately, as mentioned in [Brocklehurst 1989], the detail of the method suggested, which requires that complete monotonicity is preserved for the vector of future (and previously estimated) rates, would result in simply using the most recent rate for *n*-step-ahead predictions, whatever the value of *n*, in many cases. This is due to the detailed nature of the solutions for estimated rates which result for the various non-parametric models. For the extension to *OTL* all the rates are almost always equal and so to preserve complete monotonicity all future rates must be equal. For *OTY*, as observed in Chapter 5 and [Brocklehurst 1989], in many instances the estimated sets of rates have shapes which result in the most recent rates being equal, and again, to preserve complete monotonicity all future rates must be equal. It is necessary, therefore, to seek an alternative method for obtaining *n*-step-ahead predictions for the non-parametric models. Another option would be to constrain the optimum solution for *OTY* to have no identical rates, whenever possible. This approach would probably work whenever the failure data exhibits growth, as there is a tendency for highly non-unique solutions to result in the presence of growth.

Another, possibly insurmountable, problem is how to use the analysis techniques, and apply recalibration and the combined predictors to assess and to obtain such predictions. It is clear that, just as we observed in Chapter 4 that inaccuracies of model retrodictions are not representative of inaccuracies in one-step-ahead predictions, so the inaccuracies in one-step-ahead predictions cannot be expected to be representative of inaccuracies of predictions further into the future. This means that to assess the efficiency of a predictor for a particular type of prediction that same type of prediction must be used in the plots for analysis (and in the methods for recalibration and constructing combined predictors).

Consider, again, the most simple type of future prediction, that of *n* steps ahead. To decide from a set of predictors which to use for the next *n*-step-ahead prediction, we must make many such (past) *n*-step-ahead predictions and use these in the plots for analysis which compare these predictions with the (later observed) data. The larger the value of *n* the less and less likely it will become that we will have enough data to conduct such an analysis over a single data source. It may be that the only way to approach such a problem is to appeal to evidence based on another data source. Here, we mean, to use previous data sources to assess whether the ability of a particular predictor to predict the

immediate future implies an ability for this predictor to predict further into the future; if we find that this is the case we might then be able to assess predictions only in the immediate future for the data of interest, and select our prediction system for the predictions further into the future based on evidence of such correlation from other data sources.

Even if we found that such correlation existed, which is doubtful, it seems likely that the same approach for recalibration, for example, would not be successful. Here, we would need to be able to make some connection between errors in one-step-ahead predictions and errors in *n*-step-ahead predictions on previous data sources that can then be applied to achieve recalibrated *n*-step-ahead predictions for the data source under investigation, merely from the one-step-ahead predictions for that data source. This is unlikely to be the case. Similar comments apply in the case of combining predictions. It may be the case, therefore, that to achieve good predictions further into the future than one-step-ahead, our only option will be to seek a model, or method, which can be shown to give accurate future predictions consistently on many data sources.

References

[Abdel-Ghaly 1986] A.A. Abdel-Ghaly. *Analysis of Predictive Quality of Software Reliability Models,* Ph.D. dissertation, City University, London, 1986.

[Abdel-Ghaly et al. 1986] A.A. Abdel-Ghaly, P.Y. Chan and B. Littlewood. "Evaluation of Competing Software Reliability Predictions," *IEEE Trans. on Software Engineering*, vol. SE-12, no. 9, pp.950-967, 1986.

[Brocklehurst 1989] S. Brocklehurst. *A non-parametric approach to software reliability modelling,* PDCS1 (ESPRIT Project 3092) Technical Report number 4, Centre for Software Reliability, City University, London, 1989.

[Brocklehurst et al. 1990] S. Brocklehurst, P.Y. Chan, B. Littlewood and J. Snell. "Recalibrating Software Reliability Models," *IEEE Trans. on Software Engineering*, vol. SE-16, no. 4, pp.458-470, 1990.

[Brocklehurst et al. 1991] S. Brocklehurst, K. Kanoun, J.C. Laprie, B. Littlewood, S. Metge, P. Mellor and A. Tanner. "Reliability analyses of workstation failure data," in *Proc. ESPRIT '91,* pp. 806-821, Brussels, CEC, 1991.

[Brocklehurst and Littlewood 1992] S. Brocklehurst and B. Littlewood. "New Ways to Get Accurate Reliability Measures," *IEEE Software special issue on Reliability Measurement*, pp.34-42, July, 1992.

[Brocklehurst et al. 1993] S. Brocklehurst, B. Littlewood, E. Jonsson and T. Olovsson. *Data Collection for Security Fault Forecasting: Pilot Experiment,* PDCS2 Project (ESPRIT Project 6362) First Year Report, chapter 4, part 4.4, Centre for Software Reliability, City University, London, 1993.

[Brocklehurst et al. 1994] S. Brocklehurst, B. Littlewood, E. Jonsson and T. Olovsson. "On Measurement of Operational Security," in *Proc. Ninth Annual Conference on Computer Assurance, COMPASS '94,* pp. 257-266, Gaithersburg, MD, IEEE, 1994.

[Chan 1986] P.Y. Chan. *Software Reliability Prediction,* Ph.D. dissertation, City University, London, 1986.

[Chan and Littlewood 1986] P.Y. Chan and B. Littlewood. *Parametric Spline Approach to Adaptive Reliability Modelling,* Technical Report, Centre for Software Reliability, City University, London, 1986.

[Cheung 1980] R.C. Cheung. "A User-Oriented Software Reliability Model," *IEEE Trans. on Software Engineering*, vol. SE-6, pp.118-125, 1980.

[Cox and Lewis 1966] D.R. Cox and P.A.W. Lewis. *Statistical Analysis of Series of Events,* London, Methuen, 1966.

[Crow 1977] L.H. Crow. *Confidence Interval Procedures for Reliability Growth Analysis,* Technical Report 197, US Army Material Systems Analysis Activity, Aberdeen, MD, 1977.

[Dawid 1984] A.P. Dawid. "Statistical Theory: The Prequential Approach," *J Royal Statist Soc A*, vol. 147, pp.278-292, 1984.

[DeGroot 1986] M.H. DeGroot. *Probability and Statistics,* Series in Statistics. Reading, Mass., Addison-Wesley, 1986.

[Duane 1964] J.T. Duane. "Learning Curve Approach to Reliability Monitoring," *IEEE Trans. on Aerospace*, vol. 2, pp.563-566, 1964.

[Edwards 1984] G. Edwards. "A Bayesian Procedure for Drawing Inferences from Random Data," *Reliability Engineering*, vol. 9, pp.1-17, 1984.

[Fenton and Hill 1993] N. Fenton and G. Hill. *Systems Construction and Analysis: a Mathematical and Logical Framework,* London, McGraw-Hill International (UK) Ltd., 1993.

[Fenton et al. 1994] N. Fenton, S.L. Pfleeger and R.L. Glass. "Science and Substance: A Challenge to Software Engineers," *IEEE Software special issue on Measurement-Based Process Improvement*, July 1994.

[Fetzer 1988] J.H. Fetzer. "Program Verification: The Very Idea," *Communications of the ACM*, vol. 31, no. 9, pp.1048-1063, 1988.

[Gaudoin 1988] O. Gaudoin. *Les Tests de Tendance de Fiabilite des Systemes Reparables. Application a la Fiabilite des Logiciels,* IMAG TIM3, Grenoble Cedex, 1988. In French

[Gehani and McGettrick 1986] N. Gehani and A.D. McGettrick, (Ed.). *Software Specification Techniques,* International Computer Science, Wokingham, England, Addison-Wesley Publishing Company, 1986.

[Goel and Okumoto 1979] A.L. Goel and K. Okumoto. "Time-Dependent Error-Detection Rate Model for Software and Other Performance Measures," *IEEE Trans. on Reliability*, vol. R-28, no. 3, pp.206-211, 1979.

[Jelinski and Moranda 1972] Z. Jelinski and P.B. Moranda. "Software Reliability Research," in *Statistical Computer Performance Evaluation, ed. W. Freiberger*, pp. 465-484, New York, Academic Press, 1972.

[Kanoun 1989] K. Kanoun. *Software Dependability Growth Characterization, Modelling and Evaluation,* Doctorat es-Sciences, Institut National polytechnique de Toulouse, 1989.

[Kanoun et al. 1988] K. Kanoun, J.C. Laprie and T. Sabourin. "A Method for Software Reliability Growth Analysis and Assessment," in *Proc. Int. Workshop on Software Engineering and its Applications,* pp. 859-878, Toulouse, France, 1988.

[Kanoun and Sabourin 1987] K. Kanoun and T. Sabourin. "Software Dependability of a Telephone Switching System," in *Proc. 17th IEEE Int. Symp. on Fault-Tolerant Computing (FTCS-17),* pp. 236-241, Pittsburg, PA, 1987.

[Keiller et al. 1983a] P.A. Keiller, B. Littlewood, D.R. Miller and A. Sofer. "Comparison of Software Reliability Predictions," in *Proc. 13th IEEE Int. Symp. on Fault-Tolerant Computing (FTCS-13),* pp. 128-134, Milan, IEEE Computer Society Press, 1983.

[Keiller et al. 1983b] P.A. Keiller, B. Littlewood, D.R. Miller and A. Sofer. "On the Quality of Software Reliability Predictions," in *Proc. NATO ASI,* pp. 441-460, Springer, 1983.

[Kendall and Stuart 1979] M. Kendall and A. Stuart. *The Advanced Theory of Statistics: Inference and Relationship,* London, Charles Griffin & Co. Ltd., 1979.

[Khoshgoftaar and Munson 1990] T.M. Khoshgoftaar and J.C. Munson. "Predicting Software Development Errors Using Software Complexity Metrics," *IEEE Journal on Selected Areas in Communications*, vol. 8, no. 2, pp.253-261, 1990.

