

Data Science and Predictive Analytics

Ivo D. Dinov

Data Science and Predictive Analytics

Biomedical and Health Applications using R

Ivo D. Dinov
University of Michigan–Ann Arbor
Ann Arbor, Michigan, USA

Additional material to this book can be downloaded from <http://extras.springer.com>.

ISBN 978-3-319-72346-4 ISBN 978-3-319-72347-1 (eBook)
<https://doi.org/10.1007/978-3-319-72347-1>

Library of Congress Control Number: 2018930887

© Ivo D. Dinov 2018

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG part of Springer Nature

The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

*. . . dedicated to my lovely and encouraging
wife, Magdalena, my witty and persuasive
kids, Anna-Sophia and Radina, my very
insightful brother, Konstantin, and my
nurturing parents, Yordanka and Dimitar . . .*

Foreword

Instructors, formal and informal learners, working professionals, and readers looking to enhance, update, or refresh their interactive data skills and methodological developments may selectively choose sections, chapters, and examples they want to cover in more depth. Everyone who expects to gain new knowledge or acquire computational abilities should review the overall textbook organization before they decide what to cover, how deeply, and in what order. The organization of the chapters in this book reflects an order that may appeal to many, albeit not all, readers.

Chapter 1 (Motivation) presents (1) the DSPA mission and objectives, (2) several driving biomedical challenges including Alzheimer’s disease, Parkinson’s disease, drug and substance use, and amyotrophic lateral sclerosis, (3) provides demonstrations of brain visualization, neurodegeneration, and genomics computing, (4) identifies the six defining characteristics of big (biomedical and healthcare) data, (5) explains the concepts of *data science* and *predictive analytics*, and (6) sets the DSPA expectations.

Chapter 2 (Foundations of R) justifies the use of the statistical programming language *R* and (1) presents the fundamental programming principles; (2) illustrates basic examples of data transformation, generation, ingestion, and export; (3) shows the main mathematical operators; and (4) presents basic data and probability distribution summaries and visualization.

In **Chap. 3 (Managing Data in R)**, we present additional *R* programming details about (1) loading, manipulating, visualizing, and saving *R* Data Structures; (2) present sample-based statistics measuring central tendency and dispersion; (3) explore different types of variables; (4) illustrate scrapping data from public websites; and (5) show examples of cohort-rebalancing.

A detailed discussion of **Visualization** is presented in **Chap. 4** where we (1) show graphical techniques for exposing composition, comparison, and relationships in multivariate data; and (2) present 1D, 2D, 3D, and 4D distributions along with surface plots.

The foundations of **Linear Algebra and Matrix Computing** are shown in **Chap. 5**. We (1) show how to create, interpret, process, and manipulate

second-order tensors (matrices); (2) illustrate variety of matrix operations and their interpretations; (3) demonstrate linear modeling and solutions of matrix equations; and (4) discuss the eigen-spectra of matrices.

Chapter 6 (Dimensionality Reduction) starts with a simple example reducing 2D data to 1D signal. We also discuss (1) matrix rotations, (2) principal component analysis (PCA), (3) singular value decomposition (SVD), (4) independent component analysis (ICA), and (5) factor analysis (FA).

The discussion of machine learning model-based and model-free techniques commences in **Chap. 7 (Lazy Learning – Classification Using Nearest Neighbors)**. In the scope of the k-nearest neighbor algorithm, we present (1) the general concept of divide-and-conquer for splitting the data into training and validation sets, (2) evaluation of model performance, and (3) improving prediction results.

Chapter 8 (Probabilistic Learning: Classification Using Naive Bayes) presents the naive Bayes and linear discriminant analysis classification algorithms, identifies the assumptions of each method, presents the Laplace estimator, and demonstrates step by step the complete protocol for training, testing, validating, and improving the classification results.

Chapter 9 (Decision Tree Divide and Conquer Classification) focuses on decision trees and (1) presents various classification metrics (e.g., entropy, misclassification error, Gini index), (2) illustrates the use of the C5.0 decision tree algorithm, and (3) shows strategies for pruning decision trees.

The use of linear prediction models is highlighted in **Chap. 10 (Forecasting Numeric Data Using Regression Models)**. Here, we present (1) the fundamentals of multivariate linear modeling, (2) contrast regression trees vs. model trees, and (3) present several complete end-to-end predictive analytics examples.

Chapter 11 (Black Box Machine-Learning Methods: Neural Networks and Support Vector Machines) lays out the foundation of Neural Networks as silicon analogues to biological neurons. We discuss (1) the effects of network layers and topology on the resulting classification, (2) present support vector machines (SVM), and (3) demonstrate classification methods for optical character recognition (OCR), iris flowers clustering, Google trends and the stock market prediction, and quantifying quality of life in chronic disease.

Apriori Association Rules Learning is presented in **Chap. 12** where we discuss (1) the foundation of association rules and the Apriori algorithm, (2) support and confidence measures, and (3) present several examples based on grocery shopping and head and neck cancer treatment.

Chapter 13 (k-Means Clustering) presents (1) the basics of machine learning clustering tasks, (2) silhouette plots, (3) strategies for model tuning and improvement, (4) hierarchical clustering, and (5) Gaussian mixture modeling.

General protocols for measuring the performance of different types of classification methods are presented in **Chap. 14 (Model Performance Assessment)**. We discuss (1) evaluation strategies for binary, categorical, and continuous outcomes; (2) confusion matrices quantifying classification and prediction accuracy; (3) visualization of algorithm performance and ROC curves; and (4) introduce the foundations of internal statistical validation.

Chapter 15 (Improving Model Performance) demonstrates (1) strategies for manual and automated model tuning, (2) improving model performance with meta-learning, and (3) ensemble methods based on bagging, boosting, random forest, and adaptive boosting.

Chapter 16 (Specialized Machine Learning Topics) presents some technical details that may be useful for some computational scientists and engineers. There, we discuss (1) data format conversion; (2) SQL data queries; (3) reading and writing XML, JSON, XLSX, and other data formats; (4) visualization of network bioinformatics data; (4) data streaming and on-the-fly stream classification and clustering; (5) optimization and improvement of computational performance; and (6) parallel computing.

The classical approaches for feature selection are presented in **Chap. 17 (Variable/Feature Selection)** where we discuss (1) filtering, wrapper, and embedded techniques, and (2) show the entire protocols from data collection and preparation to model training, testing, evaluation and comparison using recursive feature elimination.

In **Chap. 18 (Regularized Linear Modeling and Controlled Variable Selection)**, we extend the mathematical foundation we presented in Chap. 5 to include *fidelity* and *regularization* terms in the objective function used for model-based inference. Specifically, we discuss (1) computational protocols for handling complex high-dimensional data, (2) model estimation by controlling the false-positive rate of selection of critical features, and (3) derivations of effective forecasting models.

