Covariate Software Reliability Models and Applications

Lance Fiondella
Department of Electrical and Computer Engineering

Some high profile failures attributed to software

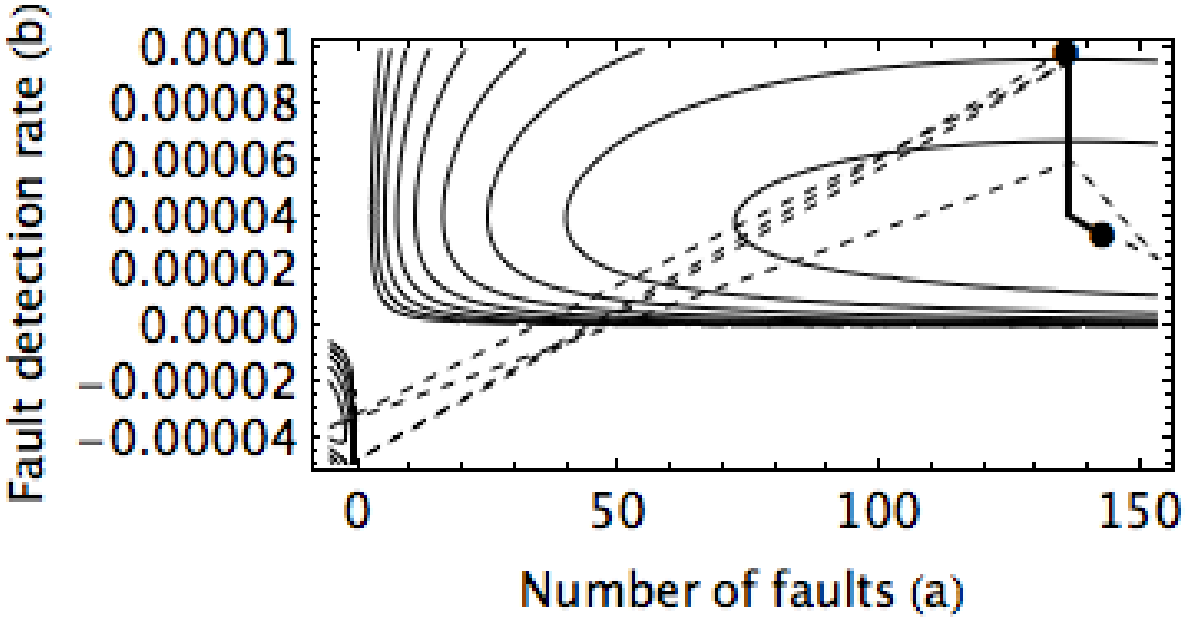
- 1991: MIM-104 Patriot Missile
  - 28 soldiers killed, 99 wounded
- 1996: Ariane 5 Flight 501
  - $370 million
- 1998: Mars Climate Orbiter
  - $193 million
- 1999: Mars Polar Lander
  - $110 million

Prominent reports on reliability

- 2002: NIST
  - *The Economic Impacts of Inadequate Infrastructure for Software Testing* (22.2,59.5) billion
- 2008: DSB
  - *Report of the Defense Science Board Task Force on Developmental Test & Evaluation*
    - The single most important step ... ensure programs are formulated to execute a viable systems engineering strategy from the beginning, including a robust RAM program.
- 2014: National Academies
  - *Reliability Growth: Enhancing Defense System Reliability*
    - Recommended use of reliability growth models to direct contractor design and test activities