[Khoshgoftaar et al. 1992] T.M. Khoshgoftaar, J.C. Munson, B.B. Bhattacharya and G.D. Richardson. "Predictive Modeling Techniques of Software Quality from Software Measures," *IEEE Trans. on Software Engineering*, vol. 18, no. 11, pp.979-987, 1992.

[Knafl 1992] G.J. Knafl. *Solving Maximum Likelihood Equations for Two-parameter Models using Grouped Data,* Technical Report, Dept. of Computer Science and Information Systems, DePaul University, 1992.

[Laprie 1984] J.C. Laprie. "Dependability Evaluation of Software Systems in Operation," *IEEE Trans. on Software Engineering*, vol. SE-10, no. 6, pp.701-714, 1984.

[Laprie 1992] J.C. Laprie, (Ed.). *Dependability: Basic Concepts and Terminology,* Dependable Computing and Fault-Tolerant Systems, Wein, New York, Springer-Verlag, 1992.

[Lee and Anderson 1990] P.A. Lee and T. Anderson. *Fault Tolerance: Principles and Practice,* Dependable Computing and Fault-Tolerant Systems, Wein, New York, Springer-Verlag, 1990.

[Littlewood 1979a] B. Littlewood. "How to Measure Software Reliability and How Not To," *IEEE Trans. on Reliability*, vol. R-28, no. 2, pp.103-110, 1979.

[Littlewood 1979b] B. Littlewood. "A Software Reliability Model for Modular Program Structure," *IEEE Trans. on Reliability*, vol. R-28, no. 2, pp.241-246, 1979.

[Littlewood 1981] B. Littlewood. "Stochastic Reliability Growth: A Model for Fault-Removal in Computer-Programs and Hardware-Designs," *IEEE Trans. on Reliability*, vol. R-30, no. 4, pp.313-320, 1981.

[Littlewood 1988] B. Littlewood. "Forecasting software reliability," in *Software Reliability Modelling and Identification, Lecture Notes in Computer Science 341, ed. S. Bittanti*, pp. 141-209, Heidelberg, Springer, 1988.

[Littlewood 1989] B. Littlewood. "Predicting Software Reliability," *Phil Trans. Royal Soc. London*, vol. A 327, pp.513-527, 1989.

[Littlewood et al. 1994] B. Littlewood, S. Brocklehurst, N.E. Fenton, P. Mellor, S. Page, D. Wright, J.E. Dobson, J.A. McDermid and D. Gollmann. "Towards operational measures of computer security," *Journal of Computer Security*, vol. 2, no. 3, 1994.

[Littlewood and Keiller 1984] B. Littlewood and P.A. Keiller. "Adaptive Software Reliability Modelling," in *Proc. 14th IEEE Int. Symp. on Fault-Tolerant Computing (FTCS-14),* pp. 108-113, 1984.

[Littlewood and Strigini 1992] B. Littlewood and L. Strigini. "The Risks of Software," *Scientific American*, vol. 267, no. 5, pp.62-75, 1992.

[Littlewood and Strigini 1993] B. Littlewood and L. Strigini. "Validation of Ultra-High Dependability for Software-based Systems," *Communications of the ACM*, vol. 36, no. 11,1993.

[Littlewood and Verrall 1973] B. Littlewood and J.L. Verrall. "A Bayesian Reliability Growth Model for Computer Software," *J Royal Statist Soc C*, vol. 22, pp.332-346, 1973.

[Littlewood and Verrall 1981] B. Littlewood and J.L. Verrall. "Likelihood Function of a Debugging Model for Computer Software Reliability," *IEEE Trans. on Reliability*, vol. R-30, no. 2, pp.145-148, 1981.

[Lu and Brocklehurst 1991] M. Lu and S. Brocklehurst. *Combination of Predictions Obtained from Different Software Reliability Growth Models,* PDCS1 (ESPRIT Project 3092) Technical Report number 56, Centre for Software Reliability, City University, London, 1991.

[Lu et al. 1993] M. Lu, S. Brocklehurst and B. Littlewood. "Combination of Predictions Obtained from Different Software Reliability Growth Models," *Journal of Computer & Software Engineering Special Issue on Reliable Software*, vol. 1, no. 4, pp.303-323, 1993.

[Lyu and Nikora 1992] M. Lyu and A. Nikora. "Applying Reliability Models More Effectively," *IEEE Software special issue on Reliability Measurement*, pp.43-52, July, 1992.

[Martini et al. 1990] M.R.B. Martini, K. Kanoun and J.M. d. Souza. "Software Reliability Evaluation of the TROPICO-R Switching System," *IEEE Trans. on Reliability*, vol. R-39, no. 3, pp.369-379, 1990.

[Mellor 1986] P. Mellor. "Software Reliability Data Collection: Problems and Standards," in *Software Reliability: A state of the Art Report, ed. P. Mellor*, pp. 165-181, London, Pergamon Infotech Ltd., 1986.

[Miller 1986] D.R. Miller. "Exponential Order Statistic Models of Software Reliability Growth," *IEEE Trans. on Software Engineering*, vol. SE-12, no. 1, pp.12-24, 1986.

[Miller and Sofer 1985] D.R. Miller and A. Sofer. "Completely Monotone Regression Estimates of Software Failure Rates," in *Proc. Eighth International Conference on Software Engineering,* pp. 343-348, Washington D.C., 1985.

[Miller and Sofer 1986a] D.R. Miller and A. Sofer. "Least-Squares Regression under Convexivity and Higher Order Difference Constraints with Application to Software Reliability," in *Advances in Order Restricted Inference, ed. R. Dykstra, T. Robertson and F. T. Wright,* pp. Springer Verlag, 1986.

[Miller and Sofer 1986b] D.R. Miller and A. Sofer. "A Non-Parametric Approach to Software Reliability, Using Complete Monotonicity," in *Software Reliability: A State of the Art Report, ed. P. Mellor,* pp. 185-195, London, Pergamon Infotech Ltd., 1986.

[Miller and Sofer 1988] D.R. Miller and A. Sofer. *A Non-Parametric Software Reliability Growth Model,* Technical Report 37, Center for Computational Statistics and Probability, 1988.

[Miller 1956] L.H. Miller. "Table of Percentage Points of Kolmogorov Statistics," *American Statistical Association Journal,* vol. 51, pp.111-121, 1956.

[Musa 1975] J.D. Musa. "A Theory of Software Reliability and Its Application," *IEEE Trans. on Software Engineering,* vol. SE-1, no. 3, pp.312-327, 1975.

[Musa et al. 1987] J.D. Musa, A. Iannino and K. Okumoto. *Software Reliability: Measurement, Prediction, Application,* New York, McGraw-Hill, 1987.

[Musa and Okumoto 1984] J.D. Musa and K. Okumoto. "A Logarithmic Poisson Execution Time Model for Software Reliability Measurement," in *Proc. 7th Int. Conf. on Software Eng.,* pp. 230-238, Chicago, IEEE Computer Society Press, 1984.

[Neumann] P. Neumann, (Ed.). *Forum on Risks to the Public in Computers and Related Systems* available as the usenet newsgroup comp.risks.

[Perrow 1984] C. Perrow. *Normal Accidents: Living with High Risk Technologies,* New York, Basic Books Inc., 1984.

[Potter 1988] M. Potter. *Dataset 5 Mappings for SRM Database,* Alvey Software Reliability Modelling Project (PRJ/SE/072) Task 9 deliverable, 1988.

[Potter 1989] M. Potter. *Software Reliability Modelling Database Technical Manual,* Alvey Software Reliability Modelling Project (PRJ/SE/072) Task 9 deliverable, Issue 1.1, 1989.

[RSCL 1988] RSCL. *Software Reliability Modelling Programs,* Version 1.0, Reliability and Statistical Consultants Ltd., 1988.

[Saglietti 1989] F. Saglietti. *Combination of Predictions Obtained by Different Software Reliability Growth Models,* Technical Report REQUEST/GRS-saf/053/S2/RL-WP/00, Gesellschaft fur Reaktorsicherheit (GRS) mbH, Forschungsgelande, 8046 Garching, 1989.

[Shaw 1984] M. Shaw. "Abstraction Techniques in Modern Programming Languages," *IEEE Software*, pp.10-26, October 1984.

[Simmonds 1988a] I. Simmonds. *Design of a Relational Database to Hold Software Reliability Data,* Alvey Software Reliability Modelling Project (PRJ/SE/072) Task 9 deliverable, Issue 1.2, 1988.

[Simmonds 1988b] I. Simmonds. *Software Reliability Modelling Database: User Manual,* Alvey Software Reliability Modelling Project (PRJ/SE/072) Task 9 deliverable, Issue 1, 1988.

[Wright 1988] D.R. Wright. *A modified u-plot applied to failure count prediction,* Technical Report, Centre for Software Reliability, City University, London, 1988.

[Wright 1993] D.R. Wright. *Recalibrated Prediction of some Software Failure-Count Sequences,* PDCS2 Project (ESPRIT Project 6362) First Year Report, chapter 4, part 4.1, Centre for Software Reliability, City University, London, 1993.

[Xie 1991] M. Xie. *Software Reliability Modelling,* Singapore, World Scientific Publishing Co. Pte. Ltd., 1991.

