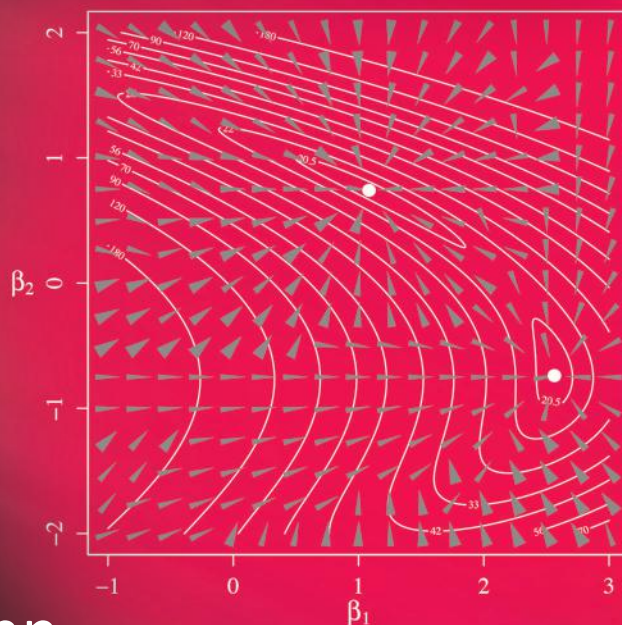


Computer Science and Data Analysis Series

# Foundations of Statistical Algorithms

With References to R Packages



Claus Weihs  
Olaf Mersmann  
Uwe Ligges

 CRC Press  
Taylor & Francis Group

A CHAPMAN & HALL BOOK



---

# Foundations of Statistical Algorithms

With References to R Packages

---

Chapman & Hall/CRC

## **Computer Science and Data Analysis Series**

The interface between the computer and statistical sciences is increasing, as each discipline seeks to harness the power and resources of the other. This series aims to foster the integration between the computer sciences and statistical, numerical, and probabilistic methods by publishing a broad range of reference works, textbooks, and handbooks.

### **SERIES EDITORS**

David Blei, Princeton University

David Madigan, Rutgers University

Marina Meila, University of Washington

Fionn Murtagh, Royal Holloway, University of London

Proposals for the series should be sent directly to one of the series editors above, or submitted to:

### **Chapman & Hall/CRC**

4th Floor, Albert House

1-4 Singer Street

London EC2A 4BQ

UK

---

### **Published Titles**

Semisupervised Learning for Computational Linguistics

*Steven Abney*

Design and Modeling for Computer Experiments

*Kai-Tai Fang, Runze Li, and Agus Sudjianto*

Microarray Image Analysis: An Algorithmic Approach

*Karl Fraser, Zidong Wang, and Xiaohui Liu*

R Programming for Bioinformatics

*Robert Gentleman*

Exploratory Multivariate Analysis by Example Using R

*François Husson, Sébastien Lê, and Jérôme Pagès*

Bayesian Artificial Intelligence, Second Edition

*Kevin B. Korb and Ann E. Nicholson*

Computational Statistics Handbook with MATLAB<sup>®</sup>, Second Edition

*Wendy L. Martinez and Angel R. Martinez*

---

**Published Titles cont.**

Exploratory Data Analysis with MATLAB<sup>®</sup>, Second Edition  
*Wendy L. Martinez, Angel R. Martinez, and Jeffrey L. Solka*

Clustering for Data Mining: A Data Recovery Approach, Second Edition  
*Boris Mirkin*

Introduction to Machine Learning and Bioinformatics  
*Sushmita Mitra, Sujay Datta, Theodore Perkins, and George Michailidis*

Introduction to Data Technologies  
*Paul Murrell*

R Graphics  
*Paul Murrell*

Correspondence Analysis and Data Coding with Java and R  
*Fionn Murtagh*

Pattern Recognition Algorithms for Data Mining  
*Sankar K. Pal and Pabitra Mitra*

Statistical Computing with R  
*Maria L. Rizzo*

Statistical Learning and Data Science  
*Mireille Gettler Summa, Léon Bottou, Bernard Goldfarb, Fionn Murtagh,  
Catherine Pardoux, and Myriam Touati*