Percentage of system functions performed by software

Technical challenge: Stability of underlying model fitting algorithms

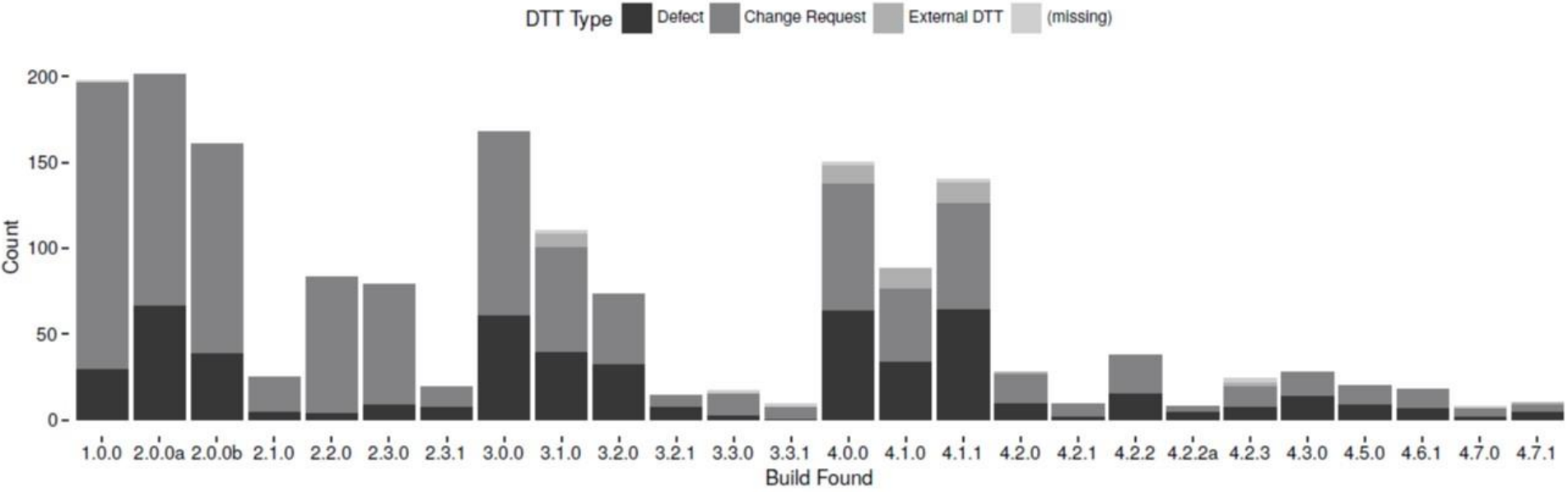


Initial estimates can behave erratically

Recent Contributions

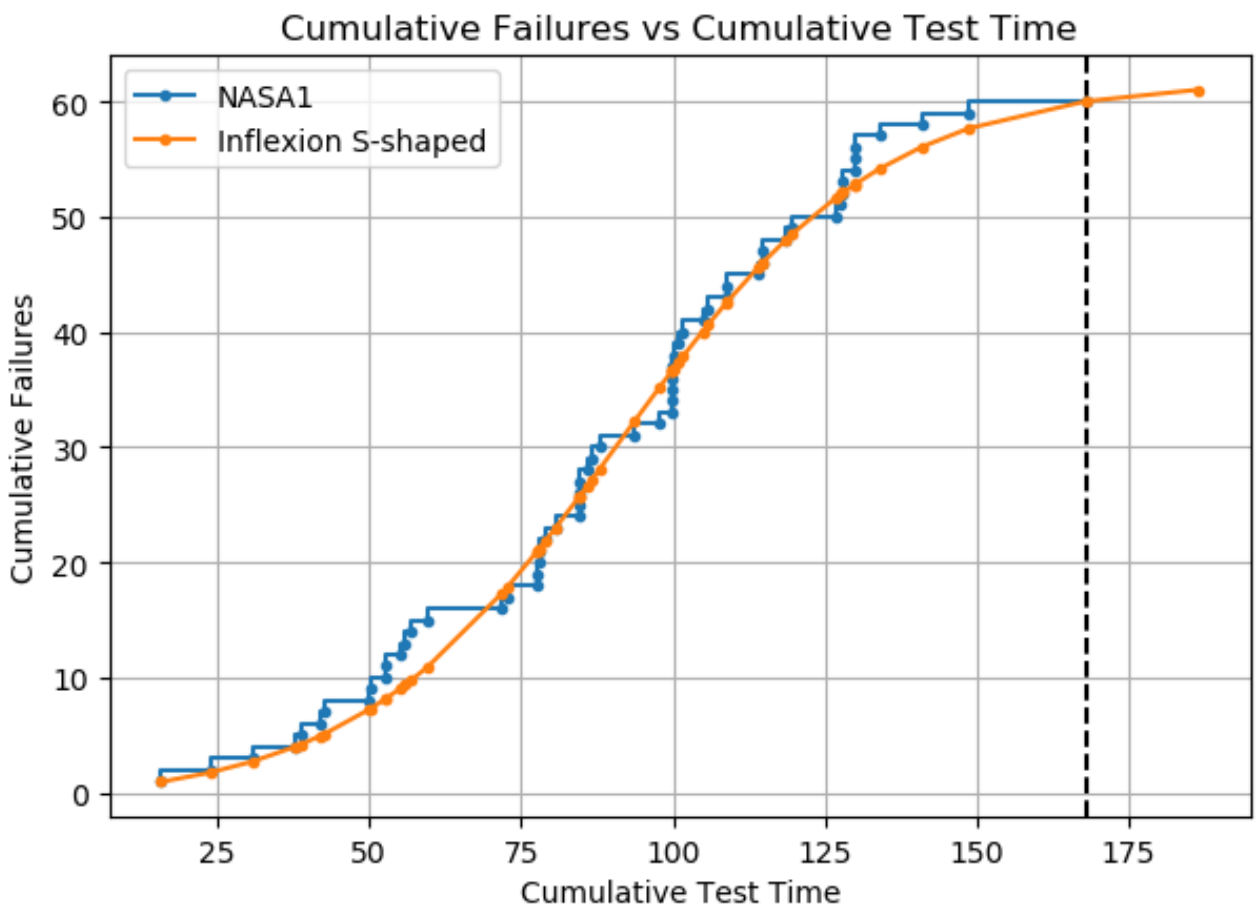
- Developed open source software reliability tool (SFRAT) to automatically apply reliability models and scripts to further automate
- Expectation conditional maximization (ECM) algorithms
  - Eliminated computationally intensive integration from update rules of ECM
  - Guarantees dimensionality reduction
  - 200-400 factor speed up over past EM and ECM algorithms (conservative estimate)

Summary of frequencies by DTT type and build found



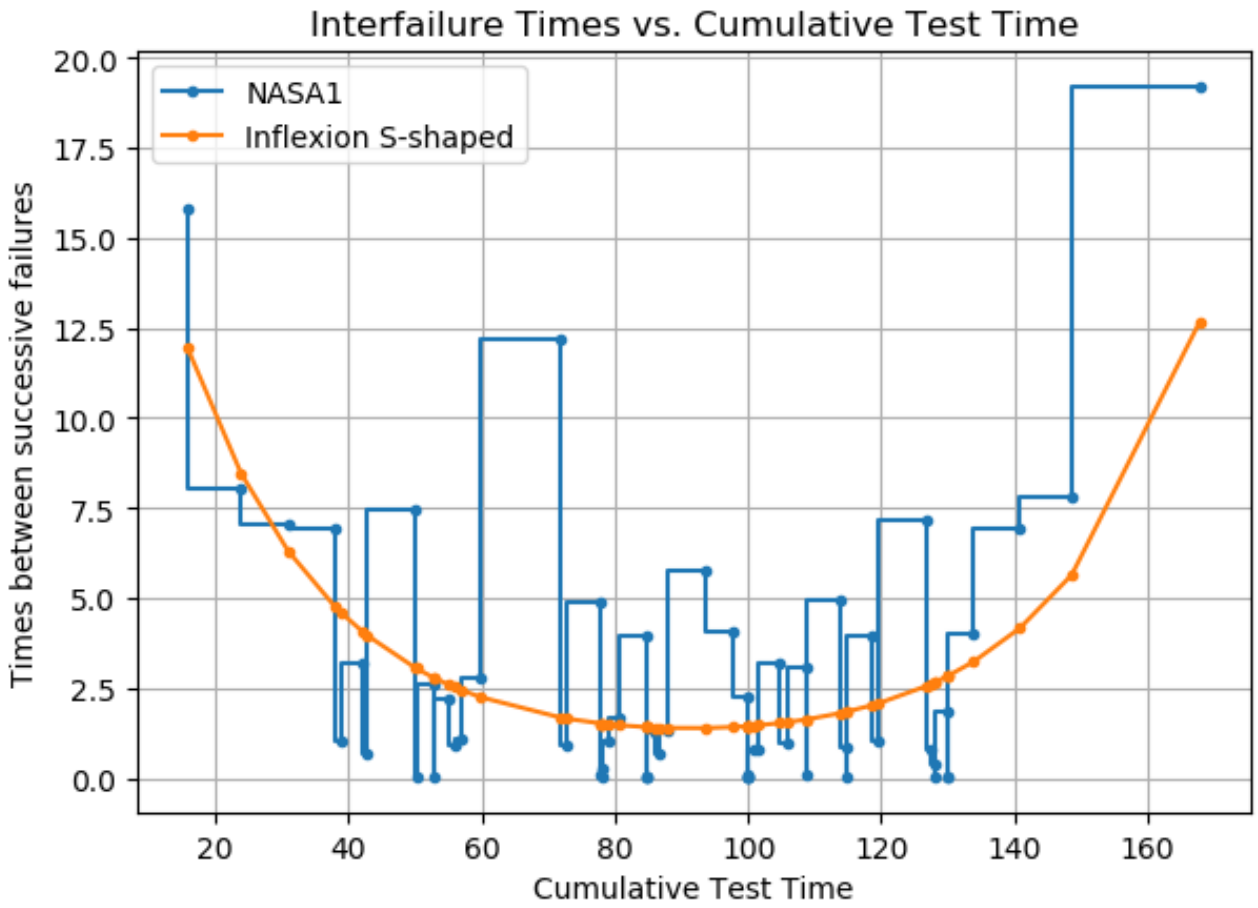
Extracted defects and change requests from major and minor versions exhibiting a large number of events

