Springer Series in Statistics

Advisors:

P. Bickel, P. Diggle, S. Fienberg, K. Krickeberg,
I. Olkin, N. Wermuth, S. Zeger

Springer Science+Business Media, LLC

Springer Series in Statistics

- Andersen/Borgan/Gill/Keiding*: Statistical Models Based on Counting Processes.
- Andrews/Herzberg*: Data: A Collection of Problems from Many Fields for the Student and Research Worker.
- Anscombe*: Computing in Statistical Science through APL.
- Berger*: Statistical Decision Theory and Bayesian Analysis, 2nd edition.
- Bolfarine/Zacks*: Prediction Theory for Finite Populations.
- Borg/Groenen*: Modern Multidimensional Scaling: Theory and Applications
- Brémaud*: Point Processes and Queues: Martingale Dynamics.
- Brockwell/Davis*: Time Series: Theory and Methods, 2nd edition.
- Daley/Vere-Jones*: An Introduction to the Theory of Point Processes.
- Dzhaparidze*: Parameter Estimation and Hypothesis Testing in Spectral Analysis of Stationary Time Series.
- Fahrmeir/Tutz*: Multivariate Statistical Modelling Based on Generalized Linear Models.
- Farebrother*: Fitting Linear Relationships: A History of the Calculus of Observations 1750 - 1900.
- Farrell*: Multivariate Calculation.
- Federer*: Statistical Design and Analysis for Intercropping Experiments, Volume I: Two Crops.
- Federer*: Statistical Design and Analysis for Intercropping Experiments, Volume II: Three or More Crops.
- Fienberg/Hoaglin/Kruskal/Tanur (Eds.)*: A Statistical Model: Frederick Mosteller's Contributions to Statistics, Science and Public Policy.
- Fisher/Sen*: The Collected Works of Wassily Hoeffding.
- Good*: Permutation Tests: A Practical Guide to Resampling Methods for Testing Hypotheses.
- Goodman/Kruskal*: Measures of Association for Cross Classifications.
- Gouriéroux*: ARCH Models and Financial Applications.
- Grandell*: Aspects of Risk Theory.
- Haberman*: Advanced Statistics, Volume I: Description of Populations.
- Hall*: The Bootstrap and Edgeworth Expansion.
- Härdle*: Smoothing Techniques: With Implementation in S.
- Hart*: Nonparametric Smoothing and Lack-of-Fit Tests.
- Hartigan*: Bayes Theory.
- Hedayat/Sloane/Stufken*: Orthogonal Arrays: Theory and Applications.
- Heyde*: Quasi-Likelihood and its Application: A General Approach to Optimal Parameter Estimation.
- Heyer*: Theory of Statistical Experiments.
- Huet/Bouvier/Gruet/Jolivet*: Statistical Tools for Nonlinear Regression: A Practical Guide with S-PLUS Examples.
- Jolliffe*: Principal Component Analysis.
- Kolen/Brennan*: Test Equating: Methods and Practices.
- Kotz/Johnson (Eds.)*: Breakthroughs in Statistics Volume I.

(continued after index)

Nozer D. Singpurwalla
Simon P. Wilson

Statistical Methods in Software Engineering

Reliability and Risk

With 55 Illustrations

Springer

Nozer D. Singpurwalla
Department of Operations Research
The George Washington University
Washington, DC 20052
USA
nozer@research.circ.gwu.edu

Simon P. Wilson
Department of Statistics
Trinity College
Dublin 2
Ireland
swilson@stats.tcd.ie

Library of Congress Cataloging-in-Publication Data
Singpurwalla, Nozer D.

Statistical methods in software engineering: reliability and risk

/Nozer D. Singpurwalla, Simon P. Wilson.

p. cm. — (Springer series in statistics)

Includes bibliographical references.

ISBN 978-1-4612-6820-8 ISBN 978-1-4612-0565-4 (eBook)

DOI 10.1007/978-1-4612-0565-4

1. Software engineering. 2. Statistical methods. I. Wilson, Simon P.

II. Title. III. Series.

QA76.758.S535 1999

005.1—dc21

99-14737

Printed on acid-free paper.

© 1999 Springer Science+Business Media New York

Originally published by Springer-Verlag New York in 1999

Softcover reprint of the hardcover 1st edition 1999

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher Springer Science+Business Media, LLC, except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed is forbidden.

The use of general descriptive names, trade names, trademarks, etc., in this publication, even if the former are not especially identified, is not to be taken as a sign that such names, as understood by the Trade Marks and Merchandise Marks Act, may accordingly be used freely by anyone.