Foundations of Statistical Algorithms: With References to R Packages  
*Claus Weihs, Olaf Mersmann, and Uwe Ligges*



# Foundations of Statistical Algorithms

With References to R Packages

Claus Weihs  
Olaf Mersmann  
Uwe Ligges

TU Dortmund University  
Germany



CRC Press

Taylor & Francis Group  
Boca Raton London New York

---

CRC Press is an imprint of the  
Taylor & Francis Group, an **informa** business  
A CHAPMAN & HALL BOOK

---

CRC Press  
Taylor & Francis Group  
6000 Broken Sound Parkway NW, Suite 300  
Boca Raton, FL 33487-2742

© 2014 by Taylor & Francis Group, LLC  
CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works  
Version Date: 20131104

International Standard Book Number-13: 978-1-4398-7887-3 (eBook - PDF)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access [www.copyright.com](http://www.copyright.com) (<http://www.copyright.com/>) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

**Trademark Notice:** Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation without intent to infringe.

**Visit the Taylor & Francis Web site at**  
**<http://www.taylorandfrancis.com>**

**and the CRC Press Web site at**  
**<http://www.crcpress.com>**



---

*To  
Heidrun and Max,  
Sabine,  
and Sandra*



---

# Contents

---

<b>Preface</b>	<b>xvii</b>
<b>The Authors</b>	<b>xix</b>
<b>List of Algorithms</b>	<b>xxi</b>
<b>Notation</b>	<b>xxiii</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Computation</b>	<b>9</b>
2.1 Motivation and History	9
2.1.1 Preliminaries	11
2.2 Models for Computing: What Can a Computer Compute?	11
2.2.1 Algorithms	11
2.2.1.1 Motivation and History: Models for Computing	11
2.2.1.2 Theory: Complexity of Algorithms	17
2.2.1.3 Example: Sorting Algorithms	22
2.2.1.4 Practice and Simulation	34
2.2.2 Turing Machines	36
2.2.2.1 Motivation and History: Turing machines	36
2.2.2.2 Theory: Definition	37
2.2.2.3 Example: Addition	42
2.2.2.4 Practice and Simulation	43
2.3 Floating-Point Computations: How Does a Computer Compute?	45
2.3.1 Motivation and History: Floating-Point Computations	45
2.3.2 Theory: Floating-Point Operations	49
2.3.2.1 Floating-Point Addition	49
2.3.2.2 Floating-Point Multiplication	52

2.3.3	Summary and Outlook	55
2.4	Precision of Computations: How Exact Does a Computer Compute?	56
2.4.1	Motivation and History: Rounding Errors in Floating-Point Operations	56
2.4.2	Theory: Error Propagation	59
2.4.3	Theory: Axiomatic Problems	62
2.4.4	Example: Calculation of the Empirical Variance	67
2.4.4.1	Algorithms	67
2.4.4.2	Condition and Error Bounds	72
2.4.4.3	Shifted Data	74
2.4.4.4	An Overall Approach	78
2.4.5	Practice and Simulation	79
2.4.6	Summary and Outlook	81
2.5	Implementation in R	81
2.5.1	Sorting	81
2.5.2	Floating-Point Numbers	81
2.5.3	(Co-)Variance	82
2.6	Conclusion	82
2.7	Exercises	83
<b>3</b>	<b>Verification</b>	<b>87</b>
3.1	Motivation and History	87
3.1.1	Preliminaries	88
3.1.2	The Condition of the Linear Least Squares Problem	91
3.1.3	Summary	97
3.2	Theory	98
3.2.1	LLS Solvers	98
3.2.1.1	Gram-Schmidt Method	98
3.2.1.2	Greville's Method	104
3.2.1.3	Summary	105
3.2.2	Test Data Generation	106
3.2.2.1	Nonsingular Test Matrices	106
3.2.2.2	Singular Test Matrices	109
3.2.2.3	Response Vectors	121
3.2.2.4	Summary	123
3.3	Practice and Simulation	123
3.3.1	Verification Procedure	123
3.3.2	Verification Results	125
3.3.3	Summary	131