Appendix A Non-Parametric Model Details

To summarise we are using inter-failure time data, $t_{j-1} = \langle t_1, t_2, ..., t_{j-1} \rangle$, to estimate the associated rates, $r_{j-1} = \langle r_1, r_2, ..., r_{j-1} \rangle$. In order to make one-step-ahead predictions about T_j, it is then assumed that T_j has an exponential distribution with rate the same as the most recently estimated rate, $\tilde{r}_{j-1}$, i.e.,

$$\hat{F}_j(t) = 1 - e^{-\tilde{r}_{j-1}t} \qquad \ldots\ldots\ldots (A.1)$$

and

$$\hat{f}_j(t) = \tilde{r}_{j-1}\, e^{-\tilde{r}_{j-1}t} \qquad \ldots\ldots\ldots (A.2)$$

A.1 OTY Model

A.1.1 Formulation

From *(5.2.2)* and from the construction of the *y*-plot as described in section 3.2 the maximum distances of the top step and the bottom step of the *y*-plot from the *45°* line are

$$D^+(r_{j-1},t_{j-1}) = \max_{1 \le m \le j-1} \left| \frac{\sum_{k=1}^{m} r_k t_k}{\sum_{k=1}^{j-1} r_k t_k} - \frac{m}{j-1} \right| \qquad \ldots\ldots (A.1.1)$$

$$D^-(r_{j-1},t_{j-1}) = \max_{1 \le m \le j-1} \left| \frac{\sum_{k=1}^{m} r_k t_k}{\sum_{k=1}^{j-1} r_k t_k} - \frac{m-1}{j-1} \right| \qquad \ldots\ldots (A.1.2)$$

respectively and the linear programming problem becomes $\underset{r_{j-1}}{Minimise}\ [K(r_{j-1},t_{j-1})]$ subject to *(5.2.1)* and *(5.2.3)* where

$$K(r_{j-1},t_{j-1}) = max[D^+(r_{j-1},t_{j-1}),D^-(r_{j-1},t_{j-1})] \qquad \ldots\ldots (A.1.3)$$

($K(r_{j-1},t_{j-1})$ is the K distance of the *y*-plot from the *45°* line).

The equality constraint *(5.2.3)* allows us to transform the problem to be linear in the variables, r_{j-1}, and if we let

$$O_y(r_{j-1},t_{j-1}) = (j-1)K(r_{j-1},t_{j-1}) \qquad \text{.....(A.1.4)}$$

then the problem becomes

$$\underset{r_{j-1}}{Minimise}\ [O_y(r_{j-1},t_{j-1})] \text{ subject to}$$

$$O_y(r_{j-1},t_{j-1}) \geq 0 \qquad \text{.....(A.1.5)}$$

$$-O_y(r_{j-1},t_{j-1}) \leq \sum_{s=1}^{k} r_s t_s - k \leq O_y(r_{j-1},t_{j-1}) \qquad k = 1, ..., j-1 \qquad \text{.....(A.1.6)}$$

$$-O_y(r_{j-1},t_{j-1}) \leq \sum_{s=1}^{k} r_s t_s - k + 1 \leq O_y(r_{j-1},t_{j-1}) \qquad k = 1, ..., j-1 \qquad \text{.....(A.1.7)}$$

and subject to *(5.2.1)* and *(5.2.3)* which reduces to

$$\underset{r_{j-1}}{Minimise}\ [O_y(r_{j-1},t_{j-1})] \text{ subject to}$$

$$O_y(r_{j-1},t_{j-1}) \geq 0 \qquad \text{.....(A.1.8)}$$

$$r_k \geq 0 \qquad k = 1, ..., j-1 \qquad \text{.....(A.1.9)}$$

$$r_k - r_{k-1} \leq 0 \qquad k = 2, ..., j-1 \qquad \text{....(A.1.10)}$$

$$r_k - 2r_{k-1} + r_{k-2} \geq 0 \qquad k = 3, ..., j-1 \qquad \text{....(A.1.11)}$$

$$\sum_{s=1}^{k} r_s t_s + O_y(r_{j-1},t_{j-1}) \geq k \qquad k = 1, ..., j-1 \qquad \text{....(A.1.12)}$$

$$\sum_{s=1}^{k} r_s t_s - O_y(r_{j-1},t_{j-1}) \leq k - 1 \qquad k = 1, ..., j-1 \qquad \text{....(A.1.13)}$$

$$\sum_{k=1}^{j-1} r_k t_k = j-1 \qquad \text{....(A.1.14)}$$

A.1.2 Solution Classifications

To summarise let

$$r_{j-1}^{h} = \frac{j-1}{\tau_{j-1}} \text{ where } \tau_{j-1} = \sum_{k=1}^{j-1} t_k \text{ and } r_{j-1}^{h} = \langle r_1, ..., r_{j-1} \rangle;\ r_k = r_{j-1}^{h},\ k = 1, ..., j-1 \quad \text{...(A.1.15)}$$

$$D^{+}(1_{j-1},t_{j-1}) = \underset{1 \leq m \leq j-1}{Max} \left| \frac{\tau_m}{\tau_{j-1}} - \frac{m}{j-1} \right| \qquad \text{....(A.1.16)}$$

$$D^{-}(1_{j-1},t_{j-1}) = \underset{1 \le m \le j-1}{Max} \left| \frac{\tau_m}{\tau_{j-1}} - \frac{m-1}{j-1} \right| \qquad \ldots.(A.1.17)$$

An alternative formulation for the *OTY* linear programming problem *(A.1.8)* - *(A.1.14)* is

$$\underset{r_{j-1}}{Minimise}\ [K(r_{j-1},t_{j-1})] \text{ (as defined in } (A.1.1)\text{-}(A.1.3)) \text{ subject to}$$

$$r_k \ge 0 \qquad k = 1, \ldots, j-1 \qquad \ldots.(A.1.18)$$

$$r_k - r_{k-1} \le 0 \qquad k = 2, \ldots, j-1 \qquad \ldots.(A.1.19)$$

$$r_k - 2r_{k-1} + r_{k-2} \ge 0 \qquad k = 3, \ldots, j-1 \qquad \ldots.(A.1.20)$$

$$\sum_{k=1}^{j-1} r_k t_k = j-1 \qquad \ldots.(A.1.21)$$

Theorem A.1.1

The *HPP* vector of rates, r^{h}_{j-1}, is always feasible.

Proof

From *(A.1.15)*, since $r_k = r^{h}_{j-1}$ for all $k = 1, \ldots, j-1$, trivially constraints *(A.1.19)* and *(A.1.20)* are satisfied and since $r^{h}_{j-1} = \frac{j-1}{\tau_{j-1}}$ constraints *(A.1.18)* and *(A.1.21)* are also satisfied. Thus, r^{h}_{j-1} is always a feasible solution given any vector of inter-failure times, t_{j-1}.

Theorem A.1.2

If $D^{-}(1_{j-1},t_{j-1}) \ge D^{+}(1_{j-1},t_{j-1})$ then the *HPP* vector of rates, r^{h}_{j-1}, is feasible, and optimal and additionally if $D^{-}(1_{j-1},t_{j-1}) = \left| \frac{\tau_n}{\tau_{j-1}} - \frac{n-1}{j-1} \right|$ and $n \ne j-1$ and $t_k \ne 0$ for all $k = 1, \ldots, j-1$, then r^{h}_{j-1} is uniquely optimal.

Proof

Let $D^{-}(1_{j-1},t_{j-1}) \ge D^{+}(1_{j-1},t_{j-1})$ $\qquad \ldots.(A.1.22)$

Then from *(A.1.16)* and *(A.1.17)* there exists an $n \in \{1, 2, ..., j-1\}$ such that

$$D^-(1_{j-1},t_{j-1}) = \frac{1}{(j-1)\tau_{j-1}} |(j-1)\tau_n - (n-1)\tau_{j-1}| \geq \frac{1}{(j-1)\tau_{j-1}} |(j-1)\tau_n - n\tau_{j-1}| \quad \ldots(A.1.23)$$

and so $$D^-(1_{j-1},t_{j-1}) = \frac{1}{(j-1)\tau_{j-1}}((j-1)\tau_n - (n-1)\tau_{j-1}) \geq 0 \quad \ldots.(A.1.24)$$

From Theorem A.1.1 r^h_{j-1} is trivially feasible and from *(A.1.1)*, *(A.1.2)*, *(A.1.3)*, *(A.1.15)*, *(A.1.16)*, *(A.1.17)*, *(A.1.22)* and *(A.1.24)*

$$K(r^h_{j-1},t_{j-1}) = D^-(1_{j-1},t_{j-1}) = \frac{1}{(j-1)\tau_{j-1}}((j-1)\tau_n - (n-1)\tau_{j-1}) \quad \ldots.(A.1.25)$$

Let $r_{j-1} = \langle r_1, ..., r_{j-1} \rangle$ be another feasible solution (distinct from r^h_{j-1}). Then, from *(A.1.19)*

$$r_k - r_{k-1} = -c_k \qquad c_k \geq 0,\ k = 2, ..., j-1 \quad \ldots.(A.1.26)$$

and from *(A.2.21)* since r_{j-1} is distinct from r^h_{j-1}, $c_k > 0$ for at least one $k \in \{2, ..., j-1\}$.