Production managed by Frank McGuckin; manufacturing supervised by Nancy Wu.

Camera-ready copy provided by the author using EXP.

9 8 7 6 5 4 3 2 1

ISBN 978-1-4612-6820-8

Preface

This preface pertains to three issues that we would like to bring to the attention of the readers: our objectives, our intended audience, and the nature of the material.

We have in mind several objectives. The first is to establish a framework for dealing with uncertainties in software engineering, and for using quantitative measures for decision making in this context. The second is to bring into perspective the large body of work having statistical content that is relevant to software engineering, which may not have appeared in the traditional outlets devoted to it. Connected with this second objective is a desire to streamline and organize our own thinking and work in this area. Our third objective is to provide a platform that facilitates an interface between computer scientists and statisticians to address a class of problems in computer science. It appears that such an interface is necessary to provide the needed synergism for solving some difficult problems that the subject poses. Our final objective is to serve as an agent for stimulating more cross-disciplinary research in computer science and statistics. To what extent the material here will meet our objectives can only be assessed with the passage of time.

Our intended audience is computer scientists, software engineers, and reliability analysts, who have some exposure to probability and statistics. Applied statisticians interested in reliability problems are also a segment of our intended audience. The content is pitched at a level that is appropriate for research workers in software reliability, and for graduate-level courses in applied statistics, computer science, operations research, and software engineering. Industrial scientists looking for a better understanding of the ideas behind the statistical tools that they use for addressing problems of software quality may also find the material of value. We have deliberately steered away from

presenting techniques that are purely data analytic, since there are ample sources that do this.

Recognizing the diverse nature of our audience, and in keeping with our stated objectives, we have adopted an expository style and have striven to give as many illustrative examples as is possible; furthermore, we have endeavored to cast the examples in a context that may appeal to software engineers. Additional examples have been delegated to exercises. Readers who are formally trained in the statistical sciences will be familiar with the material of Chapter 2; they may find little that is new here, and may therefore be tempted to skip it. The same may also be true of the initial parts of Chapter 4. However, we urge such readers not to do so because of two reasons. The first is that the illustrative examples give a flavor of the nature of problems that we are trying to address. The second reason is that our interpretation of probability is personal (subjective); it therefore makes us look at the standard material in probability and statistics at a different angle. Of course, not all are willing to subscribe to this perspective. Computer scientists, operations research analysts, and software engineers should find the material of Chapter 2 and the initial parts of Chapter 4 as a useful review, but with a focus towards a specific application. The material in the other chapters is self-evident and so does not deserve special comment.

By way of a final admission, we are anticipating the criticism that any endeavor that attempts to fill a gap in the literature which is at the interface of computer science and statistics is necessarily incomplete. If this be so, then our hope is that the material here will stimulate the next generation of writers to expand the frontiers of the interface and to eliminate the pockets of incompleteness that we undoubtedly have created.

May 1999

Nozer D. Singpurwalla
Simon P. Wilson

Acknowledgments

Nozer D. Singpurwalla acknowledges the support of The George Washington University, and the sponsorship of The Army Research Office, The Air Force Office of Scientific Research, and The Office of Naval Research for initiating and sustaining his research on the subject of this book. Drs. Jagdish Chandra, Robert Launer, Charles Holland, Jon Sjogren, Seymour Selig, Julia Abrahams, and Professors Edward Wegman, Harold Liebowitz, William Marlow, Donald Gross, and Richard Soland are singled out with heartfelt thanks. He also acknowledges the contributions to his learning of his many coauthors; Professors Richard Barlow, Frank Proschan, Dennis Lindley, and Jayaram Sethuraman deserve special mention. For ensuring quality during the final stages of this work, the keen eye and the skill of Karen Brady, Yuling Cui, Selda Kapan, Seung Byeon, Andrew Swift, and Chung-Wai Kong are acknowledged, with Chung-Wai and Yuling deserving a very singular recognition. Mrs. Teresita Abacan's infinite efforts to convert handwritten notes into a masterful manuscript, often at supersonic speeds, is gratefully admired. Professors Ingram Olkin and Stephen Feinberg cast the initial bait, a long time ago, to suggest the writing of this book, and Dr. John Kimmel of Springer Verlag ensured that the fish did not slip away. Finally, he acknowledges the support of his family members, wife, children, mother, mother-in-law, and sister, each of whom in their own subtle way contributed to his work by not imposing any demands on him. Like the hidden parameters of probability models, they are the invisible coauthors.

Simon P. Wilson acknowledges his parents, for all their support and encouragement.