Cumulative failures



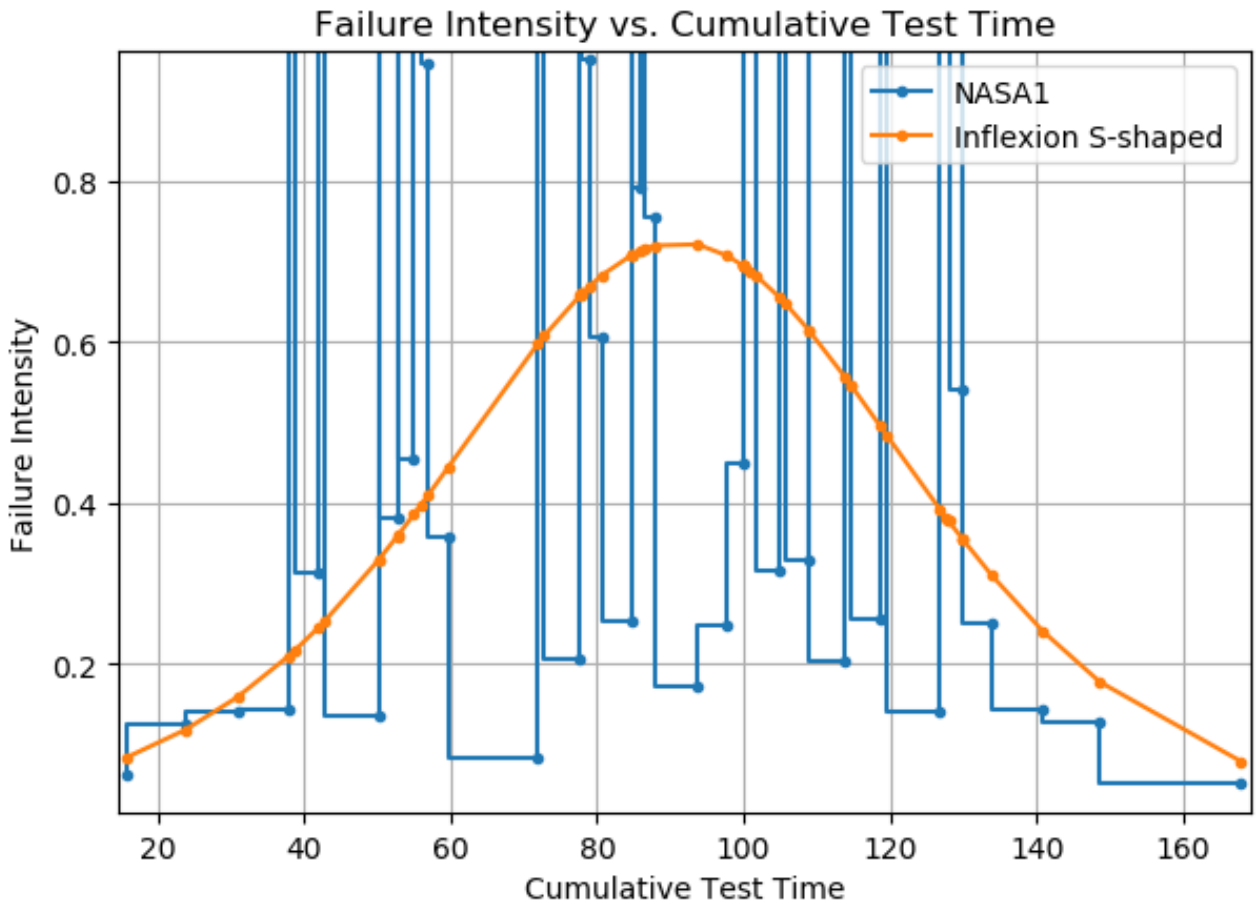
Plot enables comparison of data and model fits

Time between failures



Times between failures should increase (indicates reliability growth)

Failure intensity



Failure intensity should decrease (indicates reliability growth)

File View Windows

PROJECT // 057813
12:48:31

PROJECT // 15789
03:45:49

PROJECT // 04127
23:18:04

DATABASE SEARCH

TOTAL TRACKING
150,338

	LEO	MEO	HEO
FUNCTIONAL	465	454	49
NON-FUNCTIONAL	1,265	730	193
ROCKET BODIES	786	381	499
DEBRIS	126,613	2,925	15,978

0000080

0054000

TRACKING FIELD

SPACE FENCE SITE 3

KWAJ / -8° 43' 45.03", +167° 44' 24.11"

CHECK-IN COMPLETE

COMMON NAME:	CATALOG #:
H-2A DEB	36874
PROGRESS M-05M	36521
COSMOS 2251 DEB	35793
COSMOS 1275 DEB	14629

NEXT CHECK-IN ITEMS

COMMON NAME:	CATALOG #:
MINOTAUR R/B	35006
COSMOS 1484 DEB	26920
NAVSTAR 22	20959
NOAA 12	21263

STATION MONITOR

SYSTEM HISTORY

PREFERENCES

UNIVERSAL TIME

12:48:31

12:48:03
00:27:54
09:41:25

DISPLAY SETTINGS

- LEO
- MEO
- HEO
- FUNCTIONAL PAYLOADS
- NON-FUNCTIONAL PAYLOADS
- ROCKETS
- DEBRIS

LAUNCH SCHEDULE

13	19	21	12
DAYS	HOURS	MINUTES	SECONDS

VEHICLE NAME: RSGT-06248 | 1547-1034-05

RE-ENTRY SCHEDULE

21	04	32	46
DAYS	HOURS	MINUTES	SECONDS

VEHICLE NAME: SDVS-13486 | 1648-3482-02

OBJECT WATCHLIST

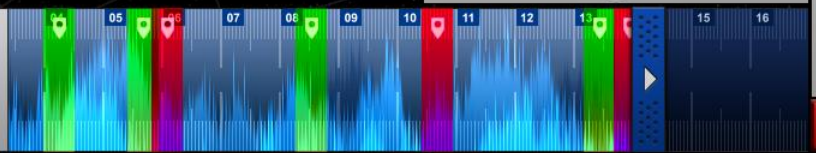
COMMON NAME:	CATALOG #:
SL-12 DEB	37148
UARS DEB	37195
ISS DEB	36824
ISS DEB	36826
IRIDIUM 33 DEB	36562
IRIDIUM 33 DEB	36563
IRIDIUM 33 DEB	36564
IRIDIUM 33 DEB	36565