Then from *(A.1.21)* and *(A.1.26)*

$$r_k = \frac{\left((j-1) - \left(\sum_{s=1}^{j-2} c_{s+1}\tau_s\right) + \tau_{j-1}\left(\sum_{s=k+1}^{j-1} c_s\right)\right)}{\tau_{j-1}} \qquad k = 1, ..., j-1 \quad \ldots.(A.1.27)$$

So from *(A.1.2)*, *(A.1.3)*, *(A.1.21)* and *(A.1.27)*

$$K(r_{j-1},t_{j-1}) \geq D^-(r_{j-1},t_{j-1}) \geq \frac{1}{j-1}\left|\sum_{k=1}^{n} r_k t_k - (n-1)\right|$$

$$= \frac{1}{(j-1)\tau_{j-1}}\left|(j-1)\tau_n - (n-1)\tau_{j-1} + \sum_{k=1}^{n}\left(\tau_{j-1}\left(\sum_{s=k+1}^{j-1} c_s\right) - \left(\sum_{s=1}^{j-2} c_{s+1}\tau_s\right)\right)t_k\right|$$

$$= \frac{1}{(j-1)\tau_{j-1}}\left|(j-1)\tau_n - (n-1)\tau_{j-1} + (\tau_{j-1}-\tau_n)\left(\sum_{k=1}^{n-1} c_{k+1}\tau_k\right) + \tau_n\left(\sum_{k=n}^{j-2} c_{k+1}(\tau_{j-1}-\tau_k)\right)\right|$$

$$= \frac{1}{(j-1)\tau_{j-1}}\left((j-1)\tau_n-(n-1)\tau_{j-1}+(\tau_{j-1}-\tau_n)\left(\sum_{k=1}^{n-1} c_{k+1}\tau_k\right)+\tau_n\left(\sum_{k=n}^{j-2} c_{k+1}(\tau_{j-1}-\tau_k)\right)\right) \quad \ldots.(A.1.28)$$

from *(A.1.24)* and *(A.1.26)*.

Finally, from *(A.1.25)* and *(A.1.28)*

$$K(r_{j-1},t_{j-1}) - K(r^{h}_{j-1},t_{j-1}) \geq \frac{1}{(j-1)\tau_{j-1}}\left((\tau_{j-1}-\tau_n)\left(\sum_{k=1}^{n-1} c_{k+1}\tau_k\right) + \tau_n\left(\sum_{k=n}^{j-2} c_{k+1}(\tau_{j-1}-\tau_k)\right)\right)$$

and so from *(A.1.26)* (since $c_k \geq 0$ for all $k = 2, ..., j-1$) $K(r_{j-1},t_{j-1}) \geq K(r^{h}_{j-1},t_{j-1})$ for all other feasible solutions r_{j-1} and so r^{h}_{j-1} is optimal. Further, provided $n \neq j-1$ and $t_k \neq 0$ for all $k = 1, ..., j-1$, then (since $c_k \geq 0$ for all $k = 2, ..., j-1$ and $c_k > 0$ for at least one $k \in \{2, ..., j-1\}$) $K(r_{j-1},t_{j-1}) > K(r^{h}_{j-1},t_{j-1})$ for all other feasible solutions r_{j-1} and so r^{h}_{j-1} is uniquely optimal.

Theorem A.1.3

If $D^{-}(1_{j-1},t_{j-1}) < D^{+}(1_{j-1},t_{j-1})$ then the *HPP* vector of rates, r^{h}_{j-1}, is feasible, but not optimal.

Proof

Let $D^{-}(1_{j-1},t_{j-1}) < D^{+}(1_{j-1},t_{j-1})$ ....*(A.1.29)*

Then from *(A.1.16)*, *(A.1.17)* and *(A.1.29)* there exists a set N, $\{1, 2, ..., j-1\} \supseteq N$, $sgn[N] \geq 1$, such that

$$D^{+}(1_{j-1},t_{j-1}) = \frac{1}{(j-1)\tau_{j-1}}\left|(j-1)\tau_{n'} - n'\tau_{j-1}\right| \qquad \text{for all } n' \in N \qquad ....(A.1.30)$$

$$D^{+}(1_{j-1},t_{j-1}) > \frac{1}{(j-1)\tau_{j-1}}\left|(j-1)\tau_{k} - (k-1)\tau_{j-1}\right| \qquad \text{for all } k \in \{1, 2, ..., j-1\} \qquad ....(A.1.31)$$

$$D^{+}(1_{j-1},t_{j-1}) > \frac{1}{(j-1)\tau_{j-1}}\left|(j-1)\tau_{k} - k\tau_{j-1}\right| \qquad \text{for all } k \in \{1, 2, ..., j-1\} - N \qquad ...(A.1.32)$$

Suppose $j-1 \in N$. Then from *(A.1.30)* $D^{+}(1_{j-1},t_{j-1}) = 0$ so from *(A.1.29)* $D^{-}(1_{j-1},t_{j-1}) < 0$ but from *(A.1.17)* $D^{-}(1_{j-1},t_{j-1}) > 0$ and so

$$\{1, 2, ..., j-2\} \supseteq N \qquad ....(A.1.33)$$

From *(A.1.30)* and *(A.1.31)*

$$\left|(j-1)\tau_{n'} - n'\tau_{j-1}\right| > \left|(j-1)\tau_{n'} - (n'-1)\tau_{j-1}\right| \qquad \text{for all } n' \in N$$

and so $$-\left((j-1)\tau_{n'} - n'\tau_{j-1}\right) > 0 \qquad \text{for all } n' \in N \qquad ....(A.1.34)$$

Further, from *(A.1.30)*, *(A.1.31)* and *(A.1.34)*

$$-\left((j-1)(\tau_{n'}+\tau_k) - (n'+k-1)\tau_{j-1}\right) > 0 \quad \text{for all } n' \in N \text{ and all } k \in \{1, ..., j-1\} \qquad ....(A.1.35)$$

and from *(A.1.30)*, *(A.1.32)* and *(A.1.34)*

$$-\left((j-1)(\tau_{n'}-\tau_k) - (n'-k)\tau_{j-1}\right) > 0 \quad \text{for all } n' \in N \text{ and all } k \in \{1, ..., j-1\}-N \qquad ....(A.1.36)$$

From Theorem A.1.1 r^h_{j-1} is trivially feasible and from *(A.1.1)*, *(A.1.2)*, *(A.1.3)*, *(A.1.15)*, *(A.1.16)*, *(A.1.17)* and *(A.1.29)*

$$K(r^h_{j-1}, t_{j-1}) = D^+(1_{j-1}, t_{j-1}) \qquad ....(A.1.37)$$

Consider the following alternative solution, r_{j-1}, defined as follows.

$$\text{Let} \quad a_k = (j-k-1)\tau_{j-1}\tau_k + \tau_{j-1}\sum_{s=1}^{k-1}\tau_s - \tau_k\sum_{s=1}^{j-2}\tau_s \qquad \text{for } k = 1, ..., j-1 \qquad ....(A.1.38)$$

which can be written as

$$a_k = (\tau_{j-1}-\tau_k)\sum_{s=1}^{k-1}\tau_s + \tau_k\sum_{s=k}^{j-2}(\tau_{j-1}-\tau_s) \qquad \text{for } k = 1, ..., j-1 \qquad ....(A.1.39)$$

$$\text{and so} \qquad a_k > 0 \qquad \text{for all } k = 1, ..., j-2 \text{ and } a_{j-1} = 0 \qquad ....(A.1.40)$$

Choose $n \in N$ so that

$$a_n - a_{n'} \leq 0 \qquad \text{for all } n' \in N \qquad ....(A.1.41)$$

Let

$$c = \min\left(\frac{j-1}{j-2}; B_n; C_n\right) \qquad ....(A.1.42)$$
$$\sum_{s=1}\tau_s$$

$$\text{where} \qquad B_n = \min_{m \in \{1 \ldots j-1\}} \left(\frac{-\left((j-1)(\tau_n+\tau_m) - (n+m-1)\tau_{j-1}\right)}{a_n + a_m}\right)$$

$$\text{and} \qquad C_n = \min_{(m \in \{1 \ldots j-1\}-N \mid a_n-a_m>0)} \left(\frac{-\left((j-1)(\tau_n-\tau_m) - (n-m)\tau_{j-1}\right)}{a_n - a_m}\right)$$

if there exists an $m \in \{1, ..., j-1\}-N$ such that $a_n - a_m > 0$ and $C_n = \frac{j-1}{j-2}$, otherwise.
$$\sum_{s=1}\tau_s$$

$$\text{Let} \qquad r_k - r_{k-1} = -c \qquad k = 2, ..., j-1 \qquad ....(A.1.43)$$

and $$r_{j-1} = \frac{j-1 - c\sum_{s=1}^{j-2}\tau_s}{\tau_{j-1}} \qquad \ldots.(A.1.44)$$

Then from *(A.1.43)* and *(A.1.44)*

$$r_k = r_{j-1} + (j-k-1)c \qquad k = 1, \ldots, j-1 \qquad \ldots.(A.1.45)$$

From *(A.1.33)*, *(A.1.35)*, *(A.1.40)* and *(A.1.42)* $B_n > 0$. From *(A.1.36)* and *(A.1.42)* if there exists an $m \in \{1, \ldots, j-1\}$-N such that $a_n - a_m > 0$ then $C_n > 0$. So, from *(A.1.42)* c is bounded ($c \leq \frac{j-1}{j-2}$) and $c > 0$. Thus, from *(A.1.43)* constraints *(A.1.19)* and *(A.1.20)* are satisfied. From *(A.1.42)* and *(A.1.44)* $r_{j-1} \geq 0$ and so from *(A.1.45)* $r_k \geq 0$ for all $k = 1, \ldots, j-1$ and so constraint *(A.1.18)* is satisfied. From *(A.1.44)* and *(A.1.45)* constraint *(A.1.21)* is satisfied. Finally, since $c > 0$, from *(A.1.45)* $r_{j-1} \neq r^h_{j-1}$. Thus we have found another feasible solution, r_{j-1}, which is distinct from r^h_{j-1}.