Chapter 19 (BigBig Longitudinal Data Analysis) is focused on interrogating time-varying observations. We illustrate (1) time series analysis, e.g., ARIMA modeling, (2) structural equation modeling (SEM) with latent variables, (3) longitudinal data analysis using linear mixed models, and (4) the generalized estimating equations (GEE) modeling.

Expanding upon the term-frequency and inverse document frequency techniques we saw in Chap. 8, **Chap. 20 (Natural Language Processing/Text Mining)** provides more details about (1) handling unstructured text documents, (2) term frequency (TF) and inverse document frequency (IDF), and (3) the cosine similarity measure.

Chapter 21 (Prediction and Internal Statistical Cross Validation) provides a broader and deeper discussion of method validation, which started in Chap. 14. Here, we present (1) general prediction and forecasting methods, (2) demonstrate internal statistical n-fold cross-validation, and (3) comparison strategies for multiple prediction models.

Chapter 22 (Function Optimization) presents technical details about minimizing objective functions, which are present virtually in any data science oriented inference or evidence-based translational study. Here, we explain (1) constrained and unconstrained cost function optimization, (2) Lagrange multipliers, (3) linear and quadratic programming, (4) general nonlinear optimization, and (5) data denoising.

The last chapter of this textbook is **Chap. 23 (Deep Learning)**. It covers (1) perceptron activation functions, (2) relations between artificial and biological

neurons and networks, (3) neural nets for computing exclusive OR (XOR) and negative AND (NAND) operators, (3) classification of handwritten digits, and (4) classification of natural images.

We compiled a few dozens of biomedical and healthcare case-studies that are used to demonstrate the presented DSPA concepts, apply the methods, and validate the software tools. For example, **Chap. 1** includes high-level driving biomedical challenges including dementia and other neurodegenerative diseases, substance use, neuroimaging, and forensic genetics. **Chapter 3** includes a traumatic brain injury (TBI) case-study, **Chap. 10** described a heart attacks case-study, and **Chap. 11** uses a quality of life in chronic disease data to demonstrate optical character recognition that can be applied to automatic reading of handwritten physician notes. **Chapter 18** presents a predictive analytics Parkinson's disease study using neuroimaging-genetics data. **Chapter 20** illustrates the applications of natural language processing to extract quantitative biomarkers from unstructured text, which can be used to study hospital admissions, medical claims, or patient satisfaction. **Chapter 23** shows examples of predicting clinical outcomes for amyotrophic lateral sclerosis and irritable bowel syndrome cohorts, as well as quantitative and qualitative classification of biological images and volumes. Indeed, these represent just a few examples, and the readers are encouraged to try the same methods, protocols and analytics on other research-derived, clinically acquired, aggregated, secondary-use, or simulated datasets.

The online appendices (<http://DSPA.predictive.space>) are continuously expanded to provide more details, additional content, and expand the DSPA methods and applications scope. Throughout this textbook, there are cross-references to appropriate chapters, sections, datasets, web services, and live demonstrations (Live Demos). The sequential arrangement of the chapters provides a suggested reading order; however, alternative sorting and pathways covering parts of the materials are also provided. Of course, readers and instructors may further choose their own coverage paths based on specific intellectual interests and project needs.

Preface

Genesis

Since the turn of the twenty-first century, the evidence overwhelming reveals that the rate of increase for the amount of data we collect doubles each 12–14 months (Kryder’s law). The growth momentum of the volume and complexity of digital information we gather far outpaces the corresponding increase of computational power, which doubles each 18 months (Moore’s law). There is a substantial imbalance between the increase of data inflow and the corresponding computational infrastructure intended to process that data. This calls into question our ability to extract valuable information and actionable knowledge from the mountains of digital information we collect. Nowadays, it is very common for researchers to work with petabytes (PB) of data, $1PB = 10^{15}$ bytes, which may include nonhomologous records that demand unconventional analytics. For comparison, the Milky Way Galaxy has approximately 2×10^{11} stars. If each star represents a byte, then one petabyte of data correspond to 5,000 Milky Way Galaxies.

This data storage-computing asymmetry leads to an explosion of innovative data science methods and disruptive computational technologies that show promise to provide effective (semi-intelligent) decision support systems. Designing, understanding and validating such new techniques require deep within-discipline basic science knowledge, transdisciplinary team-based scientific collaboration, open-scientific endeavors, and a blend of exploratory and confirmatory scientific discovery. There is a pressing demand to bridge the widening gaps between the needs and skills of practicing data scientists, advanced techniques introduced by theoreticians, algorithms invented by computational scientists, models constructed by biosocial investigators, network products and Internet of Things (IoT) services engineered by software architects.

Purpose

The purpose of this book is to provide a sufficient methodological foundation for a number of modern data science techniques along with hands-on demonstration of implementation protocols, pragmatic mechanics of protocol execution, and interpretation of the results of these methods applied on concrete case-studies. Successfully completing the Data Science and Predictive Analytics (DSPA) training materials (<http://predictive.space>) will equip readers to (1) understand the computational foundations of Big Data Science; (2) build critical inferential thinking; (3) lend a tool chest of *R* libraries for managing and interrogating raw, derived, observed, experimental, and simulated big healthcare datasets; and (4) furnish practical skills for handling complex datasets.

Limitations/Prerequisites

Prior to diving into DSPA, the readers are strongly encouraged to review the prerequisites and complete the self-assessment pretest. Sufficient remediation materials are provided or referenced throughout. The DSPA materials may be used for variety of graduate level courses with durations of 10–30 weeks, with 3–4 instructional credit hours per week. Instructors can refactor and present the materials in alternative orders. The DSPA chapters in this book are organized sequentially. However, the content can be tailored to fit the audience’s needs. Learning data science and predictive analytics is not a linear process – many alternative pathways