LAUNCH AVOIDANCE MANEUVER

SPACE FENCE SITE 1 — CONNECTED — STABLE

SPACE FENCE SITE 2 — CONNECTED — STABLE

SPACE FENCE SITE 3 — CONNECTED — STABLE



Nonhomogeneous Poisson process (NHPP) Software Reliability Model

- Stochastic process counts number of events observed as function of time
  - In context of software reliability (security), NHPP counts number of faults (vulnerabilities) detected by time t
- Counting process characterized by mean value function (MVF)
  - Form of MVF of several SRM: $m(t) = \omega \times F(t)$
    - ω - number of faults (vulnerabilities) that would be detected with indefinite testing
    - $F(t)$ - cumulative distribution function (CDF)

Weibull Software Reliability Model

- Substituting Weibull distribution for $F(t)$

$$m(t) = \omega\left(1 - e^{-bt^c}\right)$$

  - b - scale parameter
  - c - shape parameters
- $c = 1$ simplifies to exponential distribution
  - Also known as Goel-Okumoto Model

Primary limitation of traditional NHPP software reliability models: Do not explicitly identify underlying software testing activities that lead to fault discovery

Contributions

- A NHPP software reliability model possessing a discrete Cox proportional hazard rate to incorporate covariates
- Expectation conditional maximization (ECM) algorithms to efficiently estimate a model's numerical parameters
- A generalization of the testing effort allocation problem to covariate models referred to as the *optimal testing activity allocation problem* to
  - (i) Maximize fault discovery within a budget constraint
  - (ii) Minimize budget required discover a specified number of additional faults

NHPP SRM with Covariates

- Mean value function

$$H_{n;\omega,\theta,\boldsymbol{\beta}} = \omega \sum_{i=1}^{n} p_{i,\mathbf{x}_i;\theta,\boldsymbol{\beta}}$$

Number of faults that would be detected with indefinite testing

where

Discrete Cox proportional hazard model

$$p_{i,\mathbf{x}_i;\theta,\boldsymbol{\beta}} = \left(1-\left(1-h^0_{i;\theta}\right)^{g(\mathbf{x}_i;\boldsymbol{\beta})}\right)\prod_{k=1}^{i-1}\left(1-h^0_{k;\theta}\right)^{g(\mathbf{x}_k;\boldsymbol{\beta})}$$

Covariates associated with y_i in the ith testing interval

Baseline hazard function

Vector of Cox model parameters

Vector of model parameters

Baseline Hazard Functions

- Geometric (GM):

$$h_{i;b}^{0} = b$$

  - where $b \in (0,1)$
- Negative binomial of order two (NB):

$$h_{i;b}^{0} = \frac{ib^2}{1 + b(i-1)}$$

  - where $b \in (0,1)$ and exponent 2 indicates order
- Discrete Weibull of order two (DW):

$$h_{i;b}^{0} = 1 - b^{i^2-(i-1)^2}$$

  - where $b \in (0,1)$ and exponent 2 indicates order

Optimal test activity allocation to maximize fault discovery

- Given n intervals of observed data and budget B to allocate among r activities (covariates)
- Maximize total number of faults detected

$$\arg\max \widehat{H}_{(n+1);\omega,\theta,\boldsymbol{\beta}}$$

subject to

$$\sum_{j=1}^{r} \mathbf{c}_j \mathbf{x}_{j,(n+1)} \leq B$$

$$\mathbf{c}_j \left(\mathbf{x}_{j,(n+1)} - \mathbf{x}_{j,(n)} \right) \geq 0$$

– $\mathbf{c}_j > 0$ - cost associated with additional unit of activity j

Optimal test activity allocation to minimize cost

- Minimize cost to identify a k faults

$$\arg\min \sum \mathbf{c}_j \mathbf{x}_j$$

subject to

$$\widehat{H}_{(n+1);\omega,\theta,\boldsymbol{\beta}} - \widehat{H}_{n\,;\omega,\theta,\boldsymbol{\beta}} \geq k$$

$$\beta_{j+1} - \beta_j \geq 0$$

Parameter Estimation: Maximum Likelihood Estimation (MLE)

- Maximizes likelihood function, also known as joint distribution of failure data
- Likelihood function

$$\mathbf{L}(\theta, \boldsymbol{\beta}, \omega) = \Pr(Y_1 = y_1, Y_2 = y_2, \dots, Y_n = y_n)$$
$$= \prod_{i=1}^{n} \exp\left(-\omega p_{i,\mathbf{x}_i;\theta,\boldsymbol{\beta}}\right) \frac{\left(\omega p_{i,\mathbf{x}_i;\theta,\boldsymbol{\beta}}\right)^{y_i}}{y_i!}$$

- Log-likelihood function

$$LL(\theta, \boldsymbol{\beta}, \omega) = -\omega \sum_{i=1}^{n} p_{i,\mathbf{x}_i;\theta,\boldsymbol{\beta}} + \sum_{i=1}^{n} y_i \ln(\omega) + \sum_{i=1}^{n} y_i \ln\left(p_{i,\mathbf{x}_i;\theta,\boldsymbol{\beta}}\right) - \sum_{i=1}^{n} \ln(y_i!)$$

Expectation Conditional Maximization Algorithm (ECM)

- Preserves monotonicity property of EM algorithm
- Replaces M-step of EM algorithm with ν conditional maximization (CM)-steps
  - ν - number of model parameters
  - Divides single ν-dimensional problem into ν 1-dimensional problems

ECM Algorithm (2)

- (S.1) – Specify log-likelihood function
- (S.2) – Reduce log-likelihood function from ν to $\nu - 1$ parameter

$$\frac{\partial LL}{\partial \omega} = 0$$

and

$$\widehat{\omega} = \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} p_{i,\mathbf{x}_i;\theta,\boldsymbol{\beta}}}$$

- Substitute for $\widehat{\omega}$ in LL function to obtain reduced log-likelihood (RLL) function

ECM Algorithm (3)

- (S.3) – Derive conditional maximization steps for remaining $\nu - 1$ parameters by computing partial derivatives

$$\frac{\partial RLL}{\partial \theta} = \mathbf{0}$$

and

$$\frac{\partial RLL}{\partial \boldsymbol{\beta}} = \mathbf{0}$$

θ - all parameters except ω

$\boldsymbol{\beta}$ – Coefficients related to all p covariates

ECM Algorithm (4)

- (S.4) – Cycle through $(\nu - 1)$ CM-steps holding other $(\nu - 2)$ parameters constant until user specified convergence achieved
  - Applies numerical root finding algorithm in each step
  - Determines MLEs of $\hat{\theta}/\omega$
- (S.5) – Computes MLE of ω by substituting estimates of θ including $\boldsymbol{\beta}$ into $\widehat{\omega}$ equation
- Step (S.1) through (S.5) applied to different hazard functions

