THOMAS W. MILLER
Faculty Director of Northwestern University's Predictive Analytics Program

WEB AND NETWORK DATA SCIENCE

MODELING TECHNIQUES IN PREDICTIVE ANALYTICS

Web and Network Data Science

Modeling Techniques in Predictive Analytics

THOMAS W. MILLER

Editor-in-Chief: Amy Neidlinger
Executive Editor: Jeanne Glasser
Operations Specialist: Jodi Kemper
Cover Designer: Alan Clements
Managing Editor: Kristy Hart
Project Editor: Andy Beaster
Senior Compositor: Gloria Schurick
Manufacturing Buyer: Dan Uhrig

Published by Pearson Education, Inc.
Upper Saddle River, New Jersey 07458

Printed in the United States of America

First Printing December 2014

ISBN-10: 0-13-388644-1
ISBN-13: 978-0-13-388644-3

Pearson Education LTD.
Pearson Education Australia PTY, Limited.
Pearson Education Singapore, Pte. Ltd.
Pearson Education Asia, Ltd.
Pearson Education Canada, Ltd.
Pearson Educacin de Mexico, S.A. de C.V.
Pearson Education—Japan
Pearson Education Malaysia, Pte. Ltd.
Library of Congress Control Number: 2014956958

Contents

Preface

"Scotty, beam me up."

—WILLIAM SHATNER AS CAPTAIN KIRK IN
Star Trek IV: The Voyage Home (1986)

The web is a network of linked pages. The web is a communication medium. The web is the locus of the world's information. We spend much of our time searching the web, extracting relevant data, and analyzing those data. Our lives are easier when we can work efficiently on the web. This book shows how.

The book emerged from a course I teach at Northwestern University. The course started as an introduction to website analytics, looking at usage statistics and performance in search. Then I added concepts from network science and social media. After teaching the course for two years, I realized that gathering information from the web provided a unifying theme. There is much to learn about web and network data science. This book, like the course, provides a guide.

Web and network data science is data science and network science combined, focusing on the web as an information resource. And the best way to learn about it is to work through examples. We include many examples in this book. We help researchers and analysts by providing a ready resource and reference guide for modeling techniques. We show programmers how to build on a foundation of code that works to solve real business problems.

The truth about what we do is in the programs we write. It is there for everyone to see and for some to debug. To promote student learning, each program includes step-by-step comments and suggestions for taking the analysis further. Data sets and computer programs are available from the book's website at `http://www.ftpress.com/miller/`.

Python gets its name from Monty Python. We see packages with devious names such as Twisted and Scrapy. R has its lubridate and zoo. Good things come from people who work and have fun at the same time. It is fun rather than profit or fame that motivates contributors to open source, and I am happy to be part of the Python and R communities. Let the fun begin.

When working on web and network problems, some things are more easily accomplished with Python, others with R. And there are times when it is good to offer solutions in both languages, checking one against the other. Together, Python and R are good at gathering web and network data and analyzing those data.

There is a long list of programming tools we mention only in passing. Web masters, charged with the task of making things happen on the web, rely on additional languages and technologies, including JavaScript, Apache and .Net web services, and database systems. We discuss these technologies but do not provide programming code.

Most of the data in the book were obtained from public domain data sources. Supporting data for the cases come from the University of California–Irvine Machine Learning Repository and the Stanford Large Network Dataset Collection. Movie information was obtained courtesy of The Internet Movie Database, used with permission. IMDb movie reviews data were organized by Andrew L. Mass and his colleagues at Stanford University. William W. Cohen of Carnegie Mellon University maintains the data for the Enron case. Maksim Tsvetovat maintains the data for the Quake Talk case. We are most thankful to these scholars for providing access to rich data sets for research.

Many have influenced my intellectual development over the years. There were those good thinkers and good people, teachers and mentors for whom I will be forever grateful. Sadly, no longer with us are Gerald Hahn Hinkle in philosophy and Allan Lake Rice in languages at Ursinus College, and Herbert Feigl in philosophy at the University of Minnesota. I am also most thankful to David J. Weiss in psychometrics at the University of Minnesota

and Kelly Eakin in economics, formerly at the University of Oregon. Good teachers—yes, great teachers—are valued for a lifetime.

Thanks to Stan Narusiewcz who gave me my first job in business as a network engineer and to Tom Obinger who showed me how to be successful in selling computer systems as well as networks. Along with Bill JoBush and Brian Hill, they served as able managers and colleagues across various parts of my career as an information systems professional.