3.4	Implementation in R	131
3.5	Conclusion	132
3.6	Exercises	132
<b>4</b>	<b>Iteration</b>	<b>137</b>
4.1	Motivation	138
4.2	Preliminaries	138
4.3	Univariate Optimization	141
4.3.1	Golden Section Search	146
4.3.2	Convergence	147
4.3.3	Faster Methods	149
4.3.4	Newton's Method	152
4.3.5	Summary	159
4.4	Multivariate Optimization	159
4.4.1	Convex Problems	159
4.4.1.1	General Structure of Iterative Algorithms	160
4.4.1.2	Coordinate Descent	162
4.4.1.3	Gradient Descent	164
4.4.1.4	Newton Algorithm	168
4.4.1.5	Quasi-Newton Methods	169
4.4.1.6	Convex Functions: Summary	175
4.4.1.7	Example: Logistic Regression	176
4.4.2	Nonconvex Problems	183
4.4.2.1	Nonlinear Least Squares (NLS) Problems	183
4.4.2.2	Example: Comparing the Algorithms	187
4.4.3	Derivative-Free Methods	193
4.4.4	Summary	200
4.5	Example: Neural Nets	200
4.5.1	Concept	201
4.5.2	Coefficient Estimation with Neural Nets	204
4.5.3	Identifiability of Neural Nets	207
4.5.4	Summary	211
4.6	Constrained Optimization	211
4.6.1	Equality-Constrained Optimization	212
4.6.2	Inequality-Constrained Optimization	214
4.6.2.1	Applications	216
4.6.3	Linear Programming	217
4.6.3.1	Simulation	221
4.6.4	Quadratic Programming	223
4.6.5	Summary	226

4.7	Evolutionary Computing	226
4.7.1	Simulated Annealing	229
4.7.2	Evolutionary Strategies	231
4.7.2.1	Desirabilities Example	233
4.7.2.2	Maximum-Likelihood Example	236
4.7.3	Summary	238
4.8	Implementation in R	238
4.9	Conclusion	239
4.10	Exercises	240
<b>5</b>	<b>Deduction of Theoretical Properties</b>	<b>243</b>
5.1	PLS – from Algorithm to Optimality	243
5.1.1	Motivation and History	243
5.1.2	NIPALS Algorithm	244
5.1.3	Covariance Optimality	248
5.1.4	PLS Method: Examples	250
5.1.5	PLS Method: Summary and Outlook	252
5.2	EM Algorithm	252
5.2.1	Motivation and Introduction	252
5.2.2	Definition and Convergence	253
5.2.3	Example: Mixture of Gaussians	256
5.2.4	Example: $k$ -means Clustering	261
5.2.5	EM Algorithm: Summary and Outlook	263
5.2.5.1	Summary	263
5.3	Implementation in R	265
5.3.1	PLS	265
5.3.2	EM Algorithm	265
5.4	Conclusion	266
5.5	Exercises	266
<b>6</b>	<b>Randomization</b>	<b>269</b>
6.1	Motivation and History	269
6.1.1	Preliminaries	269
6.2	Theory: Univariate Randomization	270
6.2.1	Introduction	270
6.2.2	Uniform Distribution	271
6.2.2.1	Multiply-with-Carry Generators	277
6.2.2.2	Overview of Other Generators	279
6.2.2.3	Empirical Tests on Randomness	280
6.2.3	Test Suites for Random Number Generators	284

