

基于NHPP模型组合的失效时间预测

涂阳泽

北京航空航天大学, 北京
Email: yangzetu0319@163.com

收稿日期: 2020年10月5日; 录用日期: 2020年10月20日; 发布日期: 2020年10月27日

摘要

为了提高软件可靠性模型的预测精度, 本文提出了一种基于NHPP可靠性模型的组合预测模型。该模型以几个经典的NHPP模型为基预测模型, 通过贝叶斯方法推导出模型在某一次失效中占的权重值与其在上一次失效中各模型占的权重值的关系式, 再运用最大熵原理获得预测误差的分布, 迭代得出各个预测模型的权重。考虑到由时效性引起的很久前的信息导致了对当前信息判断的误差, 所以将迭代次数设为固定常数而不再迭代到初始时间, 提高了模型对软件可靠性的预测精度。实验的结果也表明, 本文所研究的模型在预测的精度上好于单一模型, 对比传统的组合模型以及不考虑时效性的模型也都有更好的表现。

关键词

软件可靠性, NHPP组合模型, 贝叶斯公式, 最大熵原理

Failure Time Prediction Based on Combination of NHPP-Models

Yangze Tu

Beihang University, Beijing
Email: yangzetu0319@163.com

Received: Oct. 5th, 2020; accepted: Oct. 20th, 2020; published: Oct. 27th, 2020

Abstract

In order to improve the prediction accuracy of software reliability models, this paper proposed a combination prediction model based on NHPP software reliability models, first using the Bayesian Approach to deduce the relation between the weight of every single model in one software failure

and the previous failure; second, obtaining the distribution of prediction error by Maximum Entropy Principle, then getting the weight of every single model by iteration. Considering that the failure occurred long time before should not affect the information we obtained in recent times, we set the iteration times as a constant, which improved the prediction accuracy. The results of experiment show that the combined model this paper proposed performs better than single model and other traditional combined models on prediction accuracy.

Keywords

Software Reliability, NHPP Combined Model, Bayesian Approach, Maximum Entropy Method

Copyright © 2020 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

近年来,随着计算机技术的日渐发展,计算机软件在我们的生活中扮演了一个重要的角色。由于人们的生活水平不断在提高,软件的需求程度也更高,因此其复杂度以及规模都有了很大的增加,随之而来的问题就是由于软件失效所造成的各种损失也开始变得严重起来。于是,作为一个评判软件质量的指标,软件可靠性受到了学者们的广泛关注[1]。

软件自身是复杂的,会受到很多方面的影响,涉及到其开发过程、操作过程等很多因素,因此存在很多的不确定性。而通过已有的失效数据进行可靠性模型的构建,模拟软件失效的过程,是一种比较常见的方式。软件可靠性模型是对可靠性进行分析的经典方法。早在1964年, J. T. Duane 就提出了 Duane 模型[2],该模型认为单位时间内的失效次数是一个非齐次泊松过程。而在1972年, Jelinsk、Morand 提出了一个软件可靠性模型 J-M 模型[3],该模型做了一个最基本的假设也就是软件失效数是一个固定的常数且不会引入新的软件失效。这个模型直接推动了可靠性模型的发展,紧随其后涌现出了许多的软件可靠性模型,其中,以 NHPP (Non-Homogeneous Poisson Process)模型为代表。而 NHPP 模型又有很多,在1975年由 Schneidewind 提出 Schneidewind 模型[4],该模型认为不同的失效时间对应不同的分布函数,并提出了在软件失效的过程里,失效时间间隔可以看作一个随机过程,这也就是 NHPP 模型类模型的建模基本思路。1975年, J. D. Musa 又提出了 M-O 模型[5],并认为测试时间中的唯一有效的测度是执行时间。1979年 Goel 等人又提出了 G-O 模型[6],假设能够在发现后立即处理软件失效,而不会因此引入新的失效。

随着科技的发展,越来越多的可靠性模型出现。然而到目前为止,依旧没有一种适应所有环境能够应对所有情况的最佳模型。即便某些模型在特定的数据集上面有很好的预测效果,但是随着失效数的增多,也无法保证该模型依旧能有很好的表现。所以,模型的组合成为了一个值得研究的方向。ELC 是最早被提出的一种模型组合方式,该模型简单地将基模型进行等权线性组合。但是这种方法考虑的太简单,由于主观上对权重进行均等划分,而并没有考虑任何失效时间所带的信息,所以使用起来的效果并不会很好。在这种模型线性组合思想的基础上,也出现了更多不同的加权方法[7] [8] [9] [10]。本文通过对模型选择的不确定性进行考虑,并通过贝叶斯公式将其转换成一个递归问题从而求得各个模型的权重值。然后,在此基础上进行改进,将迭代次数设为一个常数而不是迭代到初始值,得到的权重值是只与最近几次的失效相关的,从而不会被久远以前的信息所影响,因此提高了模型的信息更新速度,并取得了不错的效果。