Goodness-of-fit measures

- Akaike Information Criterion (AIC)
  - Information theoretic goodness-of-fit measure
  - Quantifies trade-off between model precision and complexity
  - AIC of model i function of maximized log-likelihood and number of model parameters

$$AIC_i = 2\nu - 2LL\left(\mathbf{x}_i; \widehat{\omega}, \hat{\theta}, \widehat{\boldsymbol{\beta}}\right)$$

  - Model j better than model i if $AIC_{i,j} = AIC_i - AIC_j > 2.0$

Goodness-of-fit measures (2)

- Bayesian Information Criterion (BIC)
  - Function of maximized log-likelihood, number of model parameters, and sample size n

$$BIC_i = -2LL\left(\mathbf{x}_i; \widehat{\omega}, \hat{\theta}, \widehat{\boldsymbol{\beta}}\right) + \nu \log(n)$$

  - Penalty term proportional to number of parameters multiplied by logarithm of sample size

Goodness-of-fit measures (3)

- Sum of Squares Error (SSE)
  - Also known as residual sum of squares
  - For failure count data

$$SSE = \sum_{i=1}^{n} \left(\widehat{H}_{i;\omega,\theta,\beta} - Y_i\right)^2$$

  - $Y_i = \sum_{j=1}^{i} y_i$ - cumulative number of failures observed in first i time intervals

Goodness-of-fit measures (4)

- Predictive Sum of Squares Error (PSSE)
  - Compares predictions of model with data not used to perform model fitting

$$PSSE = \sum_{i=n-\ell+1}^{n} \left(\widehat{H}_{i;\omega,\theta,\beta} - Y_i\right)^2$$

  - Maximum likelihood estimates of model parameters determined from first $(n - \ell)$ intervals
  - Common choice for ℓ approximately 10% of data

Model Selection based on Multiple Goodness-of-fit Measures

- Consider n models and m measures
  - Assign normalized score to each measure based on Critic method

$$x_{i,j} = \frac{f_{i,j} - f_j^+}{f_j^- - f_j^+}$$

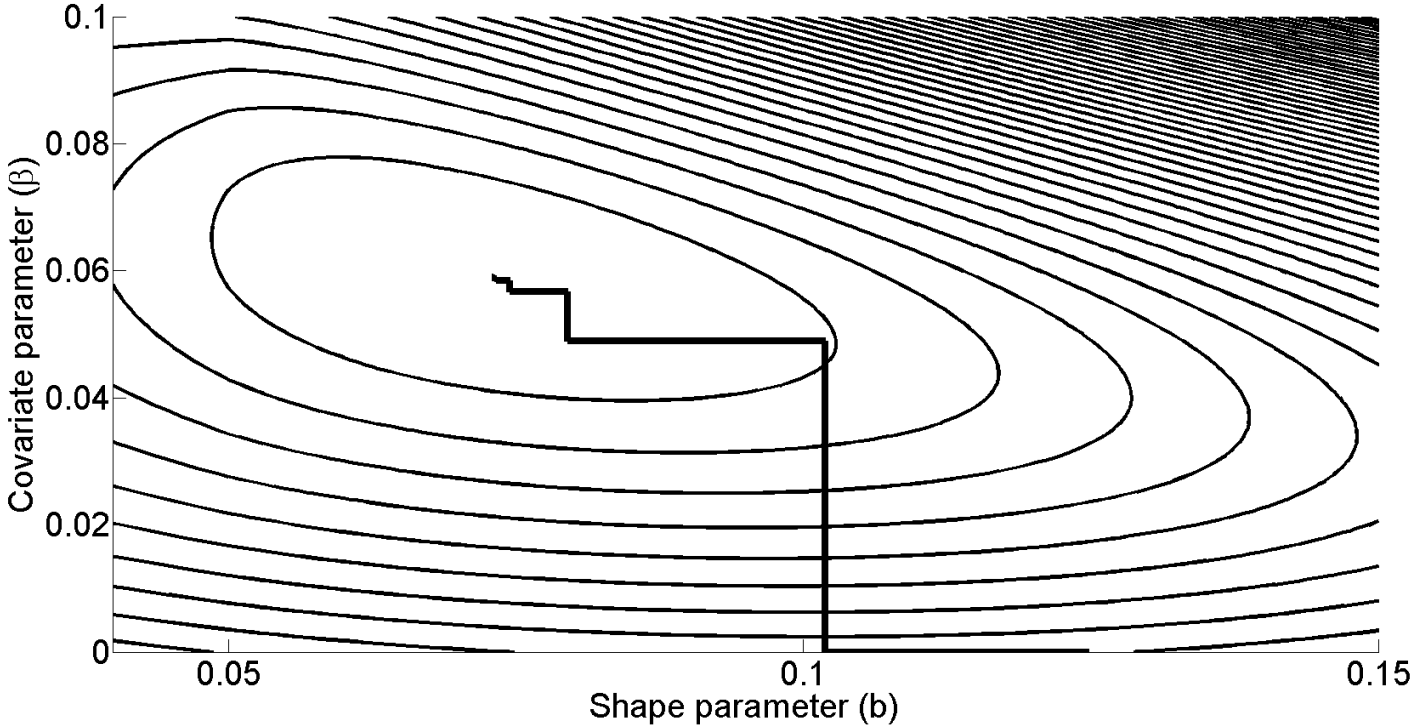
  - f_j^+ (f_j^-) - best (worst) value of measure j across all models
  - $x_{i,j}$ - indicates how close j^{th} measure of model i to ideal
- Compute median or mean of normalized scores
  - Recommend model with highest scores

Illustrations

- DS1 and DS2 with three covariates considered
  - DS1 with $n = 17$ intervals
  - DS2 with $n = 14$ intervals
- Covariates considered – E, F, C
  - E – Execution time
  - F – Failure identification work
  - C – Computer time identification work
- Hazard functions considered
  - GM, NB, and DW