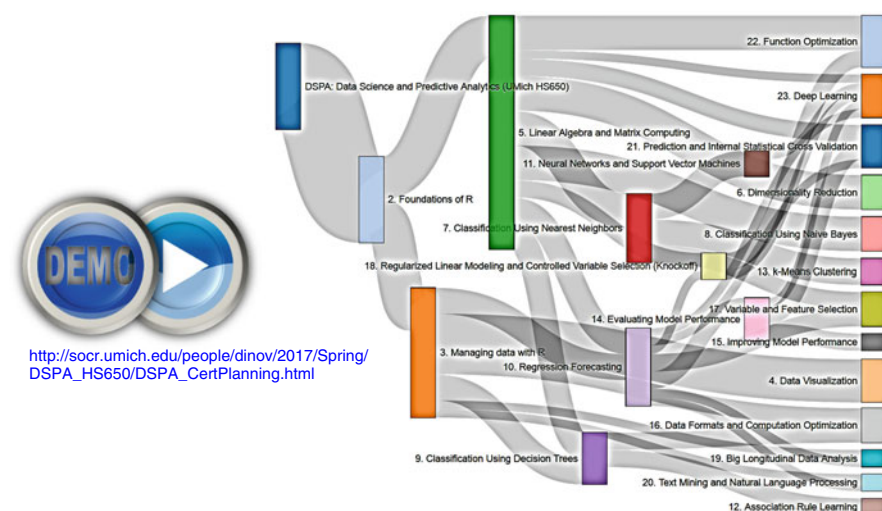


Fig. 1 DSPA topics flowchart

can be completed to gain complementary competencies. We developed an interactive and dynamic flowchart (http://socr.umich.edu/people/dinov/courses/DSPA_Book_FlowChart.html) that highlights several tracks illustrating reasonable pathways starting with Foundations of *R* and ending with specific competency topics. The content of this book may also be used for self-paced learning or as a refresher for working professionals, as well as for formal and informal data science training, including massive open online courses (MOOCs). The DSPA materials are designed to build specific data science skills and predictive analytic competencies, as described by the Michigan Institute for Data Science (MIDAS).

Scope of the Book

Throughout this book, we use a constructive definition of “Big Data” derived by examining the common characteristics of many dozens of biomedical and healthcare case-studies, involving complex datasets that required special handling, advanced processing, contemporary analytics, interactive visualization tools, and translational interpretation. These six characteristics of “Big Data” are defined in the Motivation Chapter as size, heterogeneity and complexity, representation incongruency, incompleteness, multiscale format, and multisource origins. All supporting electronic materials, including datasets, assessment problems, code, software tools, videos, and appendices, are available online at <http://DSPA.predictive.space>.

This textbook presents a balanced view of the mathematical formulation, computational implementation, and health applications of modern techniques for managing, processing, and interrogating big data. The intentional focus on human health applications is demonstrated by a diverse range of biomedical and healthcare case-studies. However, the same techniques could be applied in other domains, e.g., climate and environmental sciences, biosocial sciences, high-energy physics, astronomy, etc., that deal with complex data possessing the above characteristics. Another specific feature of this book is that it solely utilizes the statistical computing language *R*, rather than any other scripting, user-interface based, or software programming alternatives. The choice for *R* is justified in the Foundations Chapter.

All techniques presented here aim to obtain data-driven and evidence-based scientific inference. This process starts with collecting or retrieving an appropriate dataset, and identifying sources of data that need to be harmonized and aggregated into a joint computable data object. Next, the data are typically split into training and testing components. Model-based or model-free methods are fit, estimated, or learned on the training component and then validated on the complementary testing data. Different types of expected outcomes and results from this process include prediction, prognostication, or forecasting of specific clinical traits (computable phenotypes), clustering, or classification that labels units, subjects, or cases in the data. The final steps include algorithm fine-tuning, assessment, comparison, and statistical validation.

Acknowledgements

The work presented in this textbook relies on deep basic science, as well as holistic interdisciplinary connections developed by scholars, teams of scientists, and trans-disciplinary collaborations. Ideas, datasets, software, algorithms, and methods introduced by the wider scientific community were utilized throughout the DSPA resources. Specifically, methodological and algorithmic contributions from the fields of computer vision, statistical learning, mathematical optimization, scientific inference, biomedical computing, and informatics drove the concept presentations, data-driven demonstrations, and case-study reports. The enormous contributions from the entire *R* statistical computing community were critical for developing these resources. We encourage community contributions to expand the techniques, bolster their scope and applications, enhance the collection of case-studies, optimize the algorithms, and widen the applications to other data-intense disciplines or complex scientific challenges.

The author is profoundly indebted to all of his direct mentors and advisors for nurturing my curiosity, inspiring my studies, guiding the course of my career, and providing constructive and critical feedback throughout. Among these scholars are Gencho Skordev (Sofia University); Kenneth Kuttler (Michigan Tech University); De Witt L. Sumners and Fred Huffer (Florida State University); Jan de Leeuw, Nicolas Christou, and Michael Mega (UCLA); Arthur Toga (USC); and Brian Athey, Patricia Hurn, Kathleen Potempa, Janet Larson, and Gilbert Omenn (University of Michigan).

Many other colleagues, students, researchers, and fellows have shared their expertise, creativity, valuable time, and critical assessment for generating, validating, and enhancing these open-science resources. Among these are Christopher Aakre, Simeone Marino, Jiachen Xu, Ming Tang, Nina Zhou, Chao Gao, Alexandr Kalinin, Syed Husain, Brady Zhu, Farshid Sepehrband, Lu Zhao, Sam Hobel, Hanbo Sun, Tuo Wang, and many others. Many colleagues from the Statistics Online Computational Resource (SOCR), the Big Data for Discovery Science (BDDS) Center, and the Michigan Institute for Data Science (MIDAS) provided encouragement and valuable suggestions.

The development of the DSPA materials was partially supported by the US National Science Foundation (grants 1734853, 1636840, 1416953, 0716055, and 1023115), US National Institutes of Health (grants P20 NR015331, U54 EB020406, P50 NS091856, P30 DK089503, P30AG053760), and the Elsie Andresen Fiske Research Fund.

Ann Arbor, MI, USA

Ivo D. Dinov

DSPA Application and Use Disclaimer

The Data Science and Predictive Analytics (DSPA) resources are designed to help scientists, trainees, students, and professionals learn the foundation of data science, practical applications, and pragmatics of dealing with concrete datasets, and to experiment in a sandbox of specific case-studies. Neither the author nor the publisher have control over, or make any representation or warranties, expressed or implied, regarding the use of these resources by researchers, users, patients, or their healthcare provider(s), or the use or interpretation of any information stored on, derived, computed, suggested by, or received through any of the DSPA materials, code, scripts, or applications. All users are solely responsible for deriving, interpreting, and communicating any information to (and receiving feedback from) the user's representatives or healthcare provider(s).

Users, their proxies, or representatives (e.g., clinicians) are solely responsible for reviewing and evaluating the accuracy, relevance, and meaning of any information stored on, derived by, generated by, or received through the application of any of the DSPA software, protocols, or techniques. The author and the publisher cannot and do not guarantee said accuracy. *The DSPA resources, their applications, and any information stored on, generated by, or received through them are not intended to be a substitute for professional or expert advice, diagnosis, or treatment. Always seek the advice of a physician or other qualified professional with any questions regarding any real case-study (e.g., medical diagnosis, conditions, prediction, and prognostication). Never disregard professional advice or delay seeking it because of something read or learned through the use of the DSPA material or any information stored on, generated by, or received through the SOCR resources.*

All readers and users acknowledge that the DSPA copyright owners or licensors, in their sole discretion, may from time to time make modifications to the DSPA resources. Such modifications may require corresponding changes to be made in the code, protocols, learning modules, activities, case-studies, and other DSPA materials. Neither the author, publisher, nor licensors shall have any obligation to furnish any maintenance or support services with respect to the DSPA resources.

