Community Experience Distilled

Clean Data

Save time by discovering effortless strategies for cleaning, organizing, and manipulating your data

Megan Squire

[PACKT] PUBLISHING

Clean Data

Save time by discovering effortless strategies for cleaning, organizing, and manipulating your data

Megan Squire

PACKT PUBLISHING

BIRMINGHAM - MUMBAI

Clean Data

First published: May 2015

Production reference: 1190515

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham B3 2PB, UK.

ISBN 978-1-78528-401-4

www.packtpub.com

Credits

Author
Megan Squire

Reviewers
J. Benjamin Cook
Richard A. Denman, Jr.
Oskar Jarczyk

Commissioning Editor
Akram Hussain

Acquisition Editor
Harsha Bharwani

Content Development Editor
Mamata Walkar

Technical Editor
Nilesh Mangnakar

Copy Editors
Puja Lalwani
Aditya Nair
Stuti Srivastava

Project Coordinator
Shipra Chawhan

Proofreaders
Stephen Copestake
Safis Editing

Indexer
Priya Sane

Production Coordinator
Shantanu N. Zagade

Cover Work
Shantanu N. Zagade

About the Author

Megan Squire is a professor of computing sciences at Elon University. She has been collecting and cleaning dirty data for two decades. She is also the leader of `FLOSSmole.org`, a research project to collect data and analyze it in order to learn how free, libre, and open source software is made.

About the Reviewers

J. Benjamin Cook, after studying sociology at the University of Nebraska-Lincoln, earned his master's in computational science and engineering from the Institute of Applied Computational Science at Harvard University. Currently, he is helping build the data science team at Hudl, a sports software company whose mission is to capture and bring value to every moment in sports. When he's not learning about all things data, Ben spends time with his daughters and his beautiful wife, Renee.

Richard A. Denman, Jr. is a senior consultant with Numb3rs and has over 30 years of experience providing services to major companies in the areas of data analytics, data science, optimization, process improvement, and information technology. He has been a member of the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE) for over 25 years. He is also a member of the Institute for Operations Research and the Management Sciences (INFORMS) and the American Society for Quality (ASQ).

I would like to thank my wife, Barbara, my son, Ashford, and my daughter, Addie, for their support in producing this book.

Oskar Jarczyk graduated from Polish-Japanese Academy of Information Technology with an MSc Eng. degree in computer science (major databases). After three years of commercial work, he returned to academia to become a PhD student in the field of social informatics.

His academic work is connected with problems in the category of web intelligence, especially free/libre open-source software (FLOSS) and collaborative innovation networks (COINs). He specializes in analyzing the quality of work in open source software teams of developers that are on the GitHub portal. Together with colleagues from the WikiTeams research team, he coped with the problem of "clean data" on a daily basis while creating datasets in MongoDB and MySQL. They were later used with success for FLOSS scientific analyses in the R and Python language.

In his spare time, Oskar reads books about big data and practices kendo.

www.PacktPub.com

Support files, eBooks, discount offers, and more

For support files and downloads related to your book, please visit `www.PacktPub.com`.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at `www.PacktPub.com` and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at `service@packtpub.com` for more details.

At `www.PacktPub.com`, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.

`https://www2.packtpub.com/books/subscription/packtlib`

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can search, access, and read Packt's entire library of books.

Why subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print, and bookmark content
- On demand and accessible via a web browser

Free access for Packt account holders

If you have an account with Packt at `www.PacktPub.com`, you can use this to access PacktLib today and view 9 entirely free books. Simply use your login credentials for immediate access.