2. NHPP 基本模型

2.1. 模型框架

不同的软件可靠性模型都有不同的形式以及不同的表达式，但是大多数的软件可靠性模型都是在一个框架之下建立的。对于不同类别的可靠性模型，根据他们不同的模型假设，推导出来的模型框架也是不同的。NHPP 模型就是一类模型。他们的通用假设是一致的，对于不同的 NHPP 模型他们的区别在于一些特定假设的不同，这也决定了他们的结构特征的不同。NHPP 模型的基本假设如下：

- 1) 在零时刻的软件失效数 $N(0)=0$ ，也就是 $P(N(0)=0)=1$ 。
- 2) $N(t)$ 是独立增量， $N(t+\Delta t)-N(t)$ 与先前的失效无关。
- 3) 在时间 $(t, t+\Delta t]$ 一次失效发生的概率为 $\lambda(t)\Delta t+o(\Delta t)$ ，其中 $\lambda(t)$ 表示失效强度。
- 4) 在时间 $(t, t+\Delta t]$ 产生的失效次数大于 1 的概率为 $o(\Delta t)$ 。

通过以上基本假设，可以推出软件失效数 $N(t)$ 是一个均值为 $m(t)$ 的泊松过程，表达式为：

$$P_n(t) = \frac{(m(t))^n}{n!} e^{-m(t)}$$

其中， $m(t)$ 是失效均值，与失效强度 $\lambda(t)$ 的关系是： $m(t) = \int_0^t \lambda(s) ds$ 。这就是 NHPP 模型的基本模型框架。对于不同的 NHPP 模型，他们的区别在于失效均值 $m(t)$ 的不同。表 1 给出了几个经典的 NHPP 模型的模型特征。

Table 1. Features of NHPP models

表 1. NHPP 模型特征

模型	失效均值	失效强度
G-O 模型	$m(t) = a(1 - e^{-bt})$	$\lambda(t) = abe^{-bt}$
M-O 模型	$m(t) = \frac{1}{\theta} \ln(\lambda_0 \theta t + 1)$	$\lambda(t) = \frac{\lambda_0}{\lambda_0 \theta t + 1}$
Duane 模型	$m(t) = \lambda t^b$	$\lambda(t) = \lambda b t^{b-1}$

2.2. 参数估计

对于经典的软件可靠性模型的参数估计，估计方法有很多。在这里，对经典 NHPP 模型我们统一采用极大似然估计法来估计模型的参数。极大似然估计对于数据的要求分为失效时间间隔数据，以及失效计数数据。本文所使用的数据都是失效时间间隔数据，因此本文只给出数据类型为失效时间间隔数据的极大似然估计。

在 NHPP 模型基本假设下，似然函数为：

$$L(\theta) = \left(\prod_{i=1}^N \lambda(t_i) \right) e^{-m(t_N)}$$

再求对数似然函数：

$$\ln L(\theta) = \sum_{i=1}^N \ln \lambda(t_i) - m(t_N)$$

其中， θ 为待估计的参数向量， N 为实际观测到的总失效数， t_i 为第 i 次失效发生的时刻，并且 $t_i = \sum_{k=1}^i s_k$ ， s_k 是由数据得到的失效时间间隔。由于单个 NHPP 模型都只有两个待估参数，使用极大似然估计比较容易得出合适的估计结果。但是在我们进行模型组合时，每增加一个基模型，所需要估计的参数数量就增加了三个。而随着参数数量的增加，极大似然估计法的参数求解也就越来越困难。因此，对组合模型直接

使用极大似然估计很难得出理想的效果，这也就是接下来我们要讨论的问题。

3. 模型组合

3.1. 模型组合方法

本文给出接下来几次的失效时间预测的组合模型，以下简称为 COM 模型。

将我们观测到的 N 次失效时间数据的失效时间间隔记为 s_1, s_2, \dots, s_N 。假设我们一共有 M 个基模型，对第 i 个基模型，将对其第 m 次失效时间的预测值记为 $f_i(m)$ ，那么第 i 个模型对第 m 次失效时间的预测误差是：

$$e_i(m) = f_i(m) - s_m$$

假设在我们的基模型中，预测效果最好的模型是第 k 个模型，那么，我们引入随机变量 K 来表示对第 m 次失效时间的预测模型的不确定性，定义如下：

$$p_k(m) = P(K = k | s_1, s_2, \dots, s_m)$$

由于 k 的取值为从 1 到 M 的正整数，所以根据贝叶斯公式，有：

$$p_k(m) = \frac{P(K = k | s_1, s_2, \dots, s_{m-1})P(s_m | K = k, s_1, s_2, \dots, s_{m-1})}{\sum_{j=1}^M P(K = j | s_1, s_2, \dots, s_{m-1})P(s_m | K = j, s_1, s_2, \dots, s_{m-1})}$$