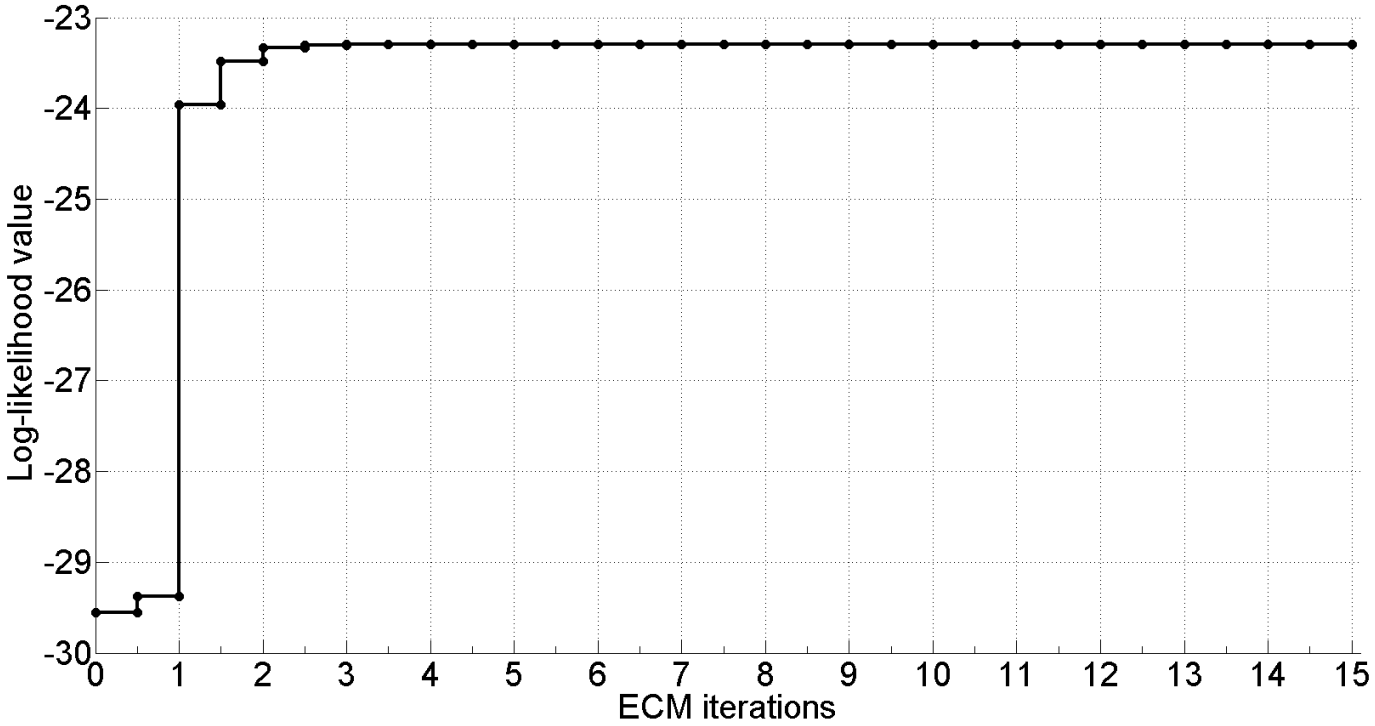
week	Execution Time (hr)	Failure Identification Work (person hr)	Computer Time-Failure Ident. (hr)	Failure Identified
1	.0531	4	1.0	1
2	.0619	20	0	1
3	.1580	1	0.5	2
4	.0810	1	0.5	1
5	1.0460	32	2.0	8
6	1.7500	32	5.0	9
7	2.9600	24	4.5	6
8	4.9700	24	2.5	7
9	0.4200	24	4.0	4
10	4.7000	30	2.0	3
11	0.9000	0	0	0
12	1.5000	8	4.0	4
13	2.0000	8	6.0	1
14	1.2000	12	4.0	0
15	1.2000	20	6.0	2
16	2.2000	32	10.0	2
17	7.6000	24	8.0	3
total	32.8000	296	60.0	54

Application of ECM to Covariate NHPP SRM: GM (F) on DS2



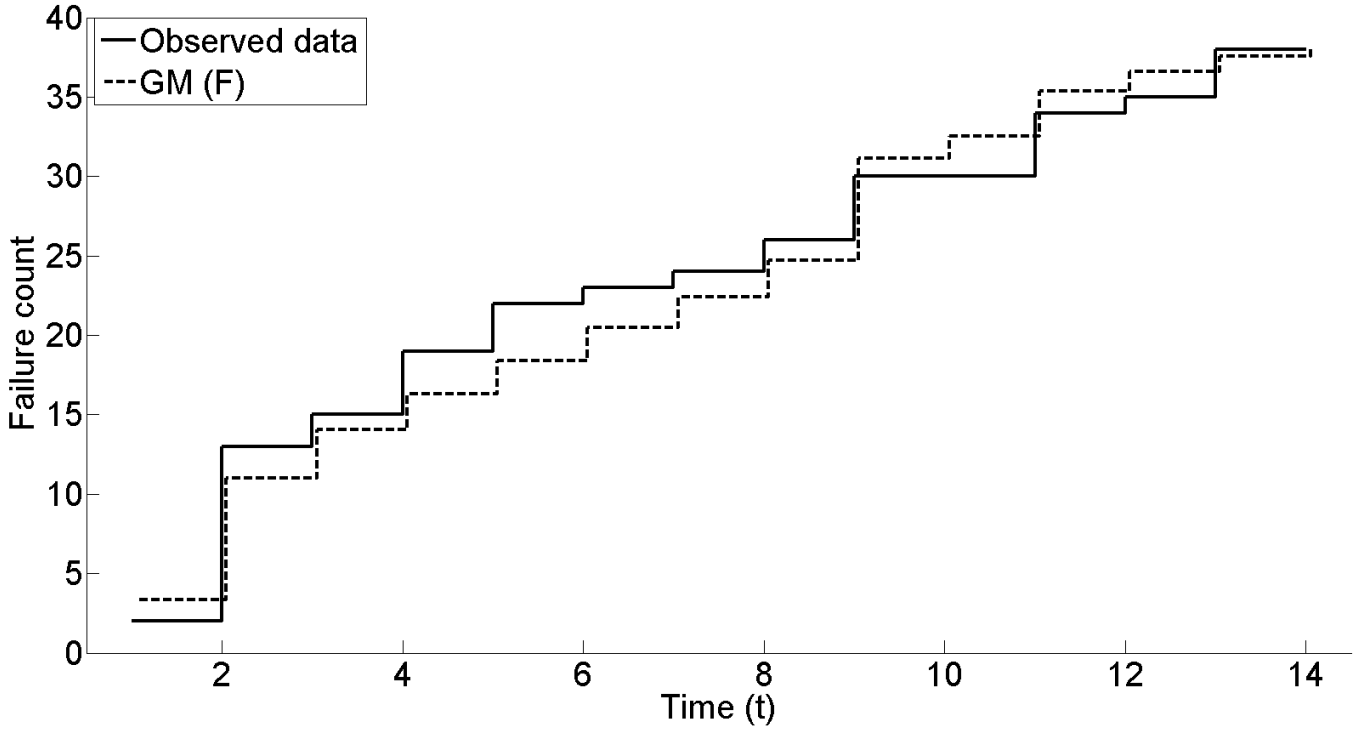
ECM takes 90° steps in search space

Log-likelihood of covariate model: GM (F) on DS2



ECM algorithm converges monotonically

Model Fit: GM (F) on DS2



Model prediction function of covariates in each interval

Goodness-of-fit measures for GM on DS1 and DS2

β	ν	DS1					DS2				
		LLF	AIC	BIC	SSE	PSSE	LLF	AIC	BIC	SSE	PSSE
-	2	−41.47	86.94	88.60	433.73	**0.25**	−29.38	62.76	64.03	45.04	**3.16**
E	3	−36.13	78.25	80.75	187.00	5.79	−24.93	55.86	57.78	80.29	12.56
F	3	−31.21	68.43	**70.93**	120.12	27.41	−23.29	**52.58**	**54.51**	49.61	10.22
C	3	−35.54	77.09	79.59	219.15	4.34	−24.33	54.65	56.57	**44.25**	14.06
EF	4	−30.03	68.05	71.38	73.41	17.35	−23.27	54.54	57.11	54.24	9.98
FC	4	−30.89	69.79	73.12	128.25	85.20	−23.01	54.03	56.58	48.63	12.40
EC	4	−30.08	68.16	71.49	48.25	2.35	−24.09	56.19	58.74	61.53	14.37
EFC	5	**−28.40**	**66.81**	70.97	**36.95**	32.99	**−23.00**	56.01	59.21	45.77	11.58