The DSPA resources are intended for educational purposes only. They are not intended to offer or replace any professional advice nor provide expert opinion. Please speak to qualified professional service providers if you have any specific concerns, case-studies, or questions.

Biomedical, Biosocial, Environmental, and Health Disclaimer

All DSPA information, materials, software, and examples are provided for general education purposes only. Persons using the DSPA data, models, tools, or services for any medical, social, healthcare, or environmental purposes should not rely on accuracy, precision, or significance of the DSPA reported results. While the DSPA resources may be updated periodically, users should independently check against other sources, latest advances, and most accurate peer-reviewed information.

Please consult appropriate professional providers prior to making any lifestyle changes or any actions that may impact those around you, your community, or various real, social, and virtual environments. Qualified and appropriate professionals represent the single best source of information regarding any Biomedical, Biosocial, Environmental, and Health decisions. None of these resources have either explicit or implicit indication of FDA approval!

Any and all liability arising directly or indirectly from the use of the DSPA resources is hereby disclaimed. The DSPA resources are provided “as is” and without any warranty expressed or implied. All direct, indirect, special, incidental, consequential, or punitive damages arising from any use of the DSPA resources or materials contained herein are disclaimed and excluded.

Notations

The following common notations are used throughout this textbook.

Notation	Description
 http://www.socr.umich.edu/people/dinov/courses/DSPA_Topics.html	A link to an interactive live web demonstration. Some of these Live Demos require modern Java and JavaScript enabled browsers and Internet access.
<pre>require(ggplot2) # Comments Loading required package: ggplot2 Data_R_SAS_SPSS_Pubs <- read.csv('https://umich.edu/data', header=T) df <- data.frame(Data_R_SAS_SPSS_Pubs) # convert to long format df <- melt(df, id.vars = 'Year', variable.name = 'Software') ggplot(data=df, aes(x=Year, y=value, color=Software, group = Software)) + geom_line() + geom ## 3 1 a ... ## 20 3 c data_Long ## CaseID Gender Feature Measurement ## 1 1 M Age 5.0 ## 2 2 F Age 6.0</pre>	<i>R</i> fragments of code, reported results in the output shell, or comments. The complete library of all code presented in the textbook is available in electronic format on the DSPA site. Note that: “#” is used for comments, “##” indicates <i>R</i> textual output, - the <i>R</i> code is color-coded to identify different types of comments, instructions, commands and parameters, - Output like “## ... ##” suggests that some of the <i>R</i> output is deleted or compressed to save space, and indenting is used to visually determine the scope of a method, command, or an expression
$\leftarrow$ or $\rightarrow$	In an asymptotic or limiting sense, tending to, convergence, or approaching a value or a limit.
$\ll$ or $\gg$	Left hand size is substantially smaller or larger than the right hand side.
$\sim$	Depending on the context, model definition, similar to, approximately equal to, or equivalent (in probability distribution sense).
package::function	A standard reference notation to functions members of specific <i>R</i> packages.
Case-studies	https://umich.instructure.com/courses/38100/files/folder/Case_Studies
Electronic Materials	http://DSPA.predictive.space
Also see the Glossary and the Index , located in the end of the book.	

Fig. 2 Common DSPA notations

Contents

1	Motivation	1
1.1	DSPA Mission and Objectives	1
1.2	Examples of Driving Motivational Problems and Challenges	2
1.2.1	Alzheimer’s Disease	2
1.2.2	Parkinson’s Disease	2
1.2.3	Drug and Substance Use	3
1.2.4	Amyotrophic Lateral Sclerosis	4
1.2.5	Normal Brain Visualization	4
1.2.6	Neurodegeneration	4
1.2.7	Genetic Forensics: 2013–2016 Ebola Outbreak	5
1.2.8	Next Generation Sequence (NGS) Analysis	6
1.2.9	Neuroimaging-Genetics	7
1.3	Common Characteristics of Big (Biomedical and Health) Data	8
1.4	Data Science	9
1.5	Predictive Analytics	9
1.6	High-Throughput Big Data Analytics	10
1.7	Examples of Data Repositories, Archives, and Services	10
1.8	DSPA Expectations	11
2	Foundations of R	13
2.1	Why Use R?	13
2.2	Getting Started	15
2.2.1	Install Basic Shell-Based R	15
2.2.2	GUI Based R Invocation (RStudio)	15
2.2.3	RStudio GUI Layout	15
2.2.4	Some Notes	16
2.3	Help	16
2.4	Simple Wide-to-Long Data format Translation	17
2.5	Data Generation	18
2.6	Input/Output (I/O)	22

2.7	Slicing and Extracting Data	24
2.8	Variable Conversion	25
2.9	Variable Information	25
2.10	Data Selection and Manipulation	27
2.11	Math Functions	30
2.12	Matrix Operations	32
2.13	Advanced Data Processing	32
2.14	Strings	37
2.15	Plotting	39
2.16	QQ Normal Probability Plot	41
2.17	Low-Level Plotting Commands	45
2.18	Graphics Parameters	45
2.19	Optimization and model Fitting	47
2.20	Statistics	48
2.21	Distributions	49
2.21.1	Programming	49
2.22	Data Simulation Primer	50
2.23	Appendix	56
2.23.1	HTML SOCR Data Import	56
2.23.2	R Debugging	57
2.24	Assignments: 2. R Foundations	60
2.24.1	Confirm that You Have Installed R/RStudio	60
2.24.2	Long-to-Wide Data Format Translation	61
2.24.3	Data Frames	61
2.24.4	Data Stratification	61
2.24.5	Simulation	61
2.24.6	Programming	62
	References	62
3	Managing Data in R	63
3.1	Saving and Loading R Data Structures	63
3.2	Importing and Saving Data from CSV Files	64
3.3	Exploring the Structure of Data	66
3.4	Exploring Numeric Variables	66
3.5	Measuring the Central Tendency: Mean, Median, Mode	67
3.6	Measuring Spread: Quartiles and the Five-Number Summary	68
3.7	Visualizing Numeric Variables: Boxplots	70
3.8	Visualizing Numeric Variables: Histograms	71
3.9	Understanding Numeric Data: Uniform and Normal Distributions	72
3.10	Measuring Spread: Variance and Standard Deviation	73
3.11	Exploring Categorical Variables	76