From *(A.1.1)*, *(A.1.2)*, *(A.1.3)*, *(A.1.21)*, *(A.1.38)*, *(A.1.44)* and *(A.1.45)*

$$K(r_{j-1},t_{j-1}) = max[D^+(r_{j-1},t_{j-1}),D^-(r_{j-1},t_{j-1})] \qquad \ldots.(A.1.46)$$

where

$$D^+(r_{j-1},t_{j-1}) = \frac{1}{(j-1)\tau_{j-1}} \underset{1 \leq m \leq j-1}{Max} |(j-1)\tau_m - m\tau_{j-1} + ca_m|$$

$$D^-(r_{j-1},t_{j-1}) = \frac{1}{(j-1)\tau_{j-1}} \underset{1 \leq m \leq j-1}{Max} |(j-1)\tau_m - (m-1)\tau_{j-1} + ca_m|$$

But from *(A.1.33)*, *(A.1.40)* and *(A.1.42)*

$$-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n \geq 0 \qquad \ldots.(A.1.47)$$

$$-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n \geq \left((j-1)\tau_k - (k-1)\tau_{j-1}\right) + ca_k \qquad \text{for all } k = 1, \ldots, j-1 \ldots(A.1.48)$$

and so

$$-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n \geq \left((j-1)\tau_k - k\tau_{j-1}\right) + ca_k \qquad \text{for all } k = 1, \ldots, j-1 \quad \ldots.(A.1.49)$$

Also, from *(A.1.33)*, *(A.1.40)* and *(A.1.42)*

$$-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n \geq -\left((j-1)\tau_k - k\tau_{j-1}\right) - ca_k \qquad \text{for all } k \in \{1, \ldots, j-1\}\text{-}N \text{ for which } a_n - a_k > 0 \qquad \ldots.(A.1.50)$$

Consider $k \varepsilon \{1, ..., j-1\}$-$N$ for which $a_n - a_k \leq 0$. Since $c > 0$, and from *(A.1.36)* we have

$$-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n \geq -\left((j-1)\tau_k - k\tau_{j-1}\right) - ca_k \quad \text{for all } k \varepsilon \{1, ..., j-1\}\text{-}N \text{ for which } a_n - a_k \leq 0 \quad \ldots.(A.1.51)$$

Now consider $n' \varepsilon N$. From *(A.1.30)*, *(A.1.34)*, *(A.1.41)* and since $c > 0$ we have

$$-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n \geq -\left((j-1)\tau_{n'} - n'\tau_{j-1}\right) - ca_{n'} \quad \text{for all } n' \varepsilon N \quad \ldots.(A.1.52)$$

So from *(A.1.50)*, *(A.1.51)* and *(A.1.52)*

$$-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n \geq -\left((j-1)\tau_k - k\tau_{j-1}\right) - ca_k \quad \text{for all } k \varepsilon \{1, ..., j-1\} \quad \ldots.(A.1.53)$$

and so

$$-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n \geq -\left((j-1)\tau_k - (k-1)\tau_{j-1}\right) - ca_k \quad \text{for all } k \varepsilon \{1, ..., j-1\} \ldots.(A.1.54)$$

So from *(A.1.46)*, *(A.1.48)* , *(A.1.49)*, *(A.1.53)* and *(A.1.54)*

$$K(r_{j-1},t_{j-1}) = \frac{-\left((j-1)\tau_n - n\tau_{j-1}\right) - ca_n}{(j-1)\tau_{j-1}} \quad \ldots.(A.1.55)$$

So, finally, from *(A.1.30)*, *(A.1.33)*, *(A.1.34)*, *(A.1.37)*, *(A.1.40)* and *(A.1.55)* (and since $c > 0$) we have

$$K(r^{h}_{j-1},t_{j-1}) - K(r_{j-1},t_{j-1}) = \frac{ca_n}{(j-1)\tau_{j-1}} > 0$$

Thus, we have found an alternative feasible solution r_{j-1} which has a smaller objective function than r^{h}_{j-1} and hence r^{h}_{j-1} is not optimal.

A.2 OTL Model

A.2.1 Formulation

To summarise the *OTL* problem is $\underset{r_{j-1}}{Minimise} |L(r_{j-1},t_{j-1})|$ subject to the difference constraints *(5.2.1)* and the scaling constraint *(5.2.3)* where

$$L(r_{j-1},t_{j-1}) = \frac{\sum_{k=1}^{j-1} r_k t_k}{2} - \frac{\sum_{k=1}^{j-2}\sum_{s=1}^{k} r_s t_s}{j-2} \quad \ldots\ldots(A.2.1)$$

From *(A.2.1)* and *(5.2.3)*

$$L(r_{j-1},t_{j-1}) = \frac{(j-1)(j-2) - 2\sum_{k=1}^{j-2}(j-k-1)r_k t_k}{2(j-2)} \qquad \ldots\ldots(A.2.2)$$

so if we let $O_l(r_{j-1},t_{j-1}) = 2(j-2)|L(r_{j-1},t_{j-1})|$ then from *(5.2.1)*, *(5.2.3)* and *(A.2.2)* the problem becomes

$$\underset{r_{j-1}}{Minimise}\ [O_l(r_{j-1},t_{j-1})] \text{ subject to}$$

$$O_l(r_{j-1},t_{j-1}) \geq 0 \qquad \ldots\ldots(A.2.3)$$

$$r_k \geq 0 \qquad k = 1, \ldots, j-1 \qquad \ldots\ldots(A.2.4)$$

$$r_k - r_{k-1} \leq 0 \qquad k = 2, \ldots, j-1 \qquad \ldots\ldots(A.2.5)$$

$$r_k - 2r_{k-1} + r_{k-2} \geq 0 \qquad k = 3, \ldots, j-1 \qquad \ldots\ldots(A.2.6)$$

$$2\sum_{k=1}^{j-2}(j-k-1)r_k t_k + O_l(r_{j-1},t_{j-1}) \geq (j-1)(j-2) \qquad \ldots\ldots(A.2.7)$$

$$2\sum_{k=1}^{j-2}(j-k-1)r_k t_k - O_l(r_{j-1},t_{j-1}) \leq (j-1)(j-2) \qquad \ldots\ldots(A.2.8)$$

$$\sum_{k=1}^{j-1} r_k t_k = j-1 \qquad \ldots\ldots(A.2.9)$$

A.2.2 Solution Classifications

To summarise let

$$r^h_{j-1} = \frac{j-1}{\tau_{j-1}} \text{ where } \tau_{j-1} = \sum_{k=1}^{j-1} t_k \text{ and } r^h_{j-1} = \langle r_1, \ldots, r_{j-1}\rangle;\ r_k = r^h_{j-1},\ k = 1, \ldots, j-1 \ \ldots(A.2.10)$$

and $$\gamma_k = \sum_{s=1}^{k}\tau_s \text{ where } \tau_k = \sum_{s=1}^{k} t_s \qquad \ldots.(A.2.11)$$

and $$L(1,t_{j-1}) = \frac{\tau_{j-1}}{2} - \frac{\gamma_{j-2}}{j-2} \qquad \ldots.(A.2.12)$$

$$l(r_{j-1},t_{j-1}) = |L(r_{j-1},t_{j-1})| = \left|\frac{\sum_{k=1}^{j-1} r_k t_k}{2} - \frac{\sum_{k=1}^{j-2}\sum_{s=1}^{k} r_s t_s}{j-2}\right| \qquad \ldots.(A.2.13)$$

Then an alternative formulation for the *OTL* linear programming problem *(A.2.3)* - *(A.2.9)* is

$$\underset{r_{j-1}}{Minimise}\ [l(r_{j-1},t_{j-1})] \text{ subject to}$$

$$r_k \geq 0 \qquad k = 1, \ldots, j-1 \qquad \ldots.(A.2.14)$$

$$r_k - r_{k-1} \leq 0 \qquad k = 2, \ldots, j-1 \qquad \ldots.(A.2.15)$$

$$r_k - 2r_{k-1} + r_{k-2} \geq 0 \qquad k = 3, \ldots, j-1 \qquad \ldots.(A.2.16)$$

$$\sum_{k=1}^{j-1} r_k t_k = j-1 \qquad \ldots.(A.2.17)$$

Theorem A.2.1

If $L(1,t_{j-1}) > 0$ then there exists a feasible optimal solution, r_{j-1}, such that $L(r_{j-1},t_{j-1}) = 0$ and $\Delta r_k = -c < 0$, for all $k \in \{2, \ldots, n-1\}$ and $\Delta r_k = 0$ for all $k \in \{n, \ldots, j-1\}$.

Proof

Let $L(1,t_{j-1}) > 0$ $\qquad \ldots.(A.2.18)$

Choose a vector of rates, r_{j-1}, as follows. Choose $n \in \{1, \ldots, j-1\}$ such that

$$3 \leq n \leq \frac{j+4}{2} \qquad \ldots.(A.2.19)$$

and let $$c = \frac{(j-1)(j-2)L(1,t_{j-1})}{n-3} \Big/ \left[(j-n+1)\tau_{j-1}\gamma_{n-2} + 2\tau_{j-1}\sum_{k=1}\gamma_k - \gamma_{j-2}\gamma_{n-2}\right] \qquad \ldots.(A.2.20)$$

and $$r_k = \begin{cases} \dfrac{j-1 - c\gamma_{n-2} + (n-k-1)c\tau_{j-1}}{\tau_{j-1}} & k = 1 \ldots n-1 \\ \dfrac{j-1 - c\gamma_{n-2}}{\tau_{j-1}} & k = n \ldots j-1 \end{cases} \qquad \ldots.(A.2.21)$$

Then $$r_k - r_{k-1} = \begin{cases} -c & k = 2 \ldots n-1 \\ 0 & k = n \ldots j-1 \end{cases} \qquad \ldots.(A.2.22)$$

and $$r_k - 2r_{k-1} - r_{k-2} = \begin{cases} c & k = n \\ 0 & k = 3 \ldots j-1 \ k \neq n \end{cases} \qquad \ldots.(A.2.23)$$