No combination of covariates preferred by all measures

Model Selection – Normalized scores – GM on DS1

	–	E	F	C	EF	FC	EC	EFC
LLF	0.00	0.41	0.79	0.45	0.88	0.81	0.87	1.00
AIC	0.00	0.43	0.92	0.49	0.94	0.85	0.93	1.00
BIC	0.00	0.44	1.00	0.51	0.97	0.88	0.97	0.99
SSE	0.00	0.62	0.79	0.54	0.91	0.77	0.97	1.00
PSSE	1.00	0.94	0.68	0.95	0.79	0.00	0.98	0.62
Median	0	0.44	0.79	0.51	0.91	0.81	0.97	**1.00**
Mean	0.2	0.57	0.84	0.59	0.91	0.66	**0.94**	0.92

Must exercise caution when selecting goodness of fit measures

Model Selection – Overall Recommendation

Method	DS1			DS2		
	1	2	3	1	2	3
Median	NB (F)	NB (EF)	NB (EFC)	GM (F)	GM (FC)	GM (EF)
Mean	NB (F)	NB (EF)	NB (EFC)	GM (F)	GM (EF)	GM (FC)

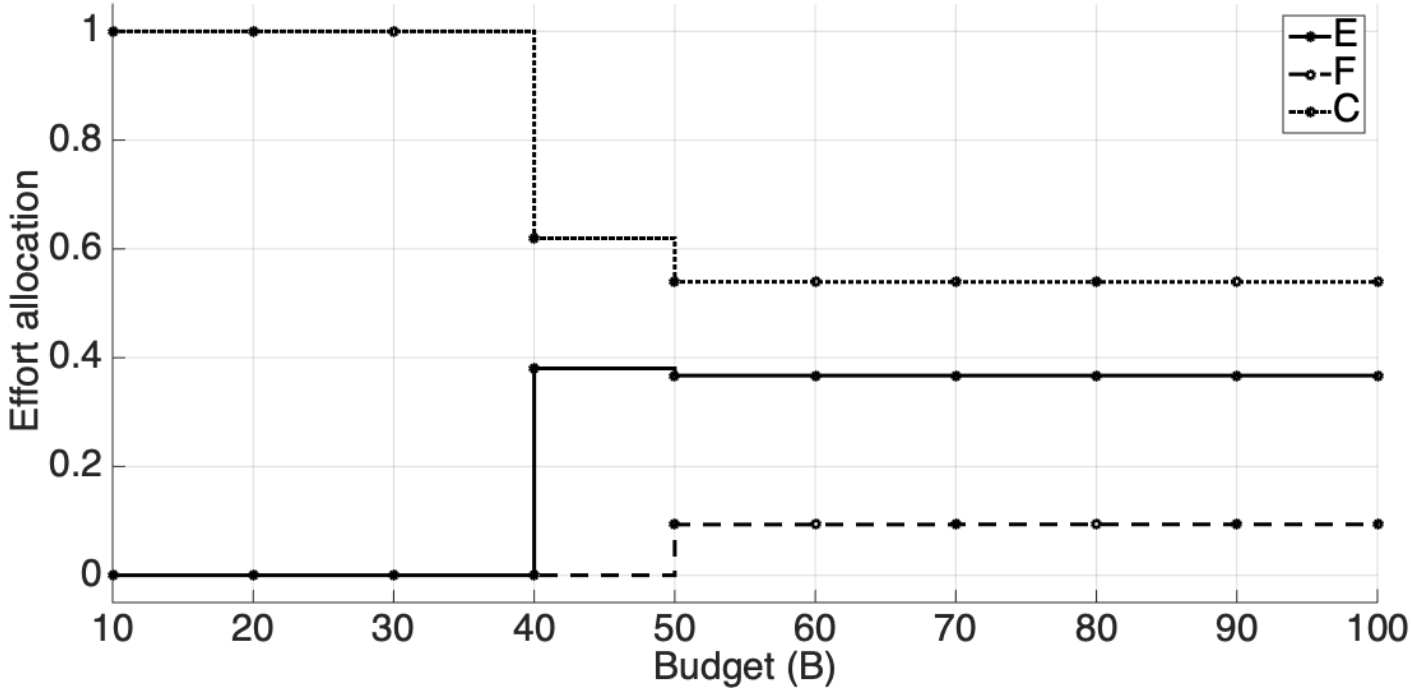
Must consider preferred combination of covariates and practical interpretation

Optimal Test Activity Allocation

β	$\widehat{H}^*_{n;\omega,\theta,\beta}$	$\%E$	$\%F$	$\%C$
E	6.63	100	–	–
F	2.61	–	100	–
C	0.41	–	–	100
EF	5.30	100	–	–
FC	1.97	–	5.70	94.30
EC	0.62	40.75	–	59.25
EFC	1.55	38.03	–	61.97