3.12	Exploring Relationships Between Variables	77
3.13	Missing Data	79
3.13.1	Simulate Some Real Multivariate Data	84
3.13.2	TBI Data Example	98
3.13.3	Imputation via Expectation-Maximization	122
3.14	Parsing Webpages and Visualizing Tabular HTML Data	130
3.15	Cohort-Rebalancing (for Imbalanced Groups)	135
3.16	Appendix	138
3.16.1	Importing Data from SQL Databases	138
3.16.2	R Code Fragments	139
3.17	Assignments: 3. Managing Data in R	140
3.17.1	Import, Plot, Summarize and Save Data	140
3.17.2	Explore some Bivariate Relations in the Data	140
3.17.3	Missing Data	141
3.17.4	Surface Plots	141
3.17.5	Unbalanced Designs	141
3.17.6	Aggregate Analysis	141
	References	141
4	Data Visualization	143
4.1	Common Questions	143
4.2	Classification of Visualization Methods	144
4.3	Composition	144
4.3.1	Histograms and Density Plots	144
4.3.2	Pie Chart	147
4.3.3	Heat Map	149
4.4	Comparison	152
4.4.1	Paired Scatter Plots	152
4.4.2	Jitter Plot	157
4.4.3	Bar Plots	159
4.4.4	Trees and Graphs	164
4.4.5	Correlation Plots	167
4.5	Relationships	171
4.5.1	Line Plots Using ggplot	171
4.5.2	Density Plots	173
4.5.3	Distributions	173
4.5.4	2D Kernel Density and 3D Surface Plots	174
4.5.5	Multiple 2D Image Surface Plots	176
4.5.6	3D and 4D Visualizations	178
4.6	Appendix	183
4.6.1	Hands-on Activity (Health Behavior Risks)	183
4.6.2	Additional ggplot Examples	187

4.7	Assignments 4: Data Visualization	198
4.7.1	Common Plots	198
4.7.2	Trees and Graphs	198
4.7.3	Exploratory Data Analytics (EDA)	199
	References	199
5	Linear Algebra & Matrix Computing	201
5.1	Matrices (Second Order Tensors)	202
5.1.1	Create Matrices	202
5.1.2	Adding Columns and Rows	203
5.2	Matrix Subscripts	204
5.3	Matrix Operations	204
5.3.1	Addition	204
5.3.2	Subtraction	205
5.3.3	Multiplication	205
5.3.4	Element-wise Division	207
5.3.5	Transpose	207
5.3.6	Multiplicative Inverse	207
5.4	Matrix Algebra Notation	209
5.4.1	Linear Models	209
5.4.2	Solving Systems of Equations	210
5.4.3	The Identity Matrix	212
5.5	Scalars, Vectors and Matrices	213
5.5.1	Sample Statistics (Mean, Variance)	215
5.5.2	Least Square Estimation	218
5.6	Eigenvalues and Eigenvectors	219
5.7	Other Important Functions	220
5.8	Matrix Notation (Another View)	220
5.9	Multivariate Linear Regression	224
5.10	Sample Covariance Matrix	227
5.11	Assignments: 5. Linear Algebra & Matrix Computing	229
5.11.1	How Is Matrix Multiplication Defined?	229
5.11.2	Scalar Versus Matrix Multiplication	229
5.11.3	Matrix Equations	229
5.11.4	Least Square Estimation	230
5.11.5	Matrix Manipulation	230
5.11.6	Matrix Transpose	230
5.11.7	Sample Statistics	230
5.11.8	Least Square Estimation	230
5.11.9	Eigenvalues and Eigenvectors	231
	References	231
6	Dimensionality Reduction	233
6.1	Example: Reducing 2D to 1D	233
6.2	Matrix Rotations	237
6.3	Notation	242

6.4	Summary (PCA vs. ICA vs. FA)	242
6.5	Principal Component Analysis (PCA)	243
6.5.1	Principal Components	243
6.6	Independent Component Analysis (ICA)	250
6.7	Factor Analysis (FA)	254
6.8	Singular Value Decomposition (SVD)	256
6.9	SVD Summary	258
6.10	Case Study for Dimension Reduction (Parkinson's Disease)	258
6.11	Assignments: 6. Dimensionality Reduction	265
6.11.1	Parkinson's Disease Example	265
6.11.2	Allometric Relations in Plants Example	266
	References	266
7	Lazy Learning: Classification Using Nearest Neighbors	267
7.1	Motivation	268
7.2	The kNN Algorithm Overview	269
7.2.1	Distance Function and Dummy Coding	269
7.2.2	Ways to Determine k	270
7.2.3	Rescaling of the Features	270
7.2.4	Rescaling Formulas	271
7.3	Case Study	271
7.3.1	Step 1: Collecting Data	271
7.3.2	Step 2: Exploring and Preparing the Data	272
7.3.3	Normalizing Data	273
7.3.4	Data Preparation: Creating Training and Testing Datasets	274
7.3.5	Step 3: Training a Model On the Data	274
7.3.6	Step 4: Evaluating Model Performance	274
7.3.7	Step 5: Improving Model Performance	275
7.3.8	Testing Alternative Values of k	276
7.3.9	Quantitative Assessment (Tables 7.2 and 7.3)	282
7.4	Assignments: 7. Lazy Learning: Classification Using Nearest Neighbors	286
7.4.1	Traumatic Brain Injury (TBI)	286
7.4.2	Parkinson's Disease	286
7.4.3	KNN Classification in a High Dimensional Space	287
7.4.4	KNN Classification in a Lower Dimensional Space	287
	References	287
8	Probabilistic Learning: Classification Using Naive Bayes	289
8.1	Overview of the Naive Bayes Algorithm	289
8.2	Assumptions	290
8.3	Bayes Formula	290
8.4	The Laplace Estimator	292

8.5	Case Study: Head and Neck Cancer Medication	293
8.5.1	Step 1: Collecting Data	293
8.5.2	Step 2: Exploring and Preparing the Data	293
8.5.3	Step 3: Training a Model on the Data	299
8.5.4	Step 4: Evaluating Model Performance	300
8.5.5	Step 5: Improving Model Performance	301
8.5.6	Step 6: Compare Naive Bayesian against LDA	302
8.6	Practice Problem	303
8.7	Assignments 8: Probabilistic Learning: Classification	
	Using Naive Bayes	304
8.7.1	Explain These Two Concepts	304
8.7.2	Analyzing Textual Data	305
	References	305
9	Decision Tree Divide and Conquer Classification	307
9.1	Motivation	307
9.2	Hands-on Example: Iris Data	308
9.3	Decision Tree Overview	310
9.3.1	Divide and Conquer	311
9.3.2	Entropy	312
9.3.3	Misclassification Error and Gini Index	313
9.3.4	C5.0 Decision Tree Algorithm	313
9.3.5	Pruning the Decision Tree	315
9.4	Case Study 1: Quality of Life and Chronic Disease	316
9.4.1	Step 1: Collecting Data	316
9.4.2	Step 2: Exploring and Preparing the Data	316
9.4.3	Step 3: Training a Model On the Data	319
9.4.4	Step 4: Evaluating Model Performance	322
9.4.5	Step 5: Trial Option	323
9.4.6	Loading the Misclassification Error Matrix	324
9.4.7	Parameter Tuning	325
9.5	Compare Different Impurity Indices	331
9.6	Classification Rules	331
9.6.1	Separate and Conquer	331
9.6.2	The One Rule Algorithm	332
9.6.3	The RIPPER Algorithm	332
9.7	Case Study 2: QoL in Chronic Disease (Take 2)	332
9.7.1	Step 3: Training a Model on the Data	332
9.7.2	Step 4: Evaluating Model Performance	333
9.7.3	Step 5: Alternative Model1	334
9.7.4	Step 5: Alternative Model2	334
9.8	Practice Problem	337