不难观察得到：

$$\begin{aligned} p_k(m-1) &= P(K = k | s_1, s_2, \dots, s_{m-1}) \\ P(s_m | K = k, s_1, s_2, \dots, s_{m-1}) \\ &= P(s_m = f_i(m) - e_i(m) | K = k, s_1, s_2, \dots, s_{m-1}) \\ &= P(e_k(m) = f_k(m) - s_m | s_1, s_2, \dots, s_{m-1}) \end{aligned}$$

第 k 个预测模型的预测误差也是一个随机变量 E_k ，也就是要求

$$P(E_k(m) = f_k(m) - s_m)$$

如果我们能够获得随机变量 E_k 的分布，就能通过递归求出 $p_k(m)$ 的值了。对于随机变量 E_k ，我们通过基本模型的拟合，获得了 N 个样本，可以通过计算得到样本的期望 μ 以及方差 σ^2 。由于对总体我们没有已知的判断，所以这里我们使用最大熵原理(Maximum-Entropy Method)，以求得的样本的期望和方差作为先验信息，获得随机变量 E_k 的分布。

对于连续型的随机变量，熵的计算表达式为：

$$H = -\int p(x) \ln p(x) dx$$

通过最大熵原理，我们获得一个等式约束的最优化问题：

$$\begin{aligned} \text{Max: } H &= -\int p(x) \ln p(x) dx \\ \text{s.t.: } &\int p(x) dx = 1 \\ &\int p(x) f_k(x) dx = K \end{aligned}$$

这个问题的解为：
$$p(x) = \frac{1}{Z(\lambda_1, \lambda_2, \dots, \lambda_m)} \exp(\lambda_1 f_1(x) + \dots + \lambda_m f_m(x))$$
，其中，

$Z(\lambda_1, \lambda_2, \dots, \lambda_m) = \int \exp(\lambda_1 f_1(x) + \dots + \lambda_m f_m(x)) dx$ 是归一化因子, λ_i 是拉格朗日乘子, 通过:

$$K = \frac{\partial}{\partial \lambda_k} Z(\lambda_1, \lambda_2, \dots, \lambda_m)$$

计算得到。事实上, 通过给定期望 μ 以及方差 σ^2 , 由上面介绍的最大熵原理得到的密度函数就是正态分布 $N(\mu, \sigma^2)$ 的密度函数。于是, 对于第 k 个基模型, 其预测误差服从正态分布 $N(\mu_k, \sigma_k^2)$, 其中 μ_k, σ_k^2 分别表示第 k 个基模型的误差样本的期望以及方差。于是, 得到递推表达式:

$$p_k(m) = \frac{p_k(m-1)(2\pi\sigma_k^2)^{-\frac{1}{2}} \exp\left(-\frac{(e_k(m) - \mu_k)^2}{2\sigma_k^2}\right)}{\sum_{j=1}^M p_k(j-1)(2\pi\sigma_k^2)^{-\frac{1}{2}} \exp\left(-\frac{(e_k(j) - \mu_k)^2}{2\sigma_k^2}\right)}$$

于是我们可以计算出第 k 个模型在第 m 次预测的效果最好的概率, 这个值也就是我们组合模型中第 m 次预测第 k 个模型所占的权重, 也就是:

$$\hat{s}(m+1) = \sum_{i=1}^M p_i(m) f_i(m)$$

算法设计如下:

- 1) 通过观测到的失效时间数据求出各个基模型的参数
- 2) 令 $p_i(0) = 1/M, i = 1, 2, \dots, M$
- 3) 计算各个基模型的预测结果及误差, 将其迭代至 N , 得出各个模型在第 N 次预测中所占的权重, 并计算出组合模型的预测值。

3.2. 组合模型的改进

通过贝叶斯公式以及最大熵原理, 我们得出了模型权重值的迭代式。但是, 从表达式中可以看出, 无论时间过去多久, 过去的信息都会对最新的预测结果产生影响。然而软件可靠性是逐渐变化的。目前的软件可靠性以及未来短期内的软件可靠性, 更应该与最近几次的软件失效相关, 而不应该被过去很久的失效信息大幅影响。

每个预测模型都有其独特的信息, 可能在不同的时期会各有优劣。有些模型在数据波动较小的时候预测效果较好, 而有些则在数据波动较大时有更好的预测效果。若是每一次预测都要与过去很久的预测信息有比较大的关系, 很有可能会大幅度减少我们所获得的最近的失效所含有的信息。如果某个模型在前期的拟合效果不理想, 而在最近几次的预测精度都很高, 我们有理由相信这个模型在下几次的预测准确度也会较高。但是如果前期的误差导致权重值 $p_k(m-1)$ 变得非常小, 那么通过迭代, 即使后期概率 $P(E_k(m) = f_k(m) - s_m)$ 很大, 得到的权重值也会很小, 也会需要很久的时间更新信息。通过以上的分析, 本文提出一种基于时间步长的模型组合方法。