Now from *(A.2.12)*

$$(j\text{-}n\text{+}1)\tau_{j-1}\gamma_{n-2} + 2\tau_{j-1}\sum_{k=1}^{n-3}\gamma_k - \gamma_{j-2}\gamma_{n-2}$$

$$= \frac{\gamma_{j-2}}{j-2}\left((j\text{-}2n\text{+}4)\gamma_{n-2} + 4\sum_{k=1}^{n-3}\gamma_k\right) + 2L(1,t_{j-1})\left((j\text{-}n\text{+}1)\gamma_{n-2} + 2\sum_{k=1}^{n-3}\gamma_k\right) > 0 \qquad \ldots.(A.2.24)$$

from *(A.2.18)* and *(A.2.19)*. So, from *(A.2.18)*, *(A.2.20)* and *(A.2.24)*, $c > 0$ and c is bounded. Further, from *(A.2.12)*, *(A.2.20)*, *(A.2.21)* and *(A.2.24)*

$$r_{j-1} = \frac{(j\text{-}1)\left((j\text{-}2n\text{+}4)\gamma_{n-2} + 4\sum_{k=1}^{n-3}\gamma_k\right)}{2\left((j\text{-}n\text{+}1)\tau_{j-1}\gamma_{n-2} + 2\tau_{j-1}\sum_{k=1}^{n-3}\gamma_k - \gamma_{j-2}\gamma_{n-2}\right)} \qquad \ldots.(A.2.25)$$

so from *(A.2.19)*, *(A.2.24)* and *(A.2.25)* $r_{j-1} \geq 0$ and r_{j-1} is bounded. So from *(A.2.21)*, *(A.2.22)* and *(A.2.23)* constraints *(A.2.14)*, *(A.2.15)*, *(A.2.16)* and *(A.2.17)* are satisfied and so r_{j-1}, as defined above, is a feasible solution.

From *(A.2.12)*, *(A.2.13)*, *(A.2.17)*, *(A.2.20)* and *(A.2.21)*

$$l(r_{j-1},t_{j-1}) = \left|\frac{(j\text{-}1)(j\text{-}2)L(1,t_{j-1}) + c\left(\gamma_{j-2}\gamma_{n-2} - (j\text{-}n\text{+}1)\tau_{j-1}\gamma_{n-2} - 2\tau_{j-1}\sum_{k=1}^{n-3}\gamma_k\right)}{(j\text{-}2)\tau_{j-1}}\right| = 0$$

and so we have defined a feasible solution r_{j-1} with zero objective and so r_{j-1} is an optimum solution.

Theorem A.2.2

If $L(1,t_{j-1}) > 0$ and $4\sum_{k=1}^{j-3}\gamma_k \geq (j\text{-}4)\gamma_{j-2}$ then there exists a feasible optimal solution, r_{j-1}, such that $L(r_{j-1},t_{j-1}) = 0$ and $\Delta r_k = -c < 0$ for all $k \in \{2, \ldots, j\text{-}1\}$.

Proof

Let $L(1,t_{j-1}) > 0$ $\qquad \ldots.(A.2.26)$

and $4\sum_{k=1}^{j-3}\gamma_k \geq (j\text{-}4)\gamma_{j-2}$ $\qquad \ldots.(A.2.27)$

Choose a vector of rates, r_{j-1}, as follows.

Let $$c = \frac{(j-1)(j-2)L(1,t_{j-1})}{\tau_{j-1}\gamma_{j-2} + 2\tau_{j-1}\sum_{k=1}^{j-3}\gamma_k - \gamma_{j-2}^2} \qquad \ldots.(A.2.28)$$

and $$r_k = \frac{j-1 + c\left((j-k-1)\tau_{j-1} - \gamma_{j-2}\right)}{\tau_{j-1}} \qquad k = 1, \ldots, j-1 \qquad \ldots.(A.2.29)$$

Then $$r_k - r_{k-1} = -c \qquad k = 2, \ldots, j-1 \qquad \ldots.(A.2.30)$$

and $$r_k - 2r_{k-1} - r_{k-2} = 0 \qquad k = 3, \ldots, j-1 \qquad \ldots.(A.2.31)$$

Now, from *(A.2.12)*,

$$\tau_{j-1}\gamma_{j-2} + 2\tau_{j-1}\sum_{k=1}^{j-3}\gamma_k - \gamma_{j-2}^2$$

$$= \frac{\gamma_{j-2}\left(4\sum_{k=1}^{j-3}\gamma_k - (j-4)\gamma_{j-2}\right)}{j-2} + 2L(1,t_{j-1})\left(\gamma_{j-2} + 2\sum_{k=1}^{j-3}\gamma_k\right) > 0 \qquad \ldots.(A.2.32)$$

from *(A.2.26)* and *(A.2.27)*. So from *(A.2.26)*, *(A.2.28)* and *(A.2.32)* $c > 0$ and c is bounded. Further, from *(A.2.12)*, *(A.2.28)*, *(A.2.29)* and *(A.2.32)*

$$r_{j-1} = \frac{(j-1)\left(4\sum_{k=1}^{j-3}\gamma_k - (j-4)\gamma_{j-2}\right)}{2\left(\tau_{j-1}\gamma_{j-2} + 2\tau_{j-1}\sum_{k=1}^{j-3}\gamma_k - \gamma_{j-2}^2\right)} \qquad \ldots.(A.2.33)$$

so from *(A.2.27)*, *(A.2.32)* and *(A.2.33)* $r_{j-1} \geq 0$ and r_{j-1} is bounded. So from *(A.2.29)*, *(A.2.30)* and *(A.2.31)* constraints *(A.2.14)*, *(A.2.15)*, *(A.2.16)* and *(A.2.17)* are satisfied and so r_{j-1}, as defined above, is a feasible solution.

From *(A.2.12)*, *(A.2.13)*, *(A.2.17)*, *(A.2.28)* and *(A.2.29)*

$$l(r_{j-1},t_{j-1}) = \left|\frac{(j-1)(j-2)L(1,t_{j-1}) + c\left(\gamma_{j-2}^2 - \tau_{j-1}\gamma_{j-2} - 2\tau_{j-1}\sum_{k=1}^{j-3}\gamma_k\right)}{(j-2)\tau_{j-1}}\right| = 0$$

and so we have defined a feasible solution r_{j-1} with zero objective and so r_{j-1} is an optimum solution.

Theorem A.2.3

If $L(1,t_{j-1}) > 0$ and $4\sum_{k=1}^{j-3}\gamma_k \le (j-4)\gamma_{j-2}$ then there exists a feasible optimal solution, r_{j-1}, such that $L(r_{j-1},t_{j-1}) = 0$, $r_{j-1} = 0$ and $\Delta r_k = -c-c'$ for all $k \in \{2, ..., n-1\}$ and $\Delta r_k = -c$ for all $k \in \{n, ..., j-1\}$, with $c > 0$ and $c' > 0$ or $c = 0$ and $c' > 0$ or $c > 0$ and $c' = 0$.

Proof

Let $L(1,t_{j-1}) > 0$ ....(A.2.34)

and $4\sum_{k=1}^{j-3}\gamma_k \le (j-4)\gamma_{j-2}$ ....(A.2.35)

Choose a vector of rates, r_{j-1}, as follows. Choose $n \in \{1, ..., j-1\}$ such that

$$3 \le n \le \frac{j+4}{2} \quad \text{....(A.2.36)}$$

and let
$$c = \frac{(j-1)\left((j-2n+4)\gamma_{n-2} + 4\sum_{k=1}^{n-3}\gamma_k\right)}{2\left((j-n)\gamma_{j-2}\gamma_{n-2} + 2\gamma_{j-2}\sum_{k=1}^{n-3}\gamma_k - 2\gamma_{n-2}\sum_{k=1}^{j-3}\gamma_k\right)} \quad \text{....(A.2.37)}$$

and
$$c' = \frac{(j-1)\left((j-4)\gamma_{j-2} - 4\sum_{k=1}^{j-3}\gamma_k\right)}{2\left((j-n)\gamma_{j-2}\gamma_{n-2} + 2\gamma_{j-2}\sum_{k=1}^{n-3}\gamma_k - 2\gamma_{n-2}\sum_{k=1}^{j-3}\gamma_k\right)} \quad \text{....(A.2.38)}$$

and

$$r_k = \begin{cases} \dfrac{j-1 + c\left((j-k-1)\tau_{j-1}-\gamma_{j-2}\right) + c'\left((n-k-1)\tau_{j-1}-\gamma_{n-2}\right)}{\tau_{j-1}} & k = 1 \ldots n-1 \\ \dfrac{j-1 + c\left((j-k-1)\tau_{j-1}-\gamma_{j-2}\right) - c'\gamma_{n-2}}{\tau_{j-1}} & k = n \ldots j-1 \end{cases} \quad \text{....(A.2.39)}$$

Then
$$r_k - r_{k-1} = \begin{cases} -c-c' & k = 2 \ldots n-1 \\ -c & k = n \ldots j-1 \end{cases} \quad \text{...(A.2.40)}$$

and
$$r_k - 2r_{k-1} - r_{k-2} = \begin{cases} c' & k = n \\ 0 & k = 3 \ldots j-1 \ k \ne n \end{cases} \quad \text{.....(A.2.41)}$$

Now $2\left((j\text{-}n)\gamma_{j-2}\gamma_{n-2} + 2\gamma_{j-2}\sum_{k=1}^{n-3}\gamma_k - 2\gamma_{n-2}\sum_{k=1}^{j-3}\gamma_k\right)$

$$= \gamma_{j-2}\left((j\text{-}2n+4)\gamma_{n-2} + 4\sum_{k=1}^{n-3}\gamma_k\right) + \gamma_{n-2}\left((j\text{-}4)\gamma_{j-2} - 4\sum_{k=1}^{j-3}\gamma_k\right)$$

and so from *(A.2.35)*, *(A.2.36)*, *(A.2.37)* and *(A.2.38)* if $4\sum_{k=1}^{j-3}\gamma_k < (j\text{-}4)\gamma_{j-2}$ then either $c > 0$ and $c' > 0$ (and they are both bounded) or $c = 0$ and $c' > 0$ (and they are both bounded). If $4\sum_{k=1}^{j-3}\gamma_k = (j\text{-}4)\gamma_{j-2}$ then choose $n \in \{3, ..., j\text{-}1\}$ so that $(j\text{-}2n+4)\gamma_{n-2} + 4\sum_{k=1}^{n-3}\gamma_k > 0$ and then $c > 0$ and $c' = 0$ (and they are both bounded). Further, from *(A.2.37)*, *(A.2.38)* and *(A.2.39)*,

$$r_{j-1} = \frac{j\text{-}1 - c\gamma_{j-2} - c'\gamma_{n-2}}{\tau_{j-1}} = 0 \qquad \ldots.(A.2.42)$$

So from *(A.2.37)*, *(A.2.38)*, *(A.2.40)*, *(A.2.41)* and *(A.2.42)* constraints *(A.2.14)*, *(A.2.15)*, *(A.2.16)* and *(A.2.17)* are satisfied and so r_{j-1}, as defined above, is a feasible solution.