9.9	Assignments 9: Decision Tree Divide and Conquer	
	Classification	342
9.9.1	Explain These Concepts	342
9.9.2	Decision Tree Partitioning	342
	References	343
10	Forecasting Numeric Data Using Regression Models	345
10.1	Understanding Regression	345
10.1.1	Simple Linear Regression	345
10.2	Ordinary Least Squares Estimation	347
10.2.1	Model Assumptions	349
10.2.2	Correlations	349
10.2.3	Multiple Linear Regression	350
10.3	Case Study 1: Baseball Players	352
10.3.1	Step 1: Collecting Data	352
10.3.2	Step 2: Exploring and Preparing the Data	352
10.3.3	Exploring Relationships Among Features: The Correlation Matrix	356
10.3.4	Visualizing Relationships Among Features: The Scatterplot Matrix	356
10.3.5	Step 3: Training a Model on the Data	358
10.3.6	Step 4: Evaluating Model Performance	359
10.4	Step 5: Improving Model Performance	361
10.4.1	Model Specification: Adding Non-linear Relationships	369
10.4.2	Transformation: Converting a Numeric Variable to a Binary Indicator	370
10.4.3	Model Specification: Adding Interaction Effects	371
10.5	Understanding Regression Trees and Model Trees	373
10.5.1	Adding Regression to Trees	373
10.6	Case Study 2: Baseball Players (Take 2)	374
10.6.1	Step 2: Exploring and Preparing the Data	374
10.6.2	Step 3: Training a Model On the Data	375
10.6.3	Visualizing Decision Trees	375
10.6.4	Step 4: Evaluating Model Performance	377
10.6.5	Measuring Performance with Mean Absolute Error	378
10.6.6	Step 5: Improving Model Performance	378
10.7	Practice Problem: Heart Attack Data	380
10.8	Assignments: 10. Forecasting Numeric Data Using Regression Models	381
	References	381

11	Black Box Machine-Learning Methods: Neural Networks and Support Vector Machines	383
11.1	Understanding Neural Networks	383
11.1.1	From Biological to Artificial Neurons	383
11.1.2	Activation Functions	384
11.1.3	Network Topology	386
11.1.4	The Direction of Information Travel	386
11.1.5	The Number of Nodes in Each Layer	386
11.1.6	Training Neural Networks with Backpropagation	387
11.2	Case Study 1: Google Trends and the Stock Market: Regression	388
11.2.1	Step 1: Collecting Data	388
11.2.2	Step 2: Exploring and Preparing the Data	389
11.2.3	Step 3: Training a Model on the Data	391
11.2.4	Step 4: Evaluating Model Performance	392
11.2.5	Step 5: Improving Model Performance	393
11.2.6	Step 6: Adding Additional Layers	394
11.3	Simple NN Demo: Learning to Compute $\sqrt{\cdot}$	394
11.4	Case Study 2: Google Trends and the Stock Market – Classification	396
11.5	Support Vector Machines (SVM)	398
11.5.1	Classification with Hyperplanes	399
11.6	Case Study 3: Optical Character Recognition (OCR)	403
11.6.1	Step 1: Prepare and Explore the Data	404
11.6.2	Step 2: Training an SVM Model	405
11.6.3	Step 3: Evaluating Model Performance	406
11.6.4	Step 4: Improving Model Performance	408
11.7	Case Study 4: Iris Flowers	409
11.7.1	Step 1: Collecting Data	409
11.7.2	Step 2: Exploring and Preparing the Data	409
11.7.3	Step 3: Training a Model on the Data	411
11.7.4	Step 4: Evaluating Model Performance	412
11.7.5	Step 5: RBF Kernel Function	413
11.7.6	Parameter Tuning	413
11.7.7	Improving the Performance of Gaussian Kernels	415
11.8	Practice	416
11.8.1	Problem 1 Google Trends and the Stock Market	416
11.8.2	Problem 2: Quality of Life and Chronic Disease	416
11.9	Appendix	420
11.10	Assignments: 11. Black Box Machine-Learning Methods: Neural Networks and Support Vector Machines	421
11.10.1	Learn and Predict a Power-Function	421
11.10.2	Pediatric Schizophrenia Study	421
	References	422

12	Apriori Association Rules Learning	423
12.1	Association Rules	423
12.2	The Apriori Algorithm for Association Rule Learning	424
12.3	Measuring Rule Importance by Using Support and Confidence	424
12.4	Building a Set of Rules with the Apriori Principle	425
12.5	A Toy Example	426
12.6	Case Study 1: Head and Neck Cancer Medications	427
12.6.1	Step 1: Collecting Data	427
12.6.2	Step 2: Exploring and Preparing the Data	427
12.6.3	Step 3: Training a Model on the Data	432
12.6.4	Step 4: Evaluating Model Performance	433
12.6.5	Step 5: Improving Model Performance	435
12.7	Practice Problems: Groceries	438
12.8	Summary	441
12.9	Assignments: 12. Apriori Association Rules Learning	442
	References	442
13	k-Means Clustering	443
13.1	Clustering as a Machine Learning Task	443
13.2	Silhouette Plots	446
13.3	The k-Means Clustering Algorithm	447
13.3.1	Using Distance to Assign and Update Clusters	447
13.3.2	Choosing the Appropriate Number of Clusters	448
13.4	Case Study 1: Divorce and Consequences on Young Adults	448
13.4.1	Step 1: Collecting Data	448
13.4.2	Step 2: Exploring and Preparing the Data	449
13.4.3	Step 3: Training a Model on the Data	450
13.4.4	Step 4: Evaluating Model Performance	451
13.4.5	Step 5: Usage of Cluster Information	454
13.5	Model Improvement	455
13.5.1	Tuning the Parameter k	457
13.6	Case Study 2: Pediatric Trauma	459
13.6.1	Step 1: Collecting Data	459
13.6.2	Step 2: Exploring and Preparing the Data	460
13.6.3	Step 3: Training a Model on the Data	461
13.6.4	Step 4: Evaluating Model Performance	462
13.6.5	Practice Problem: Youth Development	465
13.7	Hierarchical Clustering	467
13.8	Gaussian Mixture Models	470
13.9	Summary	472
13.10	Assignments: 13. k-Means Clustering	472
	References	473