我们假设下一次失效的发生只与最近的 T 次失效有关系。对于大于 T 次的失效时间预测, 我们将从第 T 次失效前开始进行迭代而不是从头开始。也就是说, 在对第 N 次失效时间进行预测时, 我们只迭代到第 $N-T$ 次。此时, 我们直接舍弃掉第 $N-T$ 次失效之前的信息, 令 $p_i(N-T) = 1/M, i = 1, 2, \dots, M$, 这样的方法能更好的掌控模型最近时间段的趋势, 也更合理一些。改进后的算法设计如下:

- 1) 通过观测到的失效时间数据求出各个基模型的参数。
- 2) 若 $N \leq T$, 令 $p_i(0) = 1/M, i = 1, 2, \dots, M$, 计算各个基模型的预测结果及误差, 将其迭代至 N , 得

出各个模型在第 N 次预测中所占的权重，并计算出组合模型的预测值。

3) 若 $N > T$ ，令 $p_i(N-T) = 1/M, i = 1, 2, \dots, M$ ，将其迭代至 N ，得出各个模型在第 N 次预测中所占的权重，并计算出组合模型的预测值。

由该算法计算出的预测值，消除了在最近 T 次失效的失效时间之前的预测结果对现在的预测结果所产生的影响，更加符合真实情况，获得的权重值更加合理。我们将改进后基于时间步长的组合模型记为 COM-T。

4. 实例分析

为了验证我们提出的组合模型 COM-T 的有效性，我们将改进后的与时间步长有关的模型 COM-T 跟考虑所有时间的信息的模型 COM 进行分析比较，这里分别取步长 $T = 3, 4, 5$ 进行讨论。本文这里使用的基模型有三个，分别是 G-O 模型、M-O 模型和 Duane 模型。将本文所提出的两个组合模型与基模型，以及等权重线性组合模型 ELC 模型进行比较，其中 $ELC = \sum_{i=1}^M f_i / M$ ， f_i 是各失效时间预测函数。两个由前 t 次失效时间数据求出的组合模型 COM-T、COM 对第 $t+1$ 次失效的预测是：

$$\hat{s}(t+1) = \sum_{i=1}^M p_i(t) f_i(t)$$

其中 $M = 3$ 表示模型个数， $p_i(t)$ 表示在对第 $t+1$ 次失效预测时第 i 个模型所占的权重值。

4.1. 评判指标

对于模型的好坏的评估，我们需要一些指标来进行比较评判。本文使用软件可靠性中常用的几个指标如下：

相对误差 RE(relative error):

$$RE = \frac{1}{n} \sum_{i=1}^n \left| \frac{\hat{y}_i - y_i}{y_i} \right|$$

均方误差 MSE(mean square error):

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2$$

相对均方误差 RMSE(relative mean square error):

$$RMSE = \frac{1}{n} \sqrt{\sum_{i=1}^n \left(\frac{\hat{y}_i - y_i}{y_i} \right)^2}$$

其中， y_i 表示数据的真实值，而 \hat{y}_i 表示数据的预测值。这三个值越小，就说明预测值与真实值的差越小，也就是模型的预测效果更好。

4.2. 模型比较

本文使用来自文献[11]和文献[12]的两个失效时间间隔数据，对模型短时间的预测效果进行分析评判。其中数据集 1 有 61 个失效时间间隔数据，而数据集 2 有 101 个失效时间间隔数据。我们将最后五个数据作为测试集，将之前的所有数据作为训练集，分别对两组数据进行分析。

4.2.1. 数据集 1

对给定的失效时间间隔数据，先使用极大似然估计法求出各个基模型的参数值，如表 2 所示，再给出基模型以及 ELC 的失效数 - 失效时间在训练集上的拟合曲线，如图 1 所示。

Table 2. Parameters of basic models in dataset 1
表 2. 数据集 1 的基模型参数

模型	参数 1	参数 2
G-O 模型	123.3	0.0001364
M-O 模型	0.01770	0.01156
Duane 模型	0.05239	0.8300