Table of Contents

Preface vii

Chapter 1: Why Do You Need Clean Data? 1

A fresh perspective 2
The data science process 3
Communicating about data cleaning 4
Our data cleaning environment 5
An introductory example 6
Summary 12

Chapter 2: Fundamentals – Formats, Types, and Encodings 13

File formats 13
Text files versus binary files 14
Opening and reading files 14
Peeking inside files 15
Common formats for text files 17
The delimited format 17
Seeing invisible characters 18
Enclosing values to trap errant characters 19
Escaping characters 20
The JSON format 20
The HTML format 23
Archiving and compression 24
Archive files 24
tar 25
Compressed files 25
How to compress files 26
How to uncompress files 26
Which compression program should I use? 28

Data types, nulls, and encodings **29**
Data types 30
Numeric data 30
Dates and time 33
Strings 35
Other data types 36
Converting between data types 36
Data loss 36
Strategies for conversion 37
Type conversion at the SQL level 37
Type conversion at the file level 42
If a null falls in a forest… 46
Zero 47
Empties 48
Null 49
Character encodings 50
Example one – finding multibyte characters in MySQL data 51
Example two – finding the UTF-8 and Latin-1 equivalents of Unicode characters stored in MySQL 53
Example three – handling UTF-8 encoding at the file level 54
Summary **57**
Chapter 3: Workhorses of Clean Data – Spreadsheets and Text Editors **59**
Spreadsheet data cleaning **60**
Text to columns in Excel 60
Splitting strings 64
Concatenating strings 64
Conditional formatting to find unusual values 64
Sorting to find unusual values 65
Importing spreadsheet data into MySQL 66
Text editor data cleaning **68**
Text tweaking 68
The column mode 70
Heavy duty find and replace 70
A word of caution 73
Text sorting and processing duplicates 73
Process Lines Containing 74
An example project **75**
Step one – state the problem 75
Step two – data collection 75
Download the data 76
Get familiar with the data 76

Step three – data cleaning 76
Extracting relevant lines 76
Transform the lines 77
Step four – data analysis 78
Summary 79
Chapter 4: Speaking the Lingua Franca – Data Conversions 81
Quick tool-based conversions 82
Spreadsheet to CSV 82
Spreadsheet to JSON 82
Step one – publish Google spreadsheet to the Web 83
Step two – create the correct URL 83
SQL to CSV or JSON using phpMyAdmin 85
Converting with PHP 87
SQL to JSON using PHP 87
SQL to CSV using PHP 88
JSON to CSV using PHP 89
CSV to JSON using PHP 90
Converting with Python 90
CSV to JSON using Python 91
CSV to JSON using csvkit 92
Python JSON to CSV 92
The example project 93
Step one – download Facebook data as GDF 93
Step two – look at the GDF file format in a text editor 94
Step three – convert the GDF file into JSON 95
Step four – build a D3 diagram 99
Step five – convert data to the Pajek file format 101
Step six – calculate simple network metrics 104
Summary 105
Chapter 5: Collecting and Cleaning Data from the Web 107
Understanding the HTML page structure 108
The line-by-line delimiter model 108
The tree structure model 109
Method one – Python and regular expressions 110
Step one – find and save a Web file for experimenting 111
Step two – look into the file and decide what is worth extracting 111
Step three – write a Python program to pull out the interesting pieces and save them to a CSV file 112
Step four – view the file and make sure it is clean 113
The limitations of parsing HTML using regular expressions 114