14	Model Performance Assessment	475
14.1	Measuring the Performance of Classification Methods	475
	Evaluation Strategies	477
14.1.1	Binary Outcomes	477
14.1.2	Confusion Matrices	478
14.1.3	Other Measures of Performance Beyond Accuracy	480
14.1.4	The Kappa (κ) Statistic	481
14.1.5	Computation of Observed Accuracy and Expected Accuracy	484
14.1.6	Sensitivity and Specificity	485
14.1.7	Precision and Recall	486
14.1.8	The F-Measure	487
14.2	Visualizing Performance Tradeoffs (ROC Curve)	488
14.3	Estimating Future Performance (Internal Statistical Validation)	491
14.3.1	The Holdout Method	491
14.3.2	Cross-Validation	492
14.3.3	Bootstrap Sampling	494
14.4	Assignment: 14. Evaluation of Model Performance	495
	References	496
15	Improving Model Performance	497
15.1	Improving Model Performance by Parameter Tuning	497
15.2	Using <code>caret</code> for Automated Parameter Tuning	497
15.2.1	Customizing the Tuning Process	501
15.2.2	Improving Model Performance with Meta-learning	502
15.2.3	Bagging	503
15.2.4	Boosting	505
15.2.5	Random Forests	506
15.2.6	Adaptive Boosting	508
15.3	Assignment: 15. Improving Model Performance	510
15.3.1	Model Improvement Case Study	511
	References	511
16	Specialized Machine Learning Topics	513
16.1	Working with Specialized Data and Databases	513
16.1.1	Data Format Conversion	514
16.1.2	Querying Data in SQL Databases	515
16.1.3	Real Random Number Generation	521
16.1.4	Downloading the Complete Text of Web Pages	522
16.1.5	Reading and Writing XML with the XML Package	523
16.1.6	Web-Page Data Scraping	524
16.1.7	Parsing JSON from Web APIs	525
16.1.8	Reading and Writing Microsoft Excel Spreadsheets Using XLSX	526

16.2	Working with Domain-Specific Data	527
16.2.1	Working with Bioinformatics Data	527
16.2.2	Visualizing Network Data	528
16.3	Data Streaming	533
16.3.1	Definition	533
16.3.2	The <code>stream</code> Package	534
16.3.3	Synthetic Example: Random Gaussian Stream	534
16.3.4	Sources of Data Streams	536
16.3.5	Printing, Plotting and Saving Streams	537
16.3.6	Stream Animation	538
16.3.7	Case-Study: SOCR Knee Pain Data	540
16.3.8	Data Stream Clustering and Classification (DSC)	542
16.3.9	Evaluation of Data Stream Clustering	545
16.4	Optimization and Improving the Computational Performance	546
16.4.1	Generalizing Tabular Data Structures with <code>dplyr</code>	547
16.4.2	Making Data Frames Faster with <code>Data.Table</code>	548
16.4.3	Creating Disk-Based Data Frames with <code>ff</code>	548
16.4.4	Using Massive Matrices with <code>bigmemory</code>	549
16.5	Parallel Computing	549
16.5.1	Measuring Execution Time	550
16.5.2	Parallel Processing with Multiple Cores	550
16.5.3	Parallelization Using <code>foreach</code> and <code>doParallel</code>	552
16.5.4	GPU Computing	553
16.6	Deploying Optimized Learning Algorithms	553
16.6.1	Building Bigger Regression Models with <code>biglm</code>	553
16.6.2	Growing Bigger and Faster Random Forests with <code>bigrf</code>	553
16.6.3	Training and Evaluation Models in Parallel with <code>caret</code>	554
16.7	Practice Problem	554
16.8	Assignment: 16. Specialized Machine Learning Topics	555
16.8.1	Working with Website Data	555
16.8.2	Network Data and Visualization	555
16.8.3	Data Conversion and Parallel Computing	555
	References	556
17	Variable/Feature Selection	557
17.1	Feature Selection Methods	557
17.1.1	Filtering Techniques	557
17.1.2	Wrapper Methods	558
17.1.3	Embedded Techniques	558
17.2	Case Study: ALS	559
17.2.1	Step 1: Collecting Data	559
17.2.2	Step 2: Exploring and Preparing the Data	559

17.2.3	Step 3: Training a Model on the Data	560
17.2.4	Step 4: Evaluating Model Performance	564
17.3	Practice Problem	569
17.4	Assignment: 17. Variable/Feature Selection	571
17.4.1	Wrapper Feature Selection	571
17.4.2	Use the PPMI Dataset	571
	References	572
18	Regularized Linear Modeling and Controlled Variable Selection	573
18.1	Questions	574
18.2	Matrix Notation	574
18.3	Regularized Linear Modeling	574
18.3.1	Ridge Regression	576
18.3.2	Least Absolute Shrinkage and Selection Operator (LASSO) Regression	579
18.3.3	Predictor Standardization	582
18.3.4	Estimation Goals	582
18.4	Linear Regression	582
18.4.1	Drawbacks of Linear Regression	583
18.4.2	Assessing Prediction Accuracy	583
18.4.3	Estimating the Prediction Error	583
18.4.4	Improving the Prediction Accuracy	584
18.4.5	Variable Selection	585
18.5	Regularization Framework	586
18.5.1	Role of the Penalty Term	586
18.5.2	Role of the Regularization Parameter	586
18.5.3	LASSO	587
18.5.4	General Regularization Framework	587
18.6	Implementation of Regularization	588
18.6.1	Example: Neuroimaging-Genetics Study of Parkinson's Disease Dataset	588
18.6.2	Computational Complexity	590
18.6.3	LASSO and Ridge Solution Paths	590
18.6.4	Choice of the Regularization Parameter	598
18.6.5	Cross Validation Motivation	599
18.6.6	n -Fold Cross Validation	599
18.6.7	LASSO 10-Fold Cross Validation	600
18.6.8	Stepwise OLS (Ordinary Least Squares)	601
18.6.9	Final Models	602
18.6.10	Model Performance	604
18.6.11	Comparing Selected Features	604
18.6.12	Summary	605
18.7	Knock-off Filtering: Simulated Example	605
18.7.1	Notes	607