CONTENTS

Preface	v
Acknowledgments	vii
1 Introduction and Overview	1
1.1 What is Software Engineering?	1
1.2 Uncertainty in Software Production	2
1.2.1 The Software Development Process	2
1.2.2 Sources of Uncertainty in the Development Process	3
1.3 The Quantification of Uncertainty	4
1.3.1 Probability as an Approach for Quantifying Uncertainty	4
1.3.2 Interpretations of Probability	6
1.3.3 Interpreting Probabilities in Software Engineering	9
1.4 The Role of Statistical Methods in Software Engineering	9
1.5 Chapter Summary	11
2 Foundational Issues: Probability and Reliability	13
2.0 Preamble	13
2.1 The Calculus of Probability	14
2.1.1 Notation and Preliminaries	14
2.1.2 Conditional Probabilities and Conditional Independence	16
2.1.3 The Calculus of Probability	17
2.1.4 The Law of Total Probability, Bayes' Law, and the Likelihood Function	20
2.1.5 The Notion of Exchangeability	25
2.2 Probability Models and Their Parameters	28
2.2.1 What is a Software Reliability Model?	28
2.2.2 Some Commonly Used Probability Models	29
2.2.3 Moments of Probability Distributions and Expectation of Random Variables	39
2.2.4 Moments of Probability Models: The Mean Time to Failure	41

2.3 Point Processes and Counting Process Models	41
2.3.1 The Nonhomogeneous Poisson Process Model	43
2.3.2 The Homogeneous Poisson Process Model	45
2.3.3 Generalizations of the Point Process Model	46
2.4 Fundamentals of Reliability	52
2.4.1 The Notion of a Failure Rate Function	53
2.4.2 Some Commonly Used Model Failure Rates	54
2.4.3 Covariates in the Failure Rate Function	57
2.4.4 The Concatenated Failure Rate Function	58
2.5 Chapter Summary	59
Exercises for Chapter 2	61
 3 Models for Measuring Software Reliability	 67
3.1 Background: The Failure of Software	67
3.1.1 The Software Failure Process and Its Associated Randomness	68
3.1.2 Classification of Software Reliability Models	70
3.2 Models Based on the Concatenated Failure Rate Function	72
3.2.1 The Failure Rate of Software	72
3.2.2 The Model of Jelinski and Moranda (1972)	72
3.2.3 Extensions and Generalizations of the Model by Jelinski and Moranda	75
3.2.4 Hierarchical Bayesian Reliability Growth Models	76
3.3 Models Based on Failure Counts	77
3.3.1 Time Dependent Error Detection Models	77
3.4 Models Based on Times Between Failures	80
3.4.1 The Random Coefficient Autoregressive Process Model	80
3.4.2 A Non-Gaussian Kalman Filter Model	81
3.5 Unification of Software Reliability Models	82
3.5.1 Unification via the Bayesian Paradigm	83
3.5.2 Unification via Self-Exciting Point Process Models	84
3.5.3 Other Approaches to Unification	86
3.6 An Adaptive Concatenated Failure Rate Model	91
3.6.1 The Model and Its Motivation	92

3.6.2	Properties of the Model and Interpretation of Model Parameters	94
3.7	Chapter Summary	95
	Exercises for Chapter 3	97
4	Statistical Analysis of Software Failure Data	101
4.1	Background: The Role of Failure Data	101
4.2	Bayesian Inference, Predictive Distributions, and Maximization of Likelihood	103
4.2.1	Bayesian Inference and Prediction	104
4.2.2	The Method of Maximum Likelihood	105
4.2.3	Application: Inference and Prediction Using Jelinski and Moranda's Model	106
4.2.4	Application: Inference and Prediction Under an Error Detection Model	110
4.3	Specification of Prior Distributions	113
4.3.1	Standard of Reference—Noninformative Priors	114
4.3.2	Subjective Priors Based on Elicitation of Specialist Knowledge	115
4.3.3	Extensions of the Elicitation Model	117
4.3.4	Example: Eliciting Priors for the Logarithmic-Poisson Model	118
4.3.5	Application: Failure Prediction Using Logarithmic-Poisson Model	120
4.4	Inference and Prediction Using a Hierarchical Model	124
4.4.1	Application to NTDS Data: Assessing Reliability Growth	126
4.5	Inference and Predictions Using Dynamic Models	129
4.5.1	Inference for the Random Coefficient Exchangeable Model	131
4.5.2	Inference for the Adaptive Kalman Filter Model	141
4.5.3	Inference for the Non-Gaussian Kalman Filter Model	143
4.6	Prequential Prediction, Bayes Factors, and Model Comparison	145