Method two – Python and BeautifulSoup **115**
Step one – find and save a file for experimenting 115
Step two – install BeautifulSoup 115
Step three – write a Python program to extract the data 115
Step four – view the file and make sure it is clean 116
Method three – Chrome Scraper **117**
Step one – install the Scraper Chrome extension 117
Step two – collect data from the website 117
Step three – final cleaning on the data columns 120
Example project – Extracting data from e-mail and web forums **121**
The background of the project 121
Part one – cleaning data from Google Groups e-mail 122
Step one – collect the Google Groups messages 122
Step two – extract data from the Google Groups messages 123
Part two – cleaning data from web forums 126
Step one – collect some RSS that points us to HTML files 126
Step two – Extract URLs from RSS; collect and parse HTML 128
Summary **133**
Chapter 6: Cleaning Data in PDF Files **135**
Why is cleaning PDF files difficult? **136**
Try simple solutions first – copying **137**
Our experimental file 137
Step one – try copying out the data we want 138
Step two – try pasting the copied data into a text editor 139
Step three – make a smaller version of the file 140
Another technique to try – pdfMiner **141**
Step one – install pdfMiner 141
Step two – pull text from the PDF file 141
Third choice – Tabula **143**
Step one – download Tabula 144
Step two – run Tabula 144
Step three – direct Tabula to extract the data 145
Step four – copy the data out 145
Step five – more cleaning 146
When all else fails – the fourth technique **146**
Summary **148**
Chapter 7: RDBMS Cleaning Techniques **151**
Getting ready **152**
Step one – download and examine Sentiment140 **152**
Step two – clean for database import **152**
Step three – import the data into MySQL in a single table **153**

Detecting and cleaning abnormalities 155
Creating our table 157
Step four – clean the & character 158
Step five – clean other mystery characters 158
Step six – clean the dates 161
Step seven – separate user mentions, hashtags, and URLs 162
Create some new tables 163
Extract user mentions 164
Extract hashtags 166
Extract URLs 167
Step eight – cleaning for lookup tables 169
Step nine – document what you did 172
Summary 172
Chapter 8: Best Practices for Sharing Your Clean Data 173
Preparing a clean data package 174
A word of caution – Using GitHub to distribute data 176
Documenting your data 177
README files 177
File headers 179
Data models and diagrams 181
Documentation wiki or CMS 183
Setting terms and licenses for your data 184
Common terms of use 184
Creative Commons 185
ODbL and Open Data Commons 186
Publicizing your data 186
Lists of datasets 186
Open Data on Stack Exchange 187
Hackathons 187
Summary 188
Chapter 9: Stack Overflow Project 189
Step one – posing a question about Stack Overflow 190
Step two – collecting and storing the Stack Overflow data 192
Downloading the Stack Overflow data dump 192
Unarchiving the files 193
Creating MySQL tables and loading data 193
Building test tables 195
Step three – cleaning the data 198
Creating the new tables 199
Extracting URLs and populating the new tables 200

Extracting code and populating new tables 202
Step four – analyzing the data 203
Which paste sites are most popular? 204
Which paste sites are popular in questions and which are popular in answers? 205
Do posts contain both URLs to paste sites and source code? 208
Step five – visualizing the data 209
Step six – problem resolution 212
Moving from test tables to full tables 213
Summary 214
Chapter 10: Twitter Project 215
Step one – posing a question about an archive of tweets 216
Step two – collecting the data 217
Download and extract the Ferguson file 217
Create a test version of the file 219
Hydrate the tweet IDs 219
Setting up a Twitter developer account 219
Installing twarc 221
Running twarc 222
Step three – data cleaning 225
Creating database tables 225
Populating the new tables in Python 227
Step four – simple data analysis 229
Step five – visualizing the data 230
Step six – problem resolution 234
Moving this process into full (non-test) tables 235
Summary 236
Index 239

Preface

"*Pray, Mr. Babbage, if you put into the machine the wrong figures, will the right answer come out?*"

– Charles Babbage (1864)

"*Garbage in, garbage out*"

– The United States Internal Revenue Service (1963)

"*There are no clean datasets.*"

– Josh Sullivan, Booz Allen VP in Fortune (2015)

In his 1864 collection of essays, Charles Babbage, the inventor of the first calculating machine, recollects being dumbfounded at the "confusion of ideas" that would prompt someone to assume that a computer could calculate the correct answer despite being given the wrong input. Fast-forward another 100 years, and the tax bureaucracy started patiently explaining "garbage in, garbage out" to express the idea that even for the all-powerful tax collector, computer processing is still dependent on the quality of its input. Fast-forward another 50 years to 2015: a seemingly magical age of machine learning, autocorrect, anticipatory interfaces, and recommendation systems that know me better than I know myself. Yet, all of these helpful algorithms still require high-quality data in order to learn properly in the first place, and we lament "there are no clean datasets".