6.2.3.1	Unrecommended Generator	284
6.2.3.2	Recommended Generators	285
6.2.4	Other Distributions	286
6.2.4.1	Bernoulli Distribution	286
6.2.4.2	Binomial Distribution	287
6.2.4.3	Hypergeometrical Distribution	288
6.2.4.4	Poisson Distribution	288
6.2.4.5	Waiting Time Distributions	289
6.2.5	Continuous Distributions	291
6.2.5.1	Continuous Uniform Distribution	291
6.2.5.2	Triangular Distribution	293
6.2.5.3	Normal Distribution	295
6.2.5.4	Mixture of Distributions	297
6.2.5.5	Exponential Distribution	299
6.2.5.6	Lognormal Distribution	300
6.2.6	Summary	301
6.3	Theory: Multivariate Randomization	302
6.3.1	Motivation and Overview	302
6.3.2	Rejection Method	304
6.3.3	Gibbs Algorithm	308
6.3.3.1	Finite Case	309
6.3.3.2	Continuous Case	312
6.3.3.3	When Does Gibbs Sampling Converge?	314
6.3.4	Metropolis-Hastings Method	316
6.3.4.1	Finite Case	316
6.3.4.2	Continuous Case	318
6.3.4.3	Possible Transition Functions	319
6.3.4.4	Implementation of the Metropolis-Hastings Algorithm	320
6.3.4.5	Convergence Rate	321
6.3.4.6	Rejection Method and Independence Chain MCMC	324
6.3.5	Summary	325
6.3.6	Simple Example	325
6.4	Practice and Simulation: Stochastic Modeling	329
6.4.1	Polyphonic Music Data	329
6.4.2	Modeling	329
6.4.3	Model Fit by Means of MCMC Methods	334
6.4.4	BUGS Implementation	346
6.4.5	Summary	349

6.5	Implementation in R	350
6.6	Conclusion	350
6.7	Exercises	351
<b>7</b>	<b>Repetition</b>	<b>355</b>
7.1	Motivation and Overview	355
7.1.1	Preliminaries	357
7.2	Model Selection	359
7.2.1	General Procedure	359
7.2.2	Model Selection in Supervised Learning	361
7.2.2.1	Resampling in Simulation	363
7.2.2.2	Resampling for Competition Data	364
7.2.2.3	Resampling in Practice	365
7.2.3	Resampling Methods	365
7.2.3.1	Cross-Validation	366
7.2.3.2	Bootstrap	367
7.2.3.3	Subsampling	371
7.2.3.4	Further Resampling Methods	371
7.2.3.5	Properties and Recommendations	372
7.2.4	Feature Selection	374
7.2.5	Hyperparameter Tuning	375
7.2.6	Summary	376
7.3	Model Selection in Classification	377
7.3.1	The Classification Problem	377
7.3.2	Classification Evaluation	379
7.3.3	Simulation: Comparison of Error Rate Estimators	382
7.3.4	Example: Discrimination of Piano and Guitar	386
7.3.4.1	Groups of Features	386
7.3.4.2	Feature Selection Results	391
7.3.4.3	Hyperparameter Tuning Results	393
7.3.5	Summary	394
7.4	Model Selection in Continuous Models	394
7.4.1	Theory: Nonlinear Prediction	396
7.4.2	Model Selection in Neural Nets: Size of Hidden Layer	401
7.4.3	Model Selection for PLS: Selection of Latent Variables	403
7.4.4	Summary	405
7.5	Implementation in R	406
7.6	Conclusion	406



---

CONTENTS	xv
7.7 Exercises	407
<b>8 Scalability and Parallelization</b>	<b>411</b>
8.1 Introduction	411
8.2 Motivation and History	413
8.2.1 Motivation	413
8.2.2 Early Years	415
8.2.2.1 Hardware	415
8.2.2.2 Software	417
8.2.3 Recent History	420
8.2.4 Summary	422
8.3 Optimization	422
8.3.1 Examples in R	422
8.3.2 Guidelines	425
8.3.3 Summary	426
8.4 Parallel Computing	427
8.4.1 Potential	427
8.4.2 Setting	430
8.4.3 Data Parallel Computing	430
8.4.3.1 Example: <i>k</i> -means Clustering	431
8.4.3.2 Summary and Outlook	438
8.4.4 Task Parallel Computing	438
8.4.5 Modern Paradigms	440
8.4.6 Summary	442
8.5 Implementation in R	443
8.6 Conclusion	443
8.7 Exercises	444
<b>Bibliography</b>	<b>449</b>



---

# Preface

---

This book is largely based on a yearly lecture, “Computer Supported Statistics” (Computergestützte Statistik), for statistics students regularly held and improved by the authors since winter 1999/2000 at the TU University Dortmund (Germany). The exercises are based on those prepared for this lecture and other related lectures. For the book, the material of this lecture was thoroughly revised, extended, and modernized. This is particularly true for Chapters 4 and 8.

This book is not “yet another treatise on computational statistics”. In fact, there is, as of this writing, no other book on the market that has a similar emphasis, for at least three reasons.