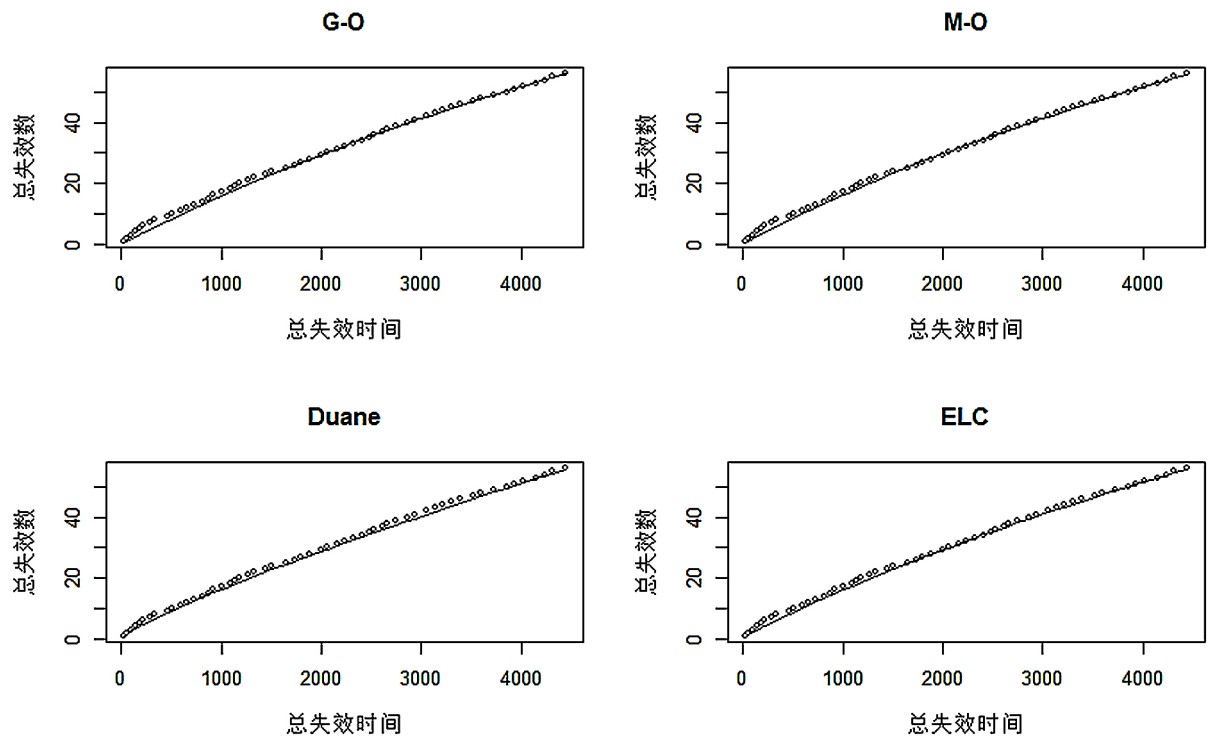


Figure 1. Fitting curve of failure numbers of basic models in dataset 1
图 1. 数据集 1 基模型失效数拟合曲线

数据集 1 所得出的 COM-T 模型对于不同的 $T = 3, 4, 5$ 以及 COM 模型接下来五次失效时间预测时的各个基模型的权重值如表 3 所示。

Table 3. The weights of basic models in five predictions in dataset 1
表 3. 数据集 1 的基模型在五次预测中的权重值

第一次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.198	0.276	0.526
COM-T = 3	0.312	0.313	0.375
COM-T = 4	0.263	0.263	0.474
COM-T = 5	0.330	0.328	0.342
第二次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.181	0.261	0.558
COM-T = 3	0.330	0.338	0.332
COM-T = 4	0.291	0.303	0.406
COM-T = 5	0.242	0.250	0.508

Continued

第三次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.174	0.260	0.566
COM-T = 3	0.318	0.325	0.357
COM-T = 4	0.321	0.340	0.339
COM-T = 5	0.282	0.304	0.414
第四次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.108	0.182	0.710
COM-T = 3	0.213	0.258	0.529
COM-T = 4	0.227	0.260	0.513
COM-T = 5	0.231	0.276	0.493
第五次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.187	0.290	0.523
COM-T = 3	0.336	0.361	0.303
COM-T = 4	0.316	0.351	0.333
COM-T = 5	0.331	0.350	0.319