This book is for anyone who works with data on a regular basis, whether as a data scientist, data journalist, software developer, or something else. The goal is to teach practical strategies to quickly and easily bridge the gap between the data we want and the data we have. We want high-quality, perfect data, but the reality is that most often, our data falls far short. Whether we are plagued with missing data, data in the wrong format, data in the wrong location, or anomalies in the data, the result is often, to paraphrase rapper Notorious B.I.G., "more data, more problems".

Throughout the book, we will envision data cleaning as an important, worthwhile step in the data science process: easily improved, never ignored. Our goal is to reframe data cleaning away from being a dreaded, tedious task that we must slog through in order to get to the *real* work. Instead, armed with a few tried-and-true procedures and tools, we will learn that just like in a kitchen, if you wash your vegetables first, your food will look better, taste better, and be better for you. If you learn a few proper knife skills, your meat will be more succulent and your vegetables will be cooked more evenly. The same way that a great chef will have their favorite knives and culinary traditions, a great data scientist will want to work with the very best data possible and under the very best conditions.

What this book covers

Chapter 1, *Why Do You Need Clean Data?* motivates our quest for clean data by showing the central role of data cleaning in the overall data science process. We follow with a simple example showing some dirty data from a real-world dataset. We weigh the pros and cons of each potential cleaning process, and then we describe how to communicate our cleaning changes to others.

Chapter 2, *Fundamentals – Formats, Types, and Encodings*, sets up some foundational knowledge about file formats, compression, and data types, including missing and empty data and character encodings. Each section has its own examples taken from real-world datasets. This chapter is important because we will rely on knowledge of these basic concepts for the rest of the book.

Chapter 3, *Workhorses of Clean Data – Spreadsheets and Text Editors*, describes how to get the most data cleaning utility out of two common tools: the text editor and the spreadsheet. We will cover simple solutions to common problems, including how to use functions, search and replace, and regular expressions to correct and transform data. At the end of the chapter, we will put our skills to test using both of these tools to clean some real-world data regarding universities.

Chapter 4, *Speaking the Lingua Franca – Data Conversions*, focuses on converting data from one format to another. This is one of the most important data cleaning tasks, and it is useful to have a variety of tools at hand to easily complete this task. We first proceed through each of the different conversions step by step, including back and forth between common formats such as comma-separated values (CSV), JSON, and SQL. To put our new data conversion skills into practice, we complete a project where we download a Facebook friend network and convert it into a few different formats so that we can visualize its shape.

Chapter 5, *Collecting and Cleaning Data from the Web*, describes three different ways to clean data found inside HTML pages. This chapter presents three popular tools to pull data elements from within marked-up text, and it also provides the conceptual foundation to understand other methods besides the specific tools shown here. As our project for this chapter, we build a set of cleaning procedures to pull data from web-based discussion forums.

Chapter 6, *Cleaning Data in PDF Files*, introduces several ways to meet this most stubborn and all-too-common challenge for data cleaners: extracting data that has been stored in Adobe's Portable Document Format (PDF) files. We first examine low-cost tools to accomplish this task, then we try a few low-barrier-to-entry tools, and finally, we experiment with the Adobe non-free software itself. As always, we use real-world data for our experiments, and this provides a wealth of experience as we learn to work through problems as they arise.

Chapter 7, *RDBMS Cleaning Techniques*, uses a publicly available dataset of tweets to demonstrate numerous strategies to clean data stored in a relational database. The database shown is MySQL, but many of the concepts, including regular-expression based text extraction and anomaly detection, are readily applicable to other storage systems as well.

Chapter 8, *Best Practices for Sharing Your Clean Data*, describes some strategies to make your hard work as easy for others to use as possible. Even if you never plan to share your data with anyone else, the strategies in this chapter will help you stay organized in your own work, saving you time in the future. This chapter describes how to create the ideal data package in a variety of formats, how to document your data, how to choose and attach a license to your data, and also how to publicize your data so that it can live on if you choose.