Thanks to Michael L. Rothschild, Neal M. Ford, Peter R. Dickson, and Janet Christopher who provided invaluable support during our years together at the University of Wisconsin–Madison. I am most grateful to the students and executive advisory board members of the A. C. Nielsen Center for Marketing Research and to Jeff Walkowski and Neli Esipova who worked with me in exploring online surveys and focus groups when those methods were just starting to be used for primary research.

I am fortunate to be involved with graduate distance education at Northwestern University's School of Professional Studies. Thanks to Glen Fogerty, who offered me the opportunity to teach and take a leadership role in the predictive analytics program at Northwestern University. Thanks to colleagues and staff who administer this exceptional graduate program. And thanks to the many students and fellow faculty from whom I have learned.

ToutBay is an emerging firm in the data science space. With co-founder Greg Blence, I have great hopes for growth in the coming years. Thanks to Greg for joining me in this effort and for keeping me grounded in the practical needs of business. Academics and data science models can take us only so far. Eventually, to make a difference, we must implement our ideas and models, sharing them with one another.

I live in California, four miles north of Dodger Stadium, teach for Northwestern University in Evanston, Illinois, and direct product development at ToutBay, a data science firm in Tampa, Florida. Such are the benefits of a good Internet connection.

Amy Hendrickson of TEXnology Inc. applied her craft, making words, tables, and figures look beautiful in print—another victory for open source. Thanks to Donald Knuth and the TEX/LATEX community for their contributions to this wonderful system for typesetting and publication.

The book draws on materials developed for the web and network data science course at Northwestern University. Students from that course provided ideas and inspiration. Lorena Martin reviewed the book and provided much needed feedback. Candice Bradley served dual roles as a reviewer and copyeditor. I am most grateful for their help and encouragement. Thanks also to my editor, Jeanne Glasser Levine, and publisher, Pearson/FT Press, for making this book possible. Any writing issues, errors, or items of unfinished business, of course, are my responsibility alone.

My good friend Brittney and her daughter Janiya keep me company when time permits. And my son Daniel is there for me in good times and bad, a friend for life. My greatest debt is to them because they believe in me.

Thomas W. Miller
Glendale, California
November 2014

Figures

Tables

This page intentionally left blank

Exhibits

1 Being Technically Inclined

"Why don't you come up sometime and see me?"

—MAE WEST AS LADY LOU IN *She Done Him Wrong* (1933)

I began my business career working as a network engineer in Roseville, Minnesota. Just out of graduate training in statistics at the University of Minnesota, I was well schooled in math and models but lacking business understanding. It did not take long to learn that success in my job meant coming up with meaningful answers for management.

In the dial-up and leased-line world of the late 1970s, asynchronous, bisynchronous, and synchronous connections ruled the day. We translated network protocols into polling and message bits and noted the bits per second that each communication line could accommodate. Queuing theory and discrete event simulation guided the analysis.

A bank teller would make a request, hitting the return key at a terminal. The terminal was connected to a control unit, which in turn was connected to a remote concentrator processor. Leased lines went from remote concentrator processors to a front-end processor, providing a channel to the mainframe computer. These were the nodes and links of networks at the time. The queuing problem involved estimating how long the bank teller would have to wait to get a response from the mainframe.

Fast forward forty years. We have moved away from dial-up and leased lines. Protocols are packet-switched and mobile. Users of networks are everywhere, not just at banks, businesses, and research establishments. Most mainframes have been replaced by clusters of microcomputers. We carry the smallest of computers in our pockets. We wear computers if we like. Of course, when making requests of remote systems, we are still waiting for responses, although now we wait wherever we are and whatever we are doing.

With computer hardware looking more like a commodity and software going open-source, established technology firms seek out new opportunities in business intelligence and data science. IBM moves from hardware to software to consulting. HP splits into two firms, one focused on hardware, the other on business services and utilities. Meanwhile, Apple fights battles with Amazon and Google over the distribution of media, while suing Samsung for copyright violations.

The big battles of today concern information and its online distribution. Intellectual property, special knowledge, competitive intelligence, expertise, and art—these add value in an online world that otherwise appears to offer information for free.

It is hard to resist the allure of the web. She is the ultimate seductress, holding the promise of unlimited information and connection to all. The web is a huge data repository, a path to the world's knowledge, and the research medium through which we develop new knowledge.