1. All the textbooks on computational statistics we know of present concise introductions to a multitude of state-of-the-art statistical algorithms without covering the historical aspect of their development, which we think is instructive in understanding the evolution of ever more powerful statistical algorithms. Many of the older algorithms are still building blocks or inspiration for current techniques. It is therefore instructive to cover these as well and present the material from a historical perspective before explaining the current best-of-breed algorithms, which naturally makes up the main body of the book.
2. With the chosen chapter titles, we try to emphasize certain recurring themes in all statistical algorithms: Computation, assessment and verification, iteration, deduction of theoretical properties, randomization, repetition and parallelization and scalability. Students should not only understand current algorithms after reading this book, but also gain a deeper understanding of how algorithms are constructed, how to evaluate new algorithms, which recurring principles are used to tackle some of the tough problems statistical programmers face, and how to take an idea for a new method and turn it into something practically useful.
3. The book contains two chapters on topics neglected in other books. One chapter is dedicated to systematic verification, a topic that is not covered in any other statistical computing book we know of. Instead of focusing on

contrived test examples, we show how to derive general classes of worst case inputs and why it is important to systematically test an algorithm over a large number of different inputs. And another chapter covers the upcoming challenge of scaling many of the established techniques to very large data sets and how the availability of many CPU cores will change the way we think about statistical computing.

To summarize, this book is based on a new and refreshingly different approach to presenting the foundations of statistical algorithms. Therefore, this book provides a great resource for both students and lecturers teaching a course in computational statistics.

#### *Acknowledgments*

We thank Daniel Horn, Sarah Schnackenberg, and Sebastian Szugat for their tireless critical proof reading, Pascal Kerschke for investigating historical literature, John Kimmel for his powerful realization of the review process, the unknown reviewers and the copy-editor for their valuable comments on draft copies of the manuscript, and last but not least Marcus Fontaine for L<sup>A</sup>T<sub>E</sub>X support in preparing the final manuscript.

---

## The Authors

---

**Prof. Dr. Claus Weihs** studied mathematics in Bonn (Germany). After his studies, he developed a large software system at the economics department in Bonn (Germany). He received his PhD (Dr. rer. nat.) in numerical mathematics from the University of Trier (Germany) in 1986. He then practiced statistics and mathematics for 9 years as a consultant for Ciba-Geigy (Basel, Switzerland). He took on his current position as the chair of Computational Statistics in Dortmund (Germany) in 1995. So far, among more than 150 publications, Prof. Weihs has published two monographs and edited four other books. Moreover, he has served as president of the German Classification Society (GfKI) from 2004 to 2013, and he is one of the editors of the journal *Advances in Data Analysis and Classification (ADAC)*.

**Olaf Mersmann** has studied physics and statistics at the TU Ilmenau, University of Marburg, and the TU Dortmund University. He finished his MA in data analysis in 2011 at the department of statistics of the TU Dortmund University. Starting with his BA, Mr. Mersmann has been researching new and innovative ways to objectively test benchmark computer algorithms. He has contributed eight packages to CRAN, the R software package repository, and worked on several more.

**Dr. Uwe Ligges** is junior-professor for Data Analysis and Statistical Algorithms at the department of statistics, TU Dortmund University. He is author of the (German) textbook *Programmieren mit R (Programming in R)* (Springer Verlag, Heidelberg), which was first published in 2004 and is currently available in its third edition. A Japanese translation of this book was published in 2006. Uwe Ligges is also known as a member of the R Core Team and the CRAN maintainer for Windows binaries of contributed packages. Additionally, he acts as one of the editors for the *Journal of Statistical Software* as well as a column editor for *The R Journal*.