再通过求得的权重得出各模型的预测值，并计算出各个评判指标。结果如图 2 以及表 4 所示。

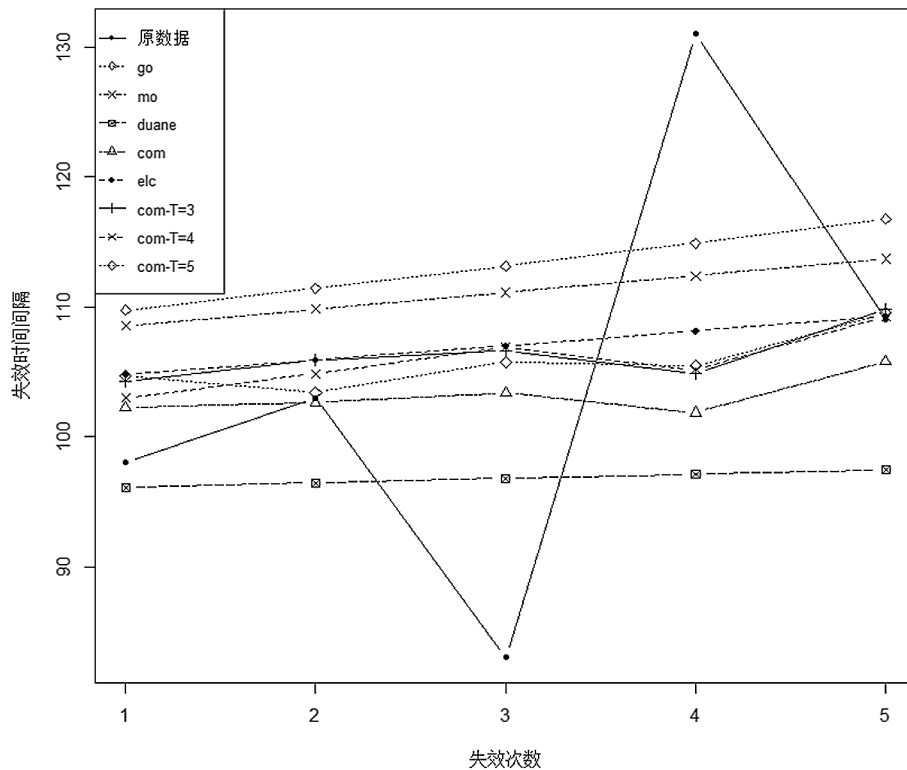


Figure 2. Predictions of failure times of basic models in dataset 1

图 2. 数据集 1 中的各模型的失效时间预测结果

Table 4. Judge index of each model in dataset 1
表 4. 数据集 1 中的各模型的评价指标

评判指标	RE	MSE	RMSE
G-O	0.152	287	0.0832
M-O	0.140	263	0.0781
Duane	0.123	303	0.0753
COM	0.113	259	0.0698
COM-T = 3	0.117	258	0.0710
COM-T = 4	0.111	253	0.0706
COM-T = 5	0.109	243	0.0687
ELC	0.113	264	0.0723

事实上, 结合预测结果与判别指标, 可以发现, 对于本文所使用的的数据集 1, 组合模型的各个指标都是优于基模型的。在 $T = 3$ 时, 基于步长的组合模型在 $T = 3$ 时的相对误差 RE 以及相对均方误差 RMSE 略高于原 COM 模型, 但是在 $T = 5$ 时却在三个指标上都优于 COM 模型。这说明选择的步长 T 的取值不同, 其与 COM 模型的优劣也会不同。当 $T = 3$ 时我们忽略了太多的信息以致于预测的效果不理想, 所以在利用更多的信息时, 预测的效果会有显著提升。对于 ELC 模型, 明显能看出其相对误差 RE 相较于 COM 模型以及 COM-T = 3 模型表现较好, 这是由其取值平均的特性导致的。而另外的两个指标上其表现就不如其他的组合模型了。

4.2.2. 数据集 2

与数据集 1 的方法相同, 我们先求出基模型的参数如表 5 所示, 再给出失效时间的拟合曲线如图 3 所示。

Table 5. Parameters of basic models in dataset 2
表 5. 数据集 2 的基模型参数

模型	参数 1	参数 2
G-O 模型	323.1	0.000362
M-O 模型	0.1177	0.003594
Duane 模型	0.1523	0.9368

我们通过观察拟合曲线, 不难发现 Duane 模型在失效时间 400~800 时的拟合偏差较大, 但是在接近我们测试集的时间终点时, 它又有较高的拟合精度。从直觉上, 如果我们使用 COM 组合模型, 考虑所有过去的时间信息, 很有可能会被这段时间的信息所影响从而较大的改变该模型所占的权重。表 6 也说明了这一点。

Table 6. The weights of basic models in five predictions in dataset 2
表 6. 数据集 2 的基模型在五次预测中的权重值

第一次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.623	0.377	2.15e-05
COM-T = 3	0.155	0.204	0.641
COM-T = 4	0.171	0.226	0.603
COM-T = 5	0.196	0.257	0.547
第二次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.586	0.414	4.54e-05
COM-T = 3	0.230	0.287	0.483
COM-T = 4	0.084	0.130	0.786
COM-T = 5	0.096	0.147	0.757

Continued

第三次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.575	0.425	4.28e-05
COM-T = 3	0.220	0.281	0.499
COM-T = 4	0.231	0.302	0.467
COM-T = 5	0.086	0.140	0.774
第四次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.550	0.450	5.76e-05
COM-T = 3	0.186	0.251	0.563
COM-T = 4	0.178	0.253	0.569
COM-T = 5	0.189	0.273	0.538
第五次失效预测	G-O 模型	M-O 模型	Duane 模型
COM	0.587	0.413	1.89e-05
COM-T = 3	0.414	0.142	0.174
COM-T = 4	0.323	0.375	0.302
COM-T = 5	0.312	0.380	0.307

