CSR: SMALL: Robust Algorithms for an Open Source Software Reliability Tool

UMass | Dartmouth

Lance Fiondella

University of Massachusetts, North Dartmouth, MA 02747

Abstract

Systems increasingly depend on software
- Mission and life critical
- Must preserve high reliability and availability

Rush to deploy new technologies may result in
- Inadequate reliability testing
- Severe economic damage and loss of life

Recent National Academies report *Reliability Growth: Enhancing Defense System Reliability (2015)* recommends:
- *"Use of reliability growth models to direct contractor design and test activities"*

Contributions:
- **Free and open source tool for users** http://sasdlc.org/lab/projects/srt.html
- **Collaborative environment for researchers**
- **Expectation conditional maximization (ECM) algorithms to ensure stability of convergence**

Goals and Objectives

Software Failure and Reliability Assessment Tool (SFRAT)
- Designed for practitioner and research community
- Programmed in R and provides functionality through Shiny graphical user interface
- Reduces the need for knowledge of the underlying statistical techniques
- Can help user quantitatively assess software as part of their data collection and reporting process

Allows users to answer following questions about a software system during test
1. Is software ready to release (Has it achieved a specified reliability goal)?
2. How much more time and test effort will be required to achieve a specified reliability goal?
3. What will be the consequences to a system's operational reliability if not enough testing resources are available?

Input Data Format

1. **Inter-failure (IF) times data:** Time between $(i-1)^{st}$ and i^{th} failure, $t_i = (T_i - T_{i-1})$
2. **Failure times (FT) data:** Vector of failure times, $\mathbf{T} =< t_i, t_2, \ldots, t_n >$
3. **Failure count data:** Length of interval and number of failures observed within it, $< \mathbf{T}, \mathbf{K} > = < (t_i, k_1), (t_2, k_2), \ldots, (t_n, k_n) >$

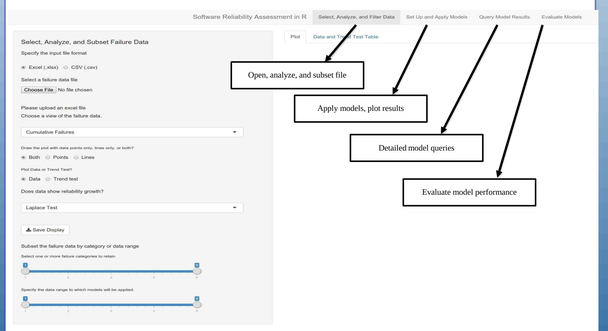
Input File format: Excel or CSV

	FN	IF	FT
2	1	3	3
3	2	30	33
4	3	113	146
5	4	81	227
6	5	115	342
7	…	…	…

Output

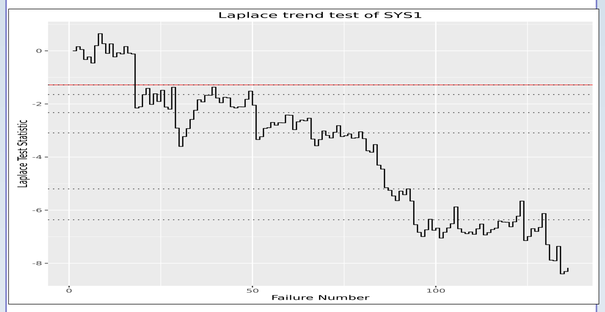
1. Trend tests (Assess if data exhibits reliability growth)
2. Model rankings (Select for prediction)
3. Visualization
   - Cumulative failure, time between failure, failure intensity, and reliability growth plots
4. Predictions
   - Time to achieve reliability
   - Expected number of faults for next t time units
   - Expected time to next k failures

SFRAT Tab view



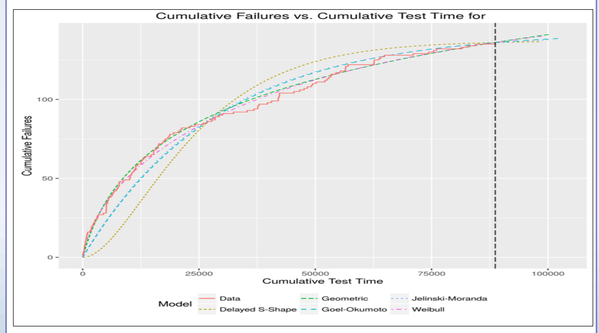
Tabs enable data input, visualization, model application, model query, and goodness of fit assessment.

Tab1: Laplace and Running arithmetic average



Decreasing trend indicates reliability growth. Red line indicates 90% confidence. Needed for model fitting algorithms to converge and predictions to be accurate.

Tab2: Model fit



Model fits (smooth lines) can be compared with data (red staircase plot). Predictions (beyond dotted black line).

Tab3: Predictions

Model	Time to achieve 90% reliability for mission of length 4116	Expected # of failures for next 4116 time units	Nth failure	Expected times to next 1 failures
Delayed S-shaped	12401.15	0.246856	1	N/A
Jelinski-Moranda	59915.29	0.856125	1	4869.807
Goel-Okumoto	62829.77	0.903615	1	4591.285
Weibull	259865.77	1.725954	1	2353.053
Geometric	1592716.46	1.877473	1	2170.031

Tab 4: Model assessment

Model	AIC	PSSE
Delayed S-shaped	2075.146	296.34925
Geometric	**1937.034**	84.32708
Goel-Okumoto	1953.613	23.07129
Jelinski-Moranda	1950.534	**19.60037**
Weibull	1938.161	74.94496

Models with lower AIC (Akaike information criterion) and PSSE (predictive sum of squares error) preferred.

Conclusions and Future Work

•Conclusions
- Free and open source tool to promote collaboration
- Application architecture enables integration of models and measures from research literature
- ECM algorithms to improve stability of model fitting

•Future research
- Expand architecture to encompass additional stages of software development lifecycle
- Explore applications to system level assurance of cyber security
- Multi-stage model fitting algorithms to improve speed and stability of convergence

Acknowledgement

This work was supported the National Science Foundation (NSF) (#1526128).

Author Contact Information

Lance Fiondella
Department of Electrical and Computer Engineering
University of Massachusetts Dartmouth
285 Old Westport Road North Dartmouth, MA USA
Email: lfiondella@umassd.edu