---

# List of Algorithms

---

2.1	Bubble Sort	23
2.2	Insertion Sort	25
2.3	Quick Sort	28
2.4	Addition of Two Floating-Point Numbers	49
2.5	Normalization of a Floating-Point Number	52
2.6	Multiplication of Two Floating-Point Numbers	53
2.7	Division of Two Floating-Point Numbers	53
2.8	Matrix Exponentiation	84
3.1	Gram-Schmidt Orthogonalization (GS1)	100
3.2	Gram-Schmidt Orthogonalization (GS2)	101
3.3	Modified Gram-Schmidt Orthogonalization (MGS)	102
3.4	Pivot Strategy (PS)	103
3.5	Generation of Response Vectors (RVs)	122
3.6	Givens Orthogonalization	134
4.1	Generic Univariate Section Search	143
4.2	Golden Section Search	148
4.3	Parabolic Interpolation Search	151
4.4	Univariate Newton Method	153
4.5	General Iterative Algorithm	160
4.6	Steepest Descent Method	164
4.7	Multivariate Newton Method	169
4.8	BFGS Quasi-Newton Method	174
4.9	Nelder-Mead Method	198
4.10	Sequential Unconstrained Minimization	217
4.11	General Search Distribution Based Optimization Scheme	228
4.12	Simulated Annealing	231
4.13	Evolutionary Strategy	233
5.1	NIPALS (Nonlinear Iterative Partial Least Squares) Algorithm	247
5.2	Iteration Step of the EM Algorithm (General Case)	253
5.3	$(k + 1)$ th Iteration Step of the EM Algorithm (General Case)	253

5.4	$(k + 1)$ h Iteration Step of the EM Algorithm (Exponential Family)	254
5.5	$(k + 1)$ th Iteration Step of the EM Algorithm (Mixture of Two Gaussians $Y_1$ and $Y_2$ )	258
5.6	EM Algorithm for Mixtures of two Gaussians $Y_1$ and $Y_2$	258
5.7	Expectation Step of $k$ -means ( <i>kmeans_e_step</i> )	261
5.8	Maximization Step of $k$ -means ( <i>kmeans_m_step</i> )	262
5.9	$k$ -means Clustering	263
6.1	Modulo-Operation for an LCS with $m = 2^{31}$	274
6.2	Calculation of Next Element in a Multiply-with-Carry Sequence	279
6.3	General Method for the Generation of Random Realizations $x$ of Discrete Distributions on $x = 0, 1, 2, \dots$	290
6.4	Inversion Method: Realizations of a Continuous, Strictly Monotonic Distribution	295
6.5	Rejection Method: Generation of $N$ Random Points from Density $\frac{\pi(\mathbf{x})}{\int_B \pi(\mathbf{x}) d\mathbf{x}}$	305
6.6	Gibbs Algorithm: One Iteration	309
6.7	Metropolis-Hastings Algorithm: One Iteration	318
7.1	Generic Resampling	366
7.2	Subsets for $B$ -Fold CV	367
7.3	Subsets for the Bootstrap	368
7.4	.632+ Bootstrap	370
7.5	Subsets for Subsampling	371
7.6	Greedy Forward Selection	375
7.7	Construction of Optimal Neural Net	403
8.1	Distance Matrix Calculation	429
8.2	Sequential $k$ -means Algorithm	431
8.3	Assign Cluster to an Observation in the Sequential $k$ -means Algorithm	433
8.4	Assign Cluster to an Observation in the Sequential $k$ -means Algorithm (Version 2)	434
8.5	Parallel $k$ -means Algorithm	437



---

sample content of Foundations of Statistical Algorithms: With References to R Packages (Chapman & Hall/CRC Computer Science & Data Analysis)

- [click Murder List: A Novel pdf, azw \(kindle\)](#)
- [read online Biggles - Secret Agent](#)
- [download online The Boston Girl](#)
- [Nine Faces Of Kenya pdf](#)
- [click Limulus in the Limelight: A Species 350 Million Years in the Making and in Peril? pdf, azw \(kindle\), epub](#)
  
- <http://weddingcellist.com/lib/Murder-List--A-Novel.pdf>
- <http://flog.co.id/library/Biggles---Secret-Agent.pdf>
- <http://paulczajak.com/?library/Introduction-to-Econometrics--2nd-Edition---Addison-Wesley-Series-in-Economics-.pdf>
- <http://anvilpr.com/library/Nine-Faces-Of-Kenya.pdf>
- <http://www.1973vision.com/?library/The-Adventures-of-Sir-Balin-the-III-Fated--The-Knights--Tales--Book-4-.pdf>