Web and network data science is a collection of technologies and modeling techniques, some well understood, others emerging, that help us to understand the web and the networks in our lives. The technologies of the web are many, with current market shares tracked by Alexa Internet (2014) and W3Techs (2014), among others.

To work efficiently in web and network data science, it helps to be technically inclined, with some understanding of at least three languages: Python, R, and JavaScript. Python is the tool of choice for data preparation (or data munging, as it is sometimes called). R provides specialized tools for modeling and data visualization. And JavaScript is the client-side language of the web, available on every major web browser. When working on web and network problems, it also helps to know HTML5, CSS3, XPath, a vari-

ety of text and image file formats, Java, Linux, Apache, .Net web services, database systems, and server-side languages such as Perl and PHP. It helps to be technically inclined, but there is a limit to what we can cover in one book. We provide a glossary of terms as the final appendix in the book.

From its humble beginnings as a language that Brendan Eich developed in ten days in 1995 at the former company Netscape, JavaScript has emerged as the client-side language of the web, a browser-based engine for managing user interaction. JavaScript is dominant on the client side, with an estimated 88 percent of websites using the technology and with 11.8 percent of websites being pure/static HTML sites with no client-side programming (W3Techs 2014).

Crockford (2008) tells us what is right and wrong with JavaScript. Others tell us how to use it in practice (Stefanov 2010; Flanagan 2011; Resig and BearBibeault 2013). Recently, with the emergence of Node.js, JavaScript has taken on a role on the server side (Hughes-Croucher and Wilson 2012; Wanderschneider 2013; Cantelon, Harter, Holowaychuk, and Rajlich 2014). There are those who promote end-to-end JavaScript applications with client- and server-side programs and document databases (Mikowski and Powell 2014). JavaScript Object Notation (JSON), a data interchange format, is more readable than XML and easily integrated into a MongoDB document database (Chodorow 2013; Copeland 2013; Hoberman 2014), for example. JavaScript would certainly rule the web if it had sufficient capabilities as a modeling and analysis language. It does not.

Today's world of data science brings together statisticians fluent in R and information technology professionals fluent in Python. These communities have much to learn from each other. For the practicing data scientist, there are considerable advantages to being multilingual.

Designed by Ross Ihaka and Robert Gentleman, R first appeared in 1993. R represents an extensible, object-oriented, open-source scripting language for programming with data. It is well established in the statistical community and has syntax, data structures, and methods similar to its precursors, S and S-Plus. Contributors to the language have provided more than five thousand packages, most focused on traditional statistics, machine learning, and data visualization. R is the most widely used language in data science, but it is not a general-purpose programming language.

Guido van Rossum, a fan of Monty Python, released version 1.0 of Python in 1994. This general-purpose language has grown in popularity in the ensuing years. Many systems programmers have moved from Perl to Python, and Python has a strong following among mathematicians and scientists. Many universities use Python as a way to introduce basic concepts of object-oriented programming. An active open-source community has contributed more than fifteen thousand Python packages.

Sometimes referred to as a "glue language," Python provides a rich open-source environment for scientific programming and research. For computer-intensive applications, it gives us the ability to call on compiled routines from C, C++, and Fortran. We can also use Cython to convert Python code into optimized C. For modeling techniques or graphics not currently implemented in Python, we can execute R programs from Python.

Some problems are more easily solved with Python, others with R. We benefit from Python's capabilities as a general-purpose programming language. We draw on R packages for traditional statistics, time series analysis, multivariate methods, statistical graphics, and handling missing data. Accordingly, this book includes Python and R code examples and represents a dual-language guide to web and network data science.

Browser usage has changed dramatically over the years, with the rise of Google Chrome and the decline of Microsoft Internet Explorer (IE). Table 1.1 and figure 1.1 show worldwide browser usage statistics from October 2008 through October 2014. It is good to have some familiarity with browsers and the tools they provide for examining the text elements and structure of web pages.

The challenge of "big data," as they are sometimes called, is not so much the volume of data. It is that these data arise from sources poorly understood, in particular the web and social media. Data are everywhere on the web. We need to find our way to the relevant data and obtain those data in an efficient manner.