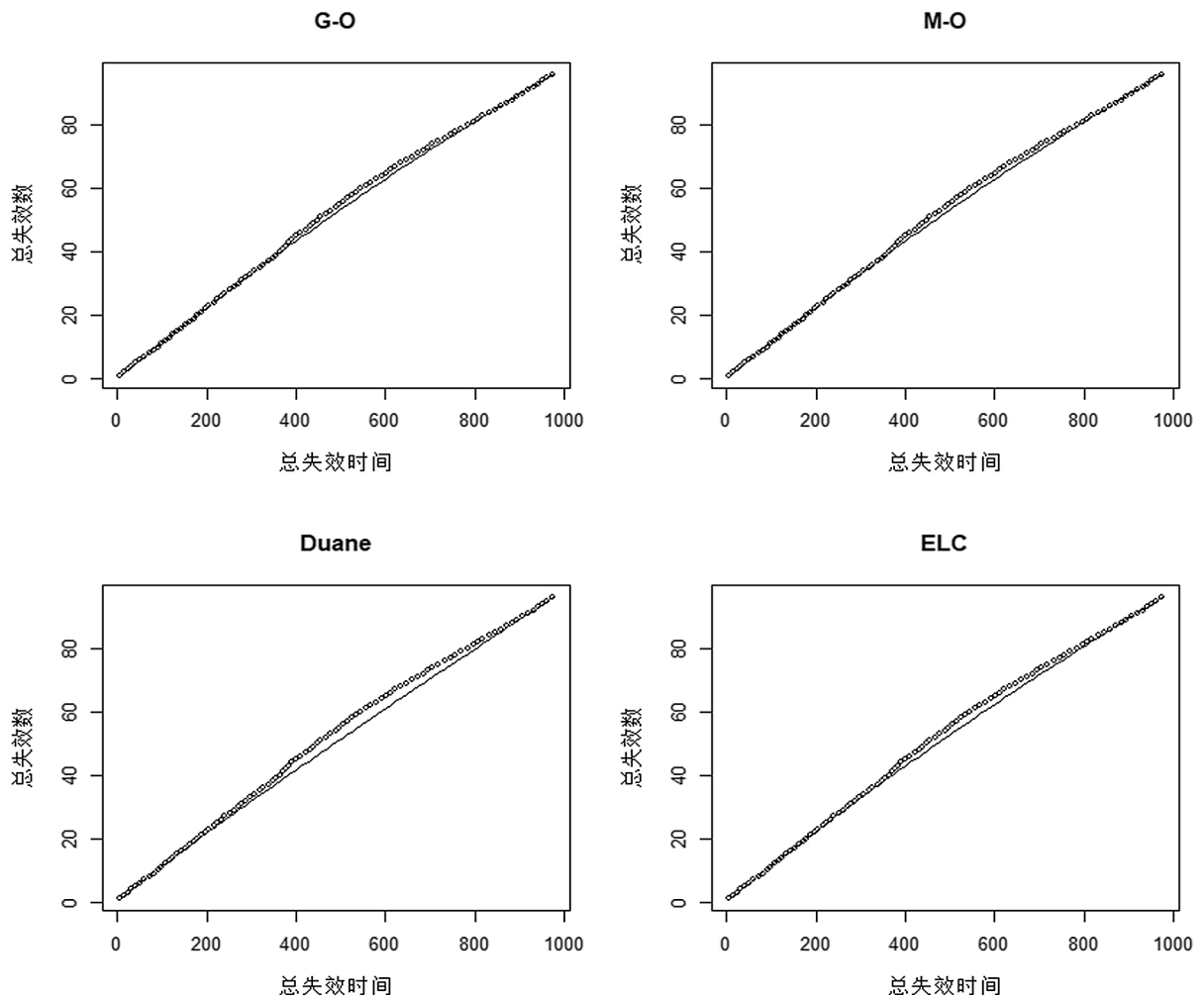


Figure 3. Fitting curve of failure numbers of basic models in dataset 2

图 3. 数据集 2 基模型失效数拟合曲线

可以看出 COM 模型在预测过程中，基模型 Duane 模型的权重值基本上为 0，也就是直接忽略掉了该模型所蕴含的信息。那么，我们通过预测效果图 4 以及评判指标表 7 来判断一下 COM 模型的预测效果：

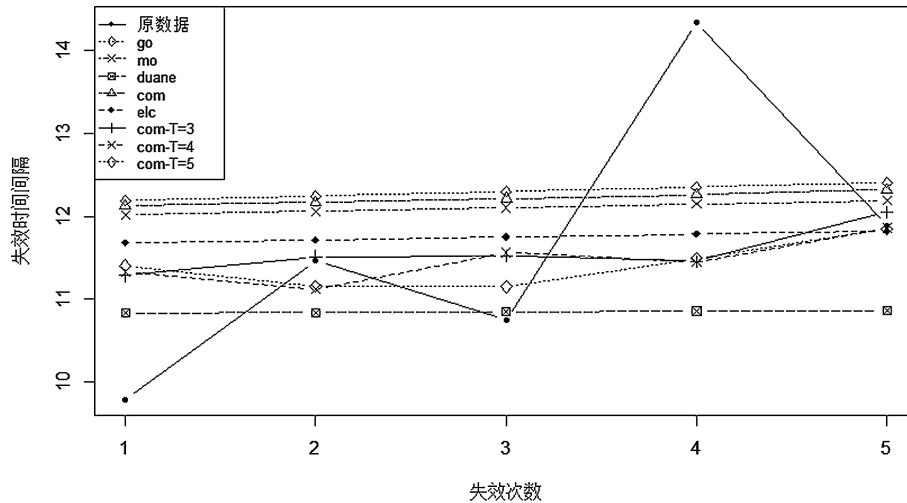


Figure 4. Predictions of failure times of basic models in dataset 2
图 4. 数据集 2 中的各模型的失效时间预测结果