18.8	PD Neuroimaging-Genetics Case-Study	608
18.8.1	Fetching, Cleaning and Preparing the Data	608
18.8.2	Preparing the Response Vector	609
18.8.3	False Discovery Rate (FDR)	617
18.8.4	Running the Knockoff Filter	620
18.9	Assignment: 18. Regularized Linear Modeling and Knockoff Filtering	621
	References	622
19	Big Longitudinal Data Analysis	623
19.1	Time Series Analysis	623
19.1.1	Step 1: Plot Time Series	626
19.1.2	Step 2: Find Proper Parameter Values for ARIMA Model	628
19.1.3	Check the Differencing Parameter	629
19.1.4	Identifying the AR and MA Parameters	630
19.1.5	Step 3: Build an ARIMA Model	632
19.1.6	Step 4: Forecasting with ARIMA Model	637
19.2	Structural Equation Modeling (SEM)-Latent Variables	638
19.2.1	Foundations of SEM	638
19.2.2	SEM Components	641
19.2.3	Case Study – Parkinson’s Disease (PD)	642
19.2.4	Outputs of Lavaan SEM	647
19.3	Longitudinal Data Analysis-Linear Mixed Models	648
19.3.1	Mean Trend	648
19.3.2	Modeling the Correlation	652
19.4	GLMM/GEE Longitudinal Data Analysis	653
19.4.1	GEE Versus GLMM	655
19.5	Assignment: 19. Big Longitudinal Data Analysis	657
19.5.1	Imaging Data	657
19.5.2	Time Series Analysis	658
19.5.3	Latent Variables Model	658
	References	658
20	Natural Language Processing/Text Mining	659
20.1	A Simple NLP/TM Example	660
20.1.1	Define and Load the Unstructured-Text Documents	661
20.1.2	Create a New VCorpus Object	663
20.1.3	To-Lower Case Transformation	664
20.1.4	Text Pre-processing	664
20.1.5	Bags of Words	666
20.1.6	Document Term Matrix	667

20.2	Case-Study: Job Ranking	669
20.2.1	Step 1: Make a VCorpus Object	670
20.2.2	Step 2: Clean the VCorpus Object	670
20.2.3	Step 3: Build the Document Term Matrix	670
20.2.4	Area Under the ROC Curve	674
20.3	TF-IDF	676
20.3.1	Term Frequency (TF)	676
20.3.2	Inverse Document Frequency (IDF)	676
20.3.3	TF-IDF	677
20.4	Cosine Similarity	685
20.5	Sentiment Analysis	686
20.5.1	Data Preprocessing	686
20.5.2	NLP/TM Analytics	689
20.5.3	Prediction Optimization	692
20.6	Assignment: 20. Natural Language Processing/Text Mining	694
20.6.1	Mining Twitter Data	694
20.6.2	Mining Cancer Clinical Notes	695
	References	695
21	Prediction and Internal Statistical Cross Validation	697
21.1	Forecasting Types and Assessment Approaches	697
21.2	Overfitting	698
21.2.1	Example (US Presidential Elections)	698
21.2.2	Example (Google Flu Trends)	698
21.2.3	Example (Autism)	700
21.3	Internal Statistical Cross-Validation is an Iterative Process	701
21.4	Example (Linear Regression)	702
21.4.1	Cross-Validation Methods	703
21.4.2	Exhaustive Cross-Validation	703
21.4.3	Non-Exhaustive Cross-Validation	704
21.5	Case-Studies	704
21.5.1	Example 1: Prediction of Parkinson's Disease Using Adaptive Boosting (AdaBoost)	705
21.5.2	Example 2: Sleep Dataset	708
21.5.3	Example 3: Model-Based (Linear Regression) Prediction Using the Attitude Dataset	710
21.5.4	Example 4: Parkinson's Data (ppmi_data)	711
21.6	Summary of CV output	712
21.7	Alternative Predictor Functions	712
21.7.1	Logistic Regression	713
21.7.2	Quadratic Discriminant Analysis (QDA)	714
21.7.3	Foundation of LDA and QDA for Prediction, Dimensionality Reduction, and Forecasting	715
21.7.4	Neural Networks	717
21.7.5	SVM	718

21.7.6	k-Nearest Neighbors Algorithm (k-NN)	719
21.7.7	k-Means Clustering (k-MC)	720
21.7.8	Spectral Clustering	727
21.8	Compare the Results	730
21.9	Assignment: 21. Prediction and Internal Statistical Cross-Validation	733
	References	734
22	Function Optimization	735
22.1	Free (Unconstrained) Optimization	735
22.1.1	Example 1: Minimizing a Univariate Function (Inverse-CDF)	736
22.1.2	Example 2: Minimizing a Bivariate Function	738
22.1.3	Example 3: Using Simulated Annealing to Find the Maximum of an Oscillatory Function	739
22.2	Constrained Optimization	740
22.2.1	Equality Constraints	740
22.2.2	Lagrange Multipliers	740
22.2.3	Inequality Constrained Optimization	741
	Quadratic Programming (QP)	747
22.3	General Non-linear Optimization	748
22.3.1	Dual Problem Optimization	749
22.4	Manual Versus Automated Lagrange Multiplier Optimization	753
22.5	Data Denoising	756
22.6	Assignment: 22. Function Optimization	761
22.6.1	Unconstrained Optimization	761
22.6.2	Linear Programming (LP)	761
22.6.3	Mixed Integer Linear Programming (MILP)	762
22.6.4	Quadratic Programming (QP)	762
22.6.5	Complex Non-linear Optimization	762
22.6.6	Data Denoising	763
	References	763
23	Deep Learning, Neural Networks	765
23.1	Deep Learning Training	766
23.1.1	Perceptrons	766
23.2	Biological Relevance	768
23.3	Simple Neural Net Examples	770
23.3.1	Exclusive OR (XOR) Operator	770
23.3.2	NAND Operator	771
23.3.3	Complex Networks Designed Using Simple Building Blocks	772
23.4	Classification	773
23.4.1	Sonar Data Example	774
23.4.2	MXNet Notes	781

23.5	Case-Studies	782
23.5.1	ALS Regression Example	783
23.5.2	Spirals 2D Data	785
23.5.3	IBS Study	789
23.5.4	Country QoL Ranking Data	792
23.5.5	Handwritten Digits Classification	795
23.6	Classifying Real-World Images	806
23.6.1	Load the Pre-trained Model	806
23.6.2	Load, Preprocess and Classify New Images – US Weather Pattern	806
23.6.3	Lake Mapourika, New Zealand	810
23.6.4	Beach Image	811
23.6.5	Volcano	812
23.6.6	Brain Surface	814
23.6.7	Face Mask	815
23.7	Assignment: 23. Deep Learning, Neural Networks	816
23.7.1	Deep Learning Classification	816
23.7.2	Deep Learning Regression	817
23.7.3	Image Classification	817
	References	817
	Summary	819
	Glossary	823
	Index	825