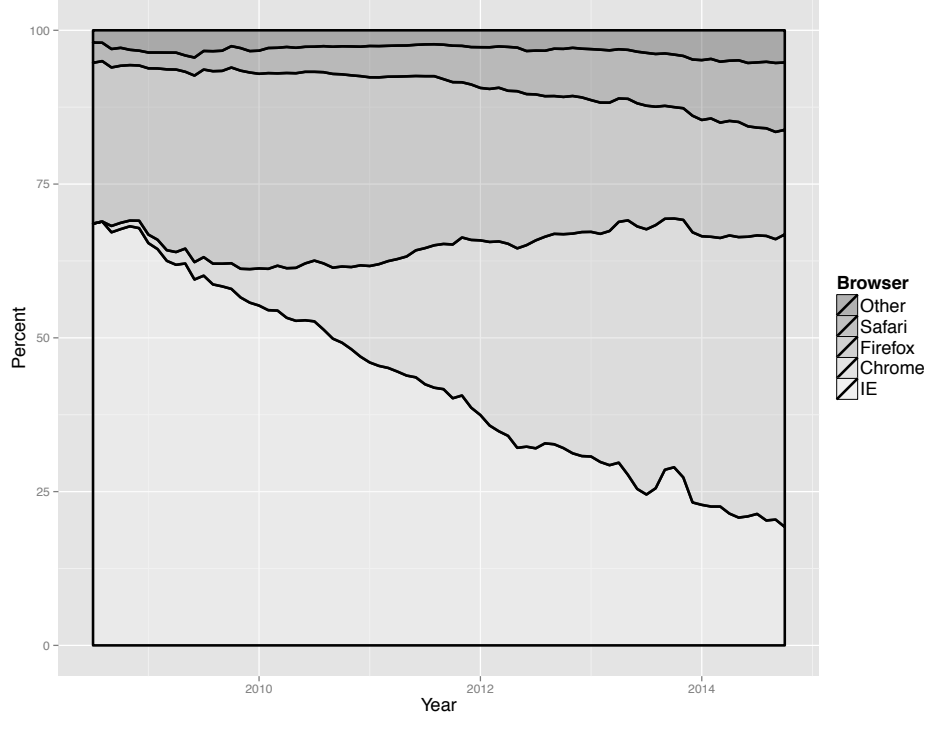
Application programming interfaces (APIs) are one way to gather data from the web, and Russell (2014) provides a useful review of social media APIs. Unfortunately, APIs have syntax, parameters, and authorization codes that can change at the whim of the data providers. We employ a different ap-

Table 1.1. *Worldwide Web Browser Usage Percentages (2008–2014)*

Year	IE	Chrome	Firefox	Safari	Other
2008	67.68	1.02	25.54	2.91	2.85
2009	57.96	4.17	31.82	3.47	2.58
2010	49.21	12.39	31.24	4.56	2.60
2011	40.18	25.00	26.39	5.93	2.50
2012	32.08	34.77	22.32	7.81	3.02
2013	28.96	40.44	18.11	8.54	3.95
2014	19.25	47.57	17.00	10.95	5.23

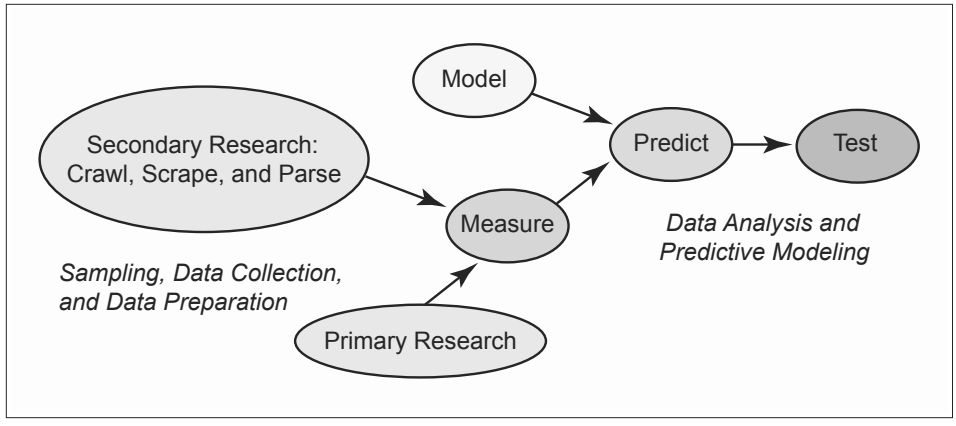
Data obtained from StatCounter (2014).

Figure 1.1. *Worldwide Web Browser Usage (July 2008 through October 2014)*



Data obtained from StatCounter (2014).