Chapter 9, *Stack Overflow Project*, guides you through a full-length project using a real-world dataset. We start by posing a set of authentic questions that we can answer about that dataset. In answering this set of questions, we will complete the entire data science process introduced in *Chapter 1*, *Why Do You Need Clean Data?* and we will put into practice many of the cleaning processes we learned in the previous chapters. In addition, because this dataset is so large, we will introduce a few new techniques to deal with the creation of test datasets.

Chapter 10, *Twitter Project*, is a full-length project that shows how to perform one of the hottest and fastest-changing data collection and cleaning tasks out there right now: mining Twitter. We will show how to find and collect an existing archive of publicly available tweets on a real-world current event while adhering to legal restrictions on the usage of the Twitter service. We will answer a simple question about the dataset while learning how to clean and extract data from JSON, the most popular format in use right now with API-accessible web data. Finally, we will design a simple data model for long-term storage of the extracted and cleaned data and show how to generate some simple visualizations.

What you need for this book

To complete the projects in this book, you will need the following tools:

- A web browser, Internet access, and a modern operating system. The browser and operating system should not matter, but access to a command-line terminal window is ideal (for example, the Terminal application in OS X). In *Chapter 5*, *Collecting and Cleaning Data from the Web*, one of the three activities relies on a browser-based utility that runs in the Chrome browser, so keep this in mind if you would like to complete this activity.
- A text editor, such as Text Wrangler for Mac OSX or Notepad++ for Windows. Some integrated development environments (IDEs, such as Eclipse) can also be used as a text editor, but they typically have many features you will not need.
- A spreadsheet application, such as Microsoft Excel or Google Spreadsheets. When possible, generic examples are provided that can work on either of these tools, but in some cases, one or the other is required.
- A Python development environment and the ability to install Python libraries. I recommend the Enthought Canopy Python environment, which is available here: `https://www.enthought.com/products/canopy/`.
- A MySQL 5.5+ server installed and running.
- A web server (running any server software) and PHP 5+ installed.
- A MySQL client interface, either the command-line interface, MySQL Workbench, or phpMyAdmin (if you have PHP running).

Who this book is for

If you are reading this book, I guess you are probably in one of two groups. One group is the group of data scientists who already spend a lot of time cleaning data, but you want to get better at it. You are probably frustrated with the tedium of data cleaning, and you are looking for ways to speed it up, become more efficient, or just use different tools to get the job done. In our kitchen metaphor, you are the chef who just needs to brush up on a few knife skills.

The other group is made up of people doing the data science work but who never really cared about data cleaning before. But now, you are starting to think that maybe your results might actually get better if you had a cleaning process. Maybe the old adage "garbage in, garbage out" is starting to feel a little too real. Maybe you are interested in sharing your data with others, but you do not feel confident about the quality of the datasets you are producing. With this book, you will gain enough confidence to "cook in public" by learning a few tricks and creating new habits that will ensure a tidy, clean data science environment.

Either way, this book will help you reframe data cleaning away from being a symbol of drudgery and toward being your hallmark of quality, good taste, style, and efficiency. You should probably have a bit of programming background, but you do not have to be great at it. As with most data science projects, a willingness to learn and experiment as well as a healthy sense of curiosity and a keen attention to detail are all very important and valued.

Conventions

In this book, you will find a number of text styles that distinguish between different kinds of information. Here are some examples of these styles and an explanation of their meaning.

Code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles are shown as follows: " The issue is that `open()` is not prepared to handle UTF-8 characters."

A block of code is set as follows:

```
for tweet in stream:
    encoded_tweet = tweet['text'].encode('ascii','ignore')
    print counter, "-", encoded_tweet[0:10]
    f.write(encoded_tweet)
```