Table 7. Judge index of each model in dataset 2

表 7. 数据集 2 中的各模型的评价指标

评判指标	RE	MSE	RMSE
G-O	0.128	2.62	0.0656
M-O	0.117	2.43	0.0617
Duane	0.100	2.96	0.0571
COM	0.124	2.54	0.0640
COM-T = 3	0.0890	2.25	0.0528
COM-T = 4	0.0938	2.32	0.0539
COM-T = 5	0.0865	2.21	0.0526
ELC	0.113	2.35	0.0562

显然，在忽略掉了 Duane 模型之后，COM 模型的实质上变成了 G-O 模型以及 M-O 模型的组合。但是实际上在预测效果上 Duane 模型也有其自身的优势，COM 模型由于其权重受长时间以前的影响较大，更新缓慢，导致了预测效果并不佳。对比考虑时间步长的 COM-T 模型，在 $T = 3, 4, 5$ 的情况下，他们的三个评判指标 RE, MSE, RMSE 都是优于其他所有模型的。其中在 $T = 5$ 时预测效果最好。这也印证了本文所提出的基于时间步长的改良组合模型的有效性。

5. 结论

本文提出了一种基于贝叶斯方法以及最大熵原理的模型组合方法进行失效时间的短期预测，通过递归的形式求出每个失效时间预测时各个基模型所占的权重值，并在此基础上进行了改良，将递归的次数固定为步长 T 而不是递归到初始时间。实验证明了 COM 模型的有效性以及不足之处，而 COM-T 模型

针对性地改进了原 COM 模型的不足, 不考虑久远前信息, 在预测精度上有很高的提升。结果说明, 本文提出的 COM-T 模型的短期失效时间预测效果要优于传统的 NHPP 模型以及等权组合的 ELC 模型, 是一种泛用性较强的组合模型。

参考文献

- [1] Humphrey, W.S. (2001) The Future of Software Engineering: I. *Watts New Column, News at SEI*, **4**.
- [2] Duane, J.T. (1964) Learning Curve Approach to Reliability Monitoring. *IEEE Transactions on Aerospace*, **2**, 563-566.
- [3] Jelinski, Z. and Moranda, P.B. (1972) Software Reliability Research. In: *Statistical Computer Performance Evaluation*, Academic Press, 465-484. <https://doi.org/10.1016/B978-0-12-266950-7.50028-1>
- [4] Schneidewind, N.F. (1975) Analysis of Error Processes in Computer Software. *AcmSigplan Notices*, **10**, 337-346. <https://doi.org/10.1145/390016.808456>
- [5] Musa, J.D. and Okumoto, K. (1984) A Logarithmic Poisson Execution Time Model for Software Reliability Measurement. *International Conference on Software Engineering*, 230-238.
- [6] Goel, A.L. and Okumoto, K. (1979) Time-Dependent Error-Detection Rate Model for Software Reliability and Other Performance Measures. *IEEE Transactions on Reliability*, **28**, 206-211. <https://doi.org/10.1109/TR.1979.5220566>
- [7] Wu, W., Han, K., He, C.M. and Wu, S.J. (2012) A Dynamically-Weighted Software Reliability Combination Model. *International Conference on Quality*, Chengdu, 15-18 June 2012, 148-151. <https://doi.org/10.1109/ICQR2MSE.2012.6246207>
- [8] Zhang, Q.Y., Zhu, X.M. and Xu, K. (2011) Combination Forecasting on Software Reliability Based on Entropy Weight. *International Conference on Electronic & Mechanical Engineering & Information Technology*, Harbin, 12-14 August 2011, 3095-3097. <https://doi.org/10.1109/EMBIT.2011.6023742>
- [9] Su, Y.S., and Huang, C.Y. (2007) Neural-Network-Based Approaches for Software Reliability Estimation Using Dynamic Weighted Combinational Models. *Journal of Systems & Software*, **80**, 606-615. <https://doi.org/10.1016/j.jss.2006.06.017>
- [10] Sarishvili, A. (2013) Software Reliability Prediction via Two Different Implementations of Bayesian Model Averaging. *Solving Complex Machine Learning Problems with Ensemble Methods*.
- [11] Wood, A. (1996) Predicting Software Reliability. *IEEE Computer*, **29**, 69-77. <https://doi.org/10.1109/2.544240>
- [12] Mohanty, R., Ravi, V. and Patra, M.R. (2013) Hybrid Intelligent Systems for Predicting Software Reliability. *Applied Soft Computing*, **13**, 189-200. <https://doi.org/10.1016/j.asoc.2012.08.015>