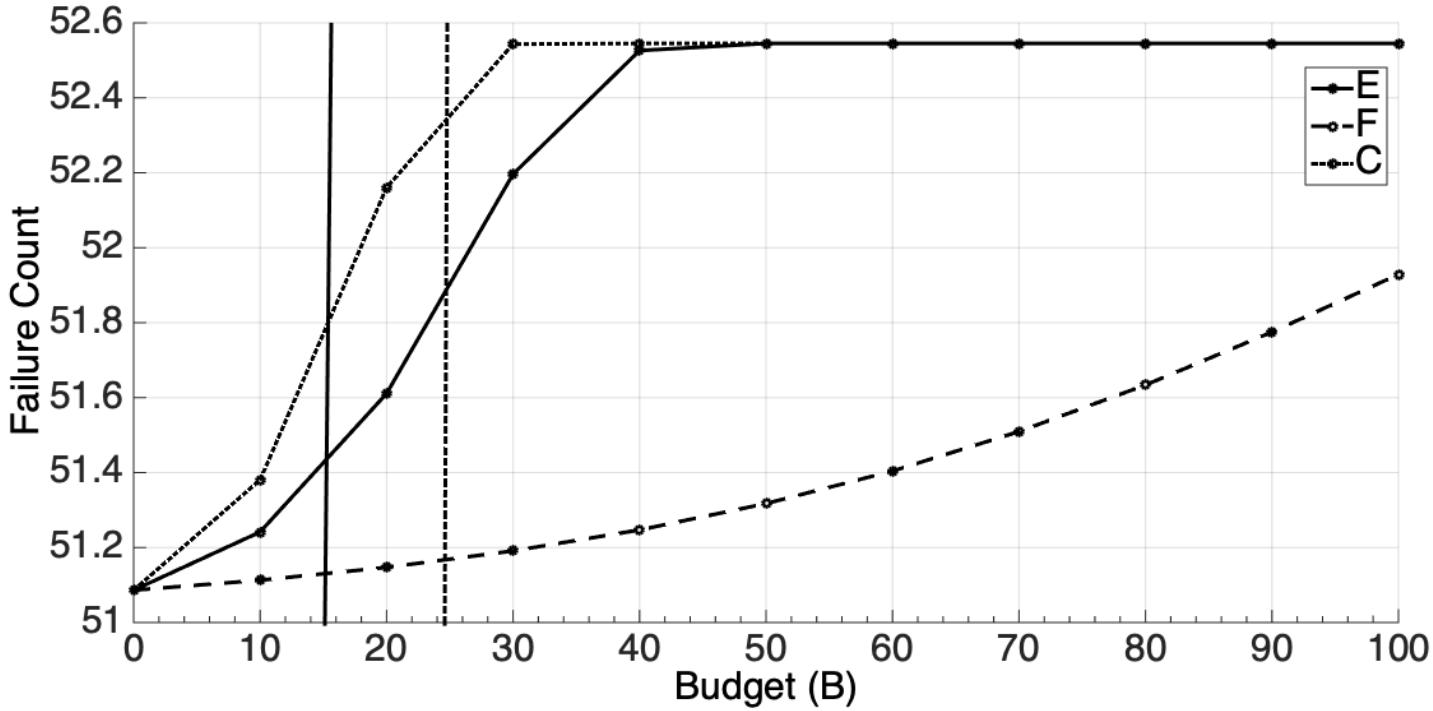
In practice, allocation may be performed on preferred model

Sensitivity of Optimal Allocation NB (EFC) on DS1



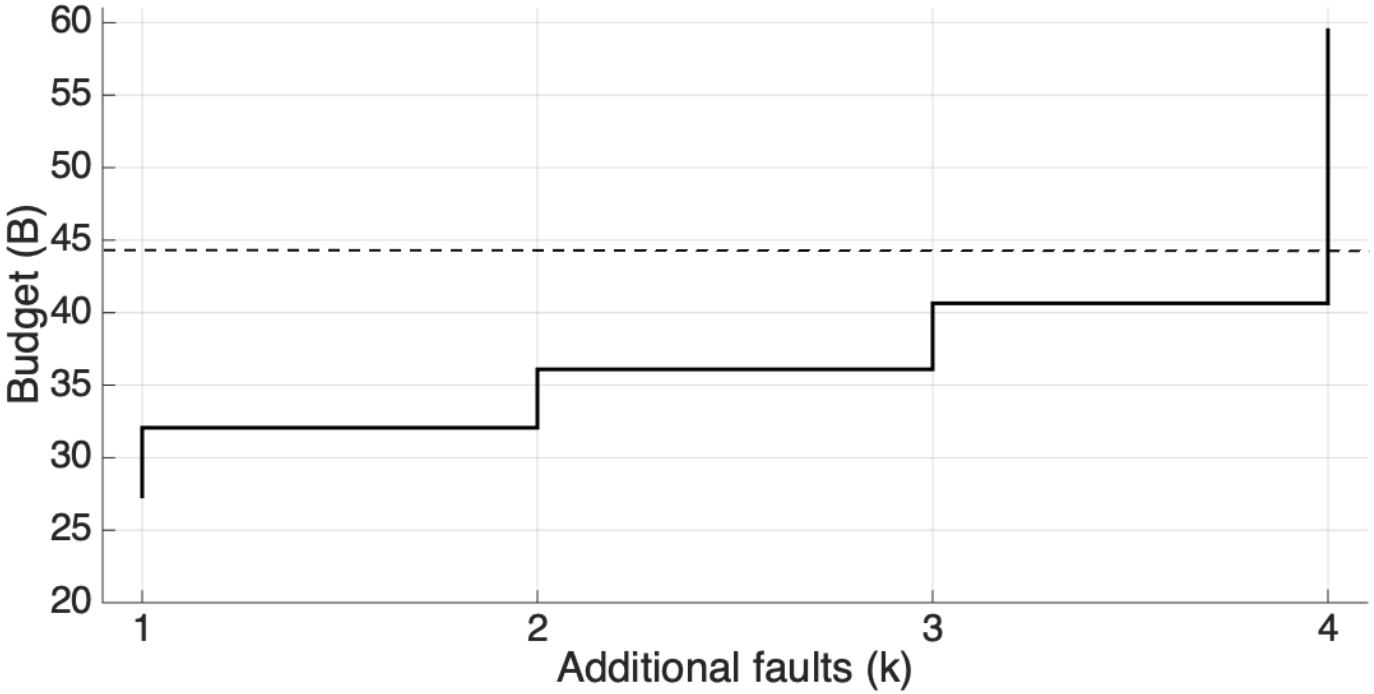
Percentage of effort allocated to each activity converges as budget increases

Marginal Utility of NB on DS1



Marginal utility of each activity explains allocation

Optimal test activity allocation to expose k additional faults



Performs better than allocation in final interval of original study, which only exposed two additional faults

Conclusions

- Proposed
  - NHPP software reliability model possessing discrete Cox proportional hazard rate to incorporate covariates
  - ECM algorithms to efficiently fit model
  - *Optimal testing activity allocation problem* to (i) maximize fault discovery within a budget constraint and (ii) minimize budget required to discover a specified number of additional faults
- Explicitly links underlying software testing activities that lead to fault discovery

Future Work & Research

- Tool implementation and automation to promote adoption
- Application to data from past and present programs (DoD, NASA, etc…) to make practical refinements
- Application in context of cybersecurity testing
- Process related studies to identify which combination of testing methods are most effective for a given class of systems

Software reliability tools

- Availability
  - Present (R)
    - https://sasdlc.org/lab/#/research
  - Forthcoming (R and Python)
    - https://lfiondella.sites.umassd.edu/
- Includes
  - Example failure data
  - Link to Github repository
  - Documentation

Acknowledgements

- This work was partially supported by (i) NAVAIR under contract N00421-16-P-0521, (ii) the National Science Foundation under awards #1526128 and #1749635, and NASA under grants #80NSSC18K0154 and 80NSSC20K0276.