Figure 1.2. *Web and Network Data Science: Online Research Process*



proach, focusing on general purpose technologies for automated data acquisition from the web.

Figure 1.2 summarizes the online research process. Sampling, data collection, and data preparation consume much of our time, with secondary research dominating primary research. Online secondary research draws from existing web data. We review secondary research methods in chapter three and use them in many subsequent chapters. Primary research online is facilitated by the web. We cover these methods in appendix B.

The domain of web and network data science is large. There are many questions to address, as shown in the list to follow.

- **Website design and user behavior.** Web analytics, as it is understood by many, involves collecting, storing, and analyzing data from users of a particular website. There are many questions to be addressed. How shall we design and implement websites (for ease of use, visibility, marketing communication, good performance in search, and/or conversion of visits to sales)? How can we gather information from the web efficiently? How can we convert semi-structured and unstructured text into data for input to analysis and modeling? What kinds of website and social media measures make the most sense? Who are the users of a website, and how do they use it? How well does a website do in serving user needs? How well does a website do compared with other websites?

- **Network paths and communication.** Web and network data science is much more than website analytics. We look at each website in the context of others on the web. We think in terms of networks—information nodes connected to one another, and users communicating with one another. What is the shortest, fastest, or lowest cost path between two locations? What is the fastest way to spread a message across a network? Which activities are on the critical path to completing a project? How long must we wait for a response from the server?
- **Communities and influence.** Social media provide a glimpse of electronic social networks in action. Here we have the questions of social network analysis. Are there identifiable groups of people in this community? Who are the key players, the most important people in a group? Who are the people with prestige, influence, or power? Who is best positioned to be the leader of a group?
- **Individual and group behavior.** As data scientists, we are often called on to go beyond description and provide predictions about future behavior or performance. So we have more questions to address. Will this person buy the product, given his/her connections with other buyers or non-buyers? Will this person vote for the candidate, given his/her connections with other voters? Given the motives of individuals, what can we predict for the group? Given growth in the network in the past, what can we expect for the future?
- **Information and networks.** As an information resource, the web is unparalleled. Additional questions arise about the nature of online information. Which are the best websites for getting information about a particular topic? Who are the most credible sources of information? How shall we characterize a domain of knowledge? How can we use the web to obtain competitive intelligence? How can we utilize web-based information as a database for answering questions (domain-specific and general questions)?

This book is designed to provide an overview of the domain of web and network data science. We illustrate measurement and modeling techniques for answering many questions, and we cite resources for additional learning. Some of the techniques may be regarded as basic, others advanced. All are important to the work of data science.

Some say that data science is the new statistics. And in a world dominated by data, data science is beginning to look like the new business and the new IT as well. Nowhere is this more apparent than when working on web and network problems. With unlimited data mediated and distributed through the web, there is certainly enough to keep us busy for a long time.

To begin the programming portion of the book, exhibit 1.1 lists a Python program for exploring web browser usage statistics. Exhibit 1.2 shows the corresponding R program and draws on graphics software from Wickham and Chang (2014).

Exhibit 1.1. *Analysis of Browser Usage (Python)*

```
# Analysis of Browser Usage (Python)

# prepare for Python version 3x features and functions
from __future__ import division, print_function

# import packages for data analysis
import pandas as pd  # data structures for time series analysis
import datetime  # date manipulation
import matplotlib.pyplot as plt

# browser usage data from StatCounter Global Stats
# retrieved from the World Wide Web, October 21, 2014:
# \url{http://gs.statcounter.com/#browser-ww-monthly-200807-201410
# read in comma-delimited text file
browser_usage = pd.read_csv('browser_usage_2008_2014.csv')
# examine the data frame object
print(browser_usage.shape)
print(browser_usage.head())

# identify date fields as dates with apply and lambda function
browser_usage['Date'] = \
    browser_usage['Date']\
    .apply(lambda d: datetime.datetime.strptime(str(d), '%Y-%m'))
# define Other category
browser_usage['Other'] = 100 -\
    browser_usage['IE'] - browser_usage['Chrome'] -\
    browser_usage['Firefox'] - browser_usage['Safari']

# examine selected columns of the data frame object
selected_browser_usage = pd.DataFrame(browser_usage,\
    columns = ['Date', 'IE', 'Chrome', 'Firefox', 'Safari', 'Other'])
print(selected_browser_usage.shape)
print(selected_browser_usage.head())

# create multiple time series plot
selected_browser_usage.plot(subplots = True,  \
     sharex = True, sharey = True, style = 'k-')
plt.legend(loc = 'best')
plt.xlabel('')
plt.savefig('fig_browser_mts_Python.pdf',
    bbox_inches = 'tight', dpi=None, facecolor='w', edgecolor='b',
    orientation='portrait', papertype=None, format=None,
    transparent=True, pad_inches=0.25, frameon=None)

# Suggestions for the student:
# Explore alternative visualizations of these data.
# Try the Python package ggplot to reproduce R graphics.
# Explore time series for other software and systems.
```