From *(A.2.13)*, *(A.2.17)*, *(A.2.37)*, *(A.2.38)* and *(A.2.39)*

$$l(r_{j-1},t_{j-1}) = \left|\frac{j\text{-}1}{2} - \frac{c\left(\gamma_{j-2} + 2\sum_{k=1}^{j-3}\gamma_k\right)}{j\text{-}2} - \frac{c'\left((j\text{-}n+1)\gamma_{n-2} + 2\sum_{k=1}^{n-3}\gamma_k\right)}{j\text{-}2}\right| = 0$$

and so we have defined a feasible solution r_{j-1} with zero objective and so r_{j-1} is an optimum solution.

(Note that *(A.2.34)* is not required, but *(A.2.35)* implies *(A.2.34)*).

Corollary A.2.4

If $L(1,t_{j-1}) > 0$ then there are at least $n(j)+1$ optimum solutions with zero objective and at most one non-zero second difference where

$$n(j) = \begin{cases} \frac{j}{2} & \text{if } j \text{ is even} \\ \frac{j\text{-}1}{2} & \text{if } j \text{ is odd} \end{cases}$$

Proof

From the proof of Theorem A.2.1 there will always be at least $n(j)$ optimum solutions (by choosing $n = 3, 4, \ldots,$ up to $\frac{j+4}{2}$ if j is even, or up to $\frac{j+3}{2}$ if j is odd). From the proofs of theorem A.2.2, if $4\sum_{k=1}^{j-3}\gamma_k \geq (j-4)\gamma_{j-2}$ then there is an additional optimum solution. From the proof of theorem A.2.3, if $4\sum_{k=1}^{j-3}\gamma_k \leq (j-4)\gamma_{j-2}$ then there will are an additional $n(j)$ optimum solutions.

Theorem A.2.5

The *HPP* vector of rates, r^h_{j-1}, is always feasible.

Proof

From *(A.2.10)*, since $r_k = r^h_{j-1}$ for all $k = 1, \ldots, j-1$, trivially constraints *(A.2.15)* and *(A.2.16)* are satisfied and since $r^h_{j-1} = \frac{j-1}{\tau_{j-1}}$ constraints *(A.2.14)* and *(A.2.17)* are also satisfied. Thus, r^h_{j-1} is always a feasible solution given any vector of inter-failure times, t_{j-1}.

Theorem A.2.6

If $L(1,t_{j-1}) > 0$ then the *HPP* vector of rates, r^h_{j-1}, is feasible, but not optimal.

Proof

Let $L(1,t_{j-1}) > 0$ ....*(A.2.43)*

From Theorem A.2.5 r^h_{j-1}, is trivially feasible. Further from *(A.2.10)*, *(A.2.12)*, *(A.2.13)* and *(A.2.43)*

$$l(r^h_{j-1},t_{j-1}) = \left|\frac{(j-1)L(1,t_{j-1})}{\tau_{j-1}}\right| > 0 \quad \ldots.(A.2.44)$$

From *(A.2.43)* and Theorem A.2.1 there exists a feasible optimal solution, r_{j-1}, such that $L(r_{j-1},t_{j-1}) = 0$ and $\Delta r_k = -c < 0$, for all $k \in \{2, \ldots, n-1\}$ and $\Delta r_k = 0$ for all $k \in \{n, \ldots, j-1\}$ and so $r_{j-1} \neq r^h_{j-1}$ and from *(A.2.13)* and *(A.2.44)*

$$l(r^{h}_{j-1},t_{j-1}) > l(r_{j-1},t_{j-1})$$

and so r^{h}_{j-1} is feasible, but not optimal.

Theorem A.2.7

If $L(1,t_{j-1}) \leq 0$ then the *HPP* vector of rates, r^{h}_{j-1}, is feasible and optimal and additionally if $t_k \neq 0$ for all $k = 1, ..., j-1$, then r^{h}_{j-1} is uniquely optimal.

Proof

Let $L(1,t_{j-1}) \leq 0$ ....*(A.2.45)*

From Theorem A.2.5 r^{h}_{j-1} is trivially feasible. Further, from *(A.2.10)*, *(A.2.12)*, *(A.2.13)* and *(A.2.45)*

$$l(r^{h}_{j-1},t_{j-1}) = \left|\frac{(j-1)L(1,t_{j-1})}{\tau_{j-1}}\right| = -\left(\frac{(j-1)L(1,t_{j-1})}{\tau_{j-1}}\right) \quad ....(A.2.46)$$

Let $r_{j-1} = \langle r_1, ..., r_{j-1}\rangle$ be another feasible solution (distinct from r^{h}_{j-1}). Then, from *(A.2.15)*

$$r_k - r_{k-1} = -c_k \qquad c_k \geq 0,\ k = 2, ..., j-1 \qquad ....(A.2.47)$$

and from *(A.2.17)* since r_{j-1} is distinct from r^{h}_{j-1}, $c_k > 0$ for at least one $k \in \{2, ..., j-1\}$.

Then from *(A.2.17)* and *(A.2.47)*

$$r_k = \frac{\left((j-1) - \sum_{s=1}^{j-2}(c_{s+1}\tau_s) + \tau_{j-1}\sum_{s=k+1}^{j-1}c_s\right)}{\tau_{j-1}} \qquad k = 1, ..., j-1 \qquad ....(A.2.48)$$

From *(A.2.12)*, *(A.2.13)*, *(A.2.17)* and *(A.2.48)*

$$l(r_{j-1},t_{j-1}) = \left|\frac{(j-1)L(1,t_{j-1}) + \sum_{k=1}^{j-2}\left(\left(\sum_{s=1}^{j-2}(c_{s+1}\tau_s) - \tau_{j-1}\sum_{s=k+1}^{j-1}c_s\right)(j-k-1)t_k\right)}{\tau_{j-1}}\right|$$

$$= \left| \frac{(j-1)L(1,t_{j-1}) - \sum_{k=1}^{j-2}\left(\sum_{s=k+1}^{j-1}\left((s-k)\left(\sum_{p=k+1}^{j-1} c_p\right)t_s t_k\right)\right)}{\tau_{j-1}} \right|$$

$$= -\left(\frac{(j-1)L(1,t_{j-1}) - \sum_{k=1}^{j-2}\left(\sum_{s=k+1}^{j-1}\left((s-k)\left(\sum_{p=k+1}^{j-1} c_p\right)t_s t_k\right)\right)}{\tau_{j-1}} \right) \qquad \ldots.(A.2.49)$$

from *(A.2.45)* and *(A.2.47)*. So, from *(A.2.46)* and *(A.2.49)*

$$l(r_{j-1},t_{j-1}) - l(r^h_{j-1},t_{j-1}) = \frac{\sum_{k=1}^{j-2}\left(\sum_{s=k+1}^{j-1}\left((s-k)\left(\sum_{p=k+1}^{j-1} c_p\right)t_s t_k\right)\right)}{(j-2)\tau_{j-1}}$$

and so from *(A.1.47)* (since $c_k \geq 0$ for all $k = 2, \ldots, j-1$) $l(r_{j-1},t_{j-1}) \geq l(r^h_{j-1},t_{j-1})$ for all other feasible solutions r_{j-1} and so r^h_{j-1} is optimal. Further, provided $t_k \neq 0$ for all $k = 1, \ldots, j-1$, then (since $c_k \geq 0$ for all $k = 2, \ldots, j-1$ and $c_k > 0$ for at least one $k \in \{2, \ldots, j-1\}$) $l(r_{j-1},t_{j-1}) > l(r^h_{j-1},t_{j-1})$ for all other feasible solutions r_{j-1} and so r^h_{j-1} is uniquely optimal.

A.3 Extension to OTL Model

To summarise let

$$L(a_{m,j-1},t_{m,j-1}) = \frac{\sum_{k=m}^{j-1} a_k t_k}{2} - \frac{\sum_{k=m}^{j-2}\sum_{s=m}^{k} a_s t_s}{j-m-1} \qquad m = 1, \ldots, j-2 \qquad \ldots\ldots(A.3.1)$$

where $t_{m,j-1} = \langle t_m, \ldots, t_{j-1} \rangle$ and $a_{m,j-1} = \langle a_m, \ldots, a_{j-1} \rangle$. Then it is easy to prove that

$$L(a_{m,j-1},t_{m,j-1}) = \frac{\sum_{k=m}^{j-1} (2k-m-j+1) a_k t_k}{2(j-m-1)} \qquad m = 1, \ldots, j-2 \qquad \ldots\ldots(A.3.2)$$

Let

$$n = \begin{cases} j-1 & \text{if } L(1_{m,j-1},t_{m,j-1}) > 0 \text{ for all } m = 1 \ldots j-1 \\ Min[m \in \{1 \ldots j-2\};\ L(1_{m,j-1},t_{m,j-1}) \leq 0] & \text{otherwise} \end{cases} \qquad \ldots(A.3.3)$$

noting that, if we define $L(1_{j-1,j-1}, t_{j-1,j-1}) = \frac{t_{j-1}}{2}$, then this will always be positive for positive t_{j-1}. Then, let

$$L(r_{m,j-1}, t_{m,j-1}) = 0 \qquad \text{for all } m = 1, \ldots, n-1 \qquad \ldots\ldots(A.3.4)$$

and $r_k = r$, say, for all $k = n, \ldots, j-1$ $\ldots\ldots(A.3.5)$

The objective of this is to constrain the $\{r_k t_k;\ k = m, \ldots, j-1\}$ to be trend free over all consecutive vectors of the data, $t_{m,j-1}$, which exhibit growth ($m = 1, \ldots, n-1$), while letting the rates be equal over the largest vector of data, $t_{n,j-1}$, which does not exhibit growth.