4.6.1	Prequential Likelihoods and Prequential Prediction	146
4.6.2	Bayes' Factors and Model Averaging	148
4.6.3	Model Complexity—Occam's Razor	150
4.6.4	Application: Comparing the Exchangeable, Adaptive, and Non-Gaussian Models	151
4.6.5	An Example of Reversals in the Prequential Likelihood Ratio	153
4.7	Inference for the Concatenated Failure Rate Model	154
4.7.1	Specification of the Prior Distribution	155
4.7.2	Calculating Posteriors by Markov Chain Monte Carlo	157
4.7.3	Testing Hypotheses About Reliability Growth or Decay	159
4.7.4	Application to System 40 Data	160
4.8	Chapter Summary	164
	Exercises for Chapter 4	166
5	Software Productivity and Process Management	169
5.1	Background: Producing Quality Software	169
5.2	A Growth-Curve Model for Estimating Software Productivity	170
5.2.1	The Statistical Model	171
5.2.2	Inference and Prediction Under the Growth-Curve Model	174
5.2.3	Application: Estimating Individual Software Productivity	176
5.3	The Capability Maturity Model for Process Management	180
5.3.1	The Conceptual Framework	181
5.3.2	The Probabilistic Approach for Hierarchical Classification	183
5.3.3	Application: Classifying a Software Developer	186
5.4	Chapter Summary	188
	Exercises for Chapter 5	190

6	The Optimal Testing and Release of Software	191
6.1	Background: Decision Making and the Calculus of Probability	191
6.2	Decision Making Under Uncertainty	192
6.3	Utility and Choosing the Optimal Decision	194
6.3.1	Maximization of Expected Utility	194
6.3.2	The Utility of Money	195
6.4	Decision Trees	196
6.4.1	Solving Decision Trees	197
6.5	Software Testing Plans	198
6.6	Examples of Optimal Testing Plans	202
6.6.1	One-Stage Testing Using the Jelinski–Moranda Model	202
6.6.2	One-and Two-Stage Testing Using the Model by Goel and Okumoto	206
6.6.3	One-Stage Lookahead Testing Using the Model by Goel and Okumoto	211
6.6.4	Fixed-Time Lookahead Testing for the Goel–Okumoto Model	212
6.6.5	One-Bug Lookahead Testing Plans	214
6.6.6	Optimality of One-Stage Look Ahead Plans	215
6.7	Application: Testing the NTDS Data	216
6.8	Chapter Summary	217
	Exercises for Chapter 6	219
7	Other Developments: Open Problems	221
7.0	Preamble	221
7.1	Dynamic Modeling and the Operational Profile	222
7.1.1	Martingales, Predictable Processes, and Compensators: An Overview	222
7.1.2	The Doob–Meyer Decomposition of Counting Processes	224
7.1.3	Incorporating the Operational Profile	227
7.2	Statistical Aspects of Software Testing: Experimental Designs	228
7.2.1	Inferential Issues in Random and Partition Testing	229

7.2.2	Comparison of Random and Partition Testing	231
7.2.3	Design of Experiments in Software Testing	232
7.2.4	Design of Experiments in Multiversion Programming	236
7.2.5	Concluding Remarks	237
7.3	The Integration of Module and System Performance	238
7.3.1	The Protocols of Control Flow and Data Flow	239
7.3.2	The Structure Function of Modularized Software	242
Appendices		247
Appendix A	Statistical Computations Using the Gibbs Sampler	249
A.1	An Overview of the Gibbs Sampler	250
A.2	Generating Random Variates—The Rejection Method	253
A.3	Examples: Using the Gibbs Sampler	254
A.3.1	Gibbs Sampling the Jelinski–Moranda Model	254
A.3.2	Gibbs Sampling the Hierarchical Model	255
A.3.3	Gibbs Sampling the Adaptive Kalman Filter Model	256
A.3.4	Gibbs Sampling the Non-Gaussian Kalman Filter Model	258
Appendix B	The Maturity Questionnaire and Responses	261
B.1	The Maturity Questionnaire	261
B.2	Binary (Yes, No) Responses to the Maturity Questionnaire	265
B.3	Prior Probabilities and Likelihoods	266
B.3.1	The Maturity Levels $\mathcal{P}(M_i M_{i-1})$	266
B.3.2	The Key Process Areas $\mathcal{P}(K_{ij})$ and $\mathcal{P}(K_{ij} M_i)$	266
B.3.3	The Likelihoods $\mathcal{L}(K_{ij}; \mathbf{R}^{ij})$	268
References		269
Author Index		283
Subject Index		287