Exhibit 1.2. *Analysis of Browser Usage (R)*

```
# Analysis of Browser Usage (R)

# begin by installing necessary package ggplot2

# load package into the workspace for this program
library(ggplot2)  # grammar of graphics plotting

# browser usage data from StatCounter Global Stats
# retrieved from the World Wide Web, October 21, 2014:
# \url{http://gs.statcounter.com/#browser-ww-monthly-200807-201410
# read in comma-delimited text file
browser_usage <- read.csv("browser_usage_2008_2014.csv")
# examine the data frame object
print(str(browser_usage))
# define Other category
browser_usage$Other <- 100 -
    browser_usage$IE - browser_usage$Chrome -
    browser_usage$Firefox - browser_usage$Safari

# define time series data objects
IE_ts <- ts(browser_usage$IE, start = c(2008, 7), frequency = 12)
Chrome_ts <- ts(browser_usage$Chrome, start = c(2008, 7), frequency = 12)
Firefox_ts <- ts(browser_usage$Firefox, start = c(2008, 7), frequency = 12)
Safari_ts <- ts(browser_usage$Safari, start = c(2008, 7), frequency = 12)
Other_ts <- ts(browser_usage$Other, start = c(2008, 7), frequency = 12)

# create a multiple time series object
browser_mts <- cbind(IE_ts, Chrome_ts, Firefox_ts, Safari_ts, Other_ts)
dimnames(browser_mts)[[2]] <- c("IE", "Chrome", "Firefox", "Safari", "Other")
# plot multiple time series object using standard R graphics
pdf(file="fig_browser_mts_R.pdf",width = 11,height = 8.5)
ts.plot(browser_mts, ylab = "Percent Usage", main="",
    plot.type = "single", col = 1:5)
legend("topright", colnames(browser_mts), col = 1:5,
    lty = 1, cex = 1)
dev.off()

# define Year as numeric with fractional values for months
browser_usage$Year <- as.numeric(time(IE_ts))

# build data frame for plotting a stacked area graph
Browser <- rep("IE", length = nrow(browser_usage))
Percent <- browser_usage$IE
Year <- browser_usage$Year
plotting_data_frame <- data.frame(Browser, Percent, Year)

Browser <- rep("Chrome", length = nrow(browser_usage))
Percent <- browser_usage$Chrome
Year <- browser_usage$Year
plotting_data_frame <- rbind(plotting_data_frame,
    data.frame(Browser, Percent, Year))
```

```
Browser <- rep("Firefox", length = nrow(browser_usage))
Percent <- browser_usage$Firefox
Year <- browser_usage$Year
plotting_data_frame <- rbind(plotting_data_frame,
    data.frame(Browser, Percent, Year))

Browser <- rep("Safari", length = nrow(browser_usage))
Percent <- browser_usage$Safari
Year <- browser_usage$Year
plotting_data_frame <- rbind(plotting_data_frame,
    data.frame(Browser, Percent, Year))

Browser <- rep("Other", length = nrow(browser_usage))
Percent <- browser_usage$Other
Year <- browser_usage$Year
plotting_data_frame <- rbind(plotting_data_frame,
    data.frame(Browser, Percent, Year))

# create ggplot plotting object and plot to external file
pdf(file = "fig_browser_usage_stacked_area_R.pdf", width = 11, height = 8.5)
area_plot <- ggplot(data = plotting_data_frame,
    aes(x = Year, y = Percent, fill = Browser)) +
    geom_area(colour = "black", size = 1, alpha = 0.4) +
    scale_fill_brewer(palette = "Blues",
        breaks = rev(levels(plotting_data_frame$Browser))) +
    theme(legend.text = element_text(size = 15))  +
    theme(legend.title = element_text(size = 15)) +
    theme(axis.title = element_text(size = 15))
print(area_plot)
dev.off()
```

This page intentionally left blank

Index

M

N

O

P

Q

R

S

T

U

V

W

X