Since, from *(A.3.4)* and *(A.3.5)* we have n variables and $n-1$ equations, in order to get a unique solution we must apply a scaling constraint, so let

$$\sum_{k=1}^{j-1} d_k r_k t_k = \sum_{k=1}^{j-1} d_k \qquad \ldots\ldots(A.3.6)$$

From *(A.3.2)*, *(A.3.4)*, *(A.3.5)* and *(A.3.6)* the problem can then be formulated in matrices as follows

$$\begin{pmatrix} d' \\ A_1 \end{pmatrix} r = \begin{pmatrix} d \\ 0 \end{pmatrix}$$

where $d' = \left(d_1 t_1\ d_2 t_2\ d_3 t_3 \cdots d_{n-1} t_{n-1}\ \sum_{k=n}^{j-1} d_k t_k \right)$ $r' = \left(r_1\ r_2\ r_3 \cdots r_{n-1}\ r \right)$, $d = \sum_{k=1}^{j-1} d_k$ and

$$A_1 = \begin{pmatrix} (2-j)t_1 & (4-j)t_2 & \cdots & (2n-2-j)t_{n-1} & \sum_{k=n}^{j-1}(2k-j)t_k \\ 0 & (3-j)t_2 & \cdots & (2n-3-j)t_{n-1} & \sum_{k=n}^{j-1}(2k-j-1)t_k \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & (n-j+2)t_{n-1} & \sum_{k=n}^{j-1}(2k-n-j+4)t_k \\ 0 & 0 & \cdots & (n-j+1)t_{n-1} & \sum_{k=n}^{j-1}(2k-n-j+3)t_k \\ 0 & 0 & \cdots & (n-j)t_{n-1} & \sum_{k=n}^{j-1}(2k-n-j+2)t_k \end{pmatrix}$$

By simple row manipulations A_1 can be reduced to the following

$$A_2 = \begin{pmatrix} A & 0 & 0 \\ 0' & (n-j-1)t_{n-2} & t_{n-1} \quad \sum_{k=n}^{j-1} t_k \\ 0' & 0 & (n-j)t_{n-1} \quad \sum_{k=n}^{j-1}(2k-n-j+2)t_k \end{pmatrix}$$

where A is the following $(n-3)$ by $(n-2)$ matrix

$$A = \begin{pmatrix} t_1 & -t_2 & 0 & \cdots & 0 & 0 \\ 0 & t_2 & -t_3 & \cdots & 0 & 0 \\ 0 & 0 & t_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -t_{n-3} & 0 \\ 0 & 0 & 0 & \cdots & t_{n-3} & -t_{n-2} \end{pmatrix}$$

and we have that

$$Mr = \begin{pmatrix} d \\ 0 \end{pmatrix} \qquad \ldots\ldots(A.3.7)$$

where

$$M = \begin{pmatrix} d' \\ A_2 \end{pmatrix} = \begin{pmatrix} & & d' & \\ & A & 0 & 0 \\ 0' & (n-j-1)t_{n-2} & t_{n-1} & \sum_{k=n}^{j-1} t_k \\ 0' & 0 & (n-j)t_{n-1} & \sum_{k=n}^{j-1} (2k-n-j+2)t_k \end{pmatrix}$$

It can easily be proved that

$$det(M) = \left| \left(\prod_{r=1}^{n-1} t_r \right) \sum_{k=n}^{j-1} \left(\left(2(k-n+1) \sum_{l=1}^{n-2} d_l + (j-n+1)((2k-n-j+2)d_{n-1} - (n-j)d_k) \right) t_k \right) \right| \quad \ldots(A.3.8)$$

It can be derived from *(A.3.7)* (or checked by substitution into the initial equations) that a solution to the problem is

$$r_m = \frac{1}{t_m} r_1 t_1 \qquad \text{for } m = 1,\ldots,n-2$$

$$r_{n-1} = \frac{(j-n+1)\left(\sum_{k=n}^{j-1} (2k-n-j+2)t_k \right)}{2t_{n-1}\left(\sum_{k=n}^{j-1} (k-n+1)t_k \right)} r_1 t_1 \qquad \ldots\ldots(A.3.9)$$

$$r = \frac{(j-n)(j-n+1)}{2\left(\sum_{k=n}^{j-1} (k-n+1)t_k \right)} r_1 t_1$$

for $n = 2,\ldots,j-1$. For prediction purposes we will be using the rate, r, in *(A.3.9)*. It can be seen that this is made up of the reciprocal of a weighted average of the largest vector (moving from the last inter-failure time backwards) of inter-failure times that exhibits no growth, with more weight given to most recent inter-failure times.

Note that for $n = j\text{-}1$ *(A.3.9)* simplifies to

$$r_m = \frac{1}{t_m} r_1 t_1 \qquad \text{for } m = 1,...,j\text{-}1 \qquad \ldots.(A.3.10)$$

In the special case of $n = 1$ the assumption that $r_1 = r_2 = ... = r_{j-1} = r$, say, gives

$$r_m = \frac{1}{t_1} r_1 t_1 \qquad \text{for } m = 1,...,j\text{-}1 \qquad \ldots.(A.3.11)$$

It can be seen from *(A.3.9)*, *(A.3.10)* and *(A.3.11)* that the "shape" of the rates (i.e., their relative ratio) is dictated entirely by the constraints of equal rates over the largest vector exhibiting no growth together with setting the appropriate Laplace statistics to zero (if applicable). In other words the impact of constraint *(A.3.6)* is truly only for scaling purposes.

If we apply the same scaling constraint as in the *OTY* model (from *(5.2.3)*, that is $d_k = 1$ for all $k = 1,...,j\text{-}1$) we obtain, from *(A.3.6)*, *(A.3.9)*, *(A.3.10)* and *(A.3.11)*, the following solution,

for $n = 2,...,j\text{-}1$,

$$r_m = \frac{1}{t_m} \qquad \text{for } m = 1,...,n\text{-}2$$

$$r_{n-1} = \frac{(j\text{-}n+1)\left(\sum_{k=n}^{j-1}(2k\text{-}n\text{-}j+2)t_k\right)}{2t_{n-1}\left(\sum_{k=n}^{j-1}(k\text{-}n+1)t_k\right)} \qquad \ldots.(A.3.12)$$

$$r = \frac{(j\text{-}n)(j\text{-}n+1)}{2\left(\sum_{k=n}^{j-1}(k\text{-}n+1)t_k\right)}$$

and for $n = j\text{-}1$ *(A.3.12)* simplifies to

$$r_m = \frac{1}{t_m} \qquad \text{for } m = 1,...,j\text{-}1 \qquad \ldots.(A.3.13)$$

and for $n = 1$, from *(A.3.6)*, we have the *HPP* solution,

$$r_m = r^h = \frac{j\text{-}1}{\sum_{k=1}^{j-1} t_k} \qquad \text{for } m = 1,...,j\text{-}1 \qquad \ldots.(A.3.14)$$

Additionally, from *(A.3.8)* we have

$$det(M) = 2(j\text{-}1)\left(\prod_{r=1}^{n-1} t_r\right)\left(\sum_{k=1}^{j-n} k t_{k+n-1}\right) \qquad \ldots.(A.3.15)$$

which is non-zero providing we exclude the possibility of zero inter-failure times. This implies that the solution given in *(A.3.12)* to *(A.3.14)* is unique.

Comparison of *(A.3.12)* and *(A.3.14)* shows that, while $n = 2,...,j-1$ gives the equal rate solution to be a weighted average of the largest vector which exhibits no growth, when $n = 1$ we get a non-weighted average. Since choice of the d_k is somewhat arbitrary it might be better to seek d_k for which the solution for $n = 1$ has similar weighting as the solution for $n = 2,...,j-1$ (specifically if we substitute $n = 1$ in *(A.3.12)* we wish to obtain solution for $n = 1$). Since the solution for $n = 1$ is given by *(A.3.6)* it is easy to see that if we choose $d_k = (k-n+1)$ we will get the desired solution. In fact this gives the solution identical to when $d_k = 1$ except that for $n = 1$,

$$r_m = \frac{j(j-1)}{\sum_{k=1}^{j-1} 2kt_k} \qquad \text{for } m = 1,...,j-1 \qquad \ldots.(A.3.16)$$

Additionally, from *(A.3.8)*, we now have

$$det(M) = \left| (j-1)(j-2n+2)\left(\prod_{r=1}^{n-1} t_r\right)\left(\sum_{k=1}^{j-n} kt_{k+n-1}\right)\right| \qquad \ldots.(A.3.17)$$

which shows that with this new choice of d_k, we are not guaranteed a unique solution when $n = \frac{j+2}{2}$. For this reason, the model is applied with $d_k = 1$, for $k = 1,...,j-1$, since we are then guaranteed a unique solution provided the inter-failure times are non-zero.