

# A Programmer's Guide to Data Mining



**The Ancient Art of the Numerati**

**Ron Zacharski**

# A Programmer's Guide to Data Mining: The Ancient Art of the Numerati

[www.guidetodatamining.com](http://www.guidetodatamining.com)

by Ron Zacharski

Creative Commons Attribution Noncommercial 3.0 license

Attribution information for all photographs is available on the website.

Thanks to ...

my wife Cheryl



Roper

my son Adam



Roz and Bodhi



also a huge thanks to all the photographers who put their work in the Creative Commons

# Preface



*If you continue this simple practice every day, you will obtain some wonderful power. Before you attain it, it is something wonderful, but after you attain it, it is nothing special.*

Shunryu Suzuki  
Zen Mind, Beginner's Mind.

Before you work through this book you might think that systems like Pandora, Amazon's recommendations, and automatic data mining for terrorists, must be very complex and the math behind the algorithms must be extremely complex requiring a PhD to understand. You might think the people who work on developing these systems are like rocket scientists. One goal I have for this book is to pull back this curtain of complexity and show some of the rudimentary methods involved. Granted there are super-smart people at Google, the National Security Agency and elsewhere developing amazingly complex algorithms, but for the most part data mining relies on easy-to-understand principles. Before you start the book you might think data mining is pretty amazing stuff. By the end of the book, I hope you will be able to say nothing special.

The Japanese characters above, Shoshin, represent the concept of Beginner's Mind—the idea of having an open mind that is eager to explore possibilities. Most of us have heard some version of the following story (possibly from Bruce Lee's *Enter the Dragon*). A professor is seeking enlightenment and goes to a wise monk for spiritual direction. The professor dominates the discussion outlining everything he has learned in his life and summarizing papers he has written. The monk asks tea? and begins to pour tea into the professor's cup. And continues to pour, and continues to pour, until the tea over pours the teacup, the table, and spills onto the floor. *What are you doing?* the professor shouts. *Pouring tea* the monk says and continues: *Your mind is like this teacup. It is so filled with ideas that nothing else will go in. You must empty your mind before we can begin.*

To me, the best programmers are empty cups, who constantly explore new technology (noSQL, node-js, whatever) with open minds. Mediocre programmers have surrounded their minds with cities of delusion—C++ is good, Java is bad, PHP is the only way to do web programming, MySQL is the only database to consider. My hope is that you will find some of the ideas in this book valuable and I ask that you keep a beginner's mind when reading it. As Shunryu Suzuki says:

*In the beginner's mind there are many possibilities,*

*In the expert's mind there are few.*

## Chapter 1 The Intro

# Intro to data mining & how to use this book

Imagine life in a small American town 150 years ago. Everyone knows one another. A crate of fabric arrives at the general store. The clerk notices that the pattern of a particular bolt would highly appeal to Mrs. Clancey because he knows that she likes bright floral patterns and makes a mental note to show it to her next time she comes to the store. Chow Winkler mentions to Mr. Wilson, the saloon keeper, that he is thinking of selling his spare Remington rifle. Mr. Wilson mentions that information to Bud Barclay, who he knows is looking for a quality rifle. Sheriff Valquez and his deputies know that Lee Pye is someone to keep an eye on as he likes to drink, has a short temper, and is strong. Life in a small town 100 years ago was all about connections.



People knew your likes and dislikes, your health, the state of your marriage. For better or worse, it was a personalized experience. And this highly personalized life in the community was true throughout most of the world.

Let's jump ahead one hundred years to the 1960s. Personalized interactions are less likely but they are still present. A regular coming into a local bookstore might be greeted with "The new James Michener is in"-- the clerk knowing that the regular loves James Michener books. Or the clerk might recommend to the regular *The Conscience of a Conservative* by Barry Goldwater, because the clerk knows the regular is a staunch conservative. A regular customer comes into a diner and the waitress says "The usual?"

Even today there are pockets of personalization. I go to my local coffee shop in Mesilla and the barista says "A venti latte with an extra shot?" knowing that is what I get every morning. I take my standard poodle to the groomers and the groomer doesn't need to ask what style of clip I want. She knows I like the no frills sports clip with the German style ears.

But things have changed since the small towns of 100 years ago. Large grocery stores and big box stores replaced neighborhood grocers and other merchants. At the start of this change choices were limited. Henry Ford once said "Any customer can have a car painted any color that he wants so long as it is black." The record store carried a limited number of records; the bookstore carried a limited number of books. Want ice cream? The choices were vanilla, chocolate, and maybe strawberry. Want a washing machine? In 1950 you had two choices at the local Sears: the standard model for \$55 or the deluxe for \$95.

## Welcome to the 21<sup>st</sup> century

In the 21st century those limited choices are a thing of the past. I want to buy some music? iTunes has some 11 million tracks to choose from. 11 million! They have sold 16 billion tracks as of October 2011. I need more choices? I can go to Spotify which has over 15 million songs.

I want to buy a book? Amazon has over 2 million titles to choose from.



I want to watch a video? There are plenty of choices:



I want to buy a laptop? When I type in *laptop* into the Amazon search box I get 3,811 results

I type in *rice cooker* and get over 1,000 possibilities.

In the near future there will be even more choice—billions of music tracks online—a wide variety of video—products that can be customized with 3D printing.



## Finding Relevant Stuff

The problem is finding relevant stuff. Amid all those 11 million tracks on iTunes, there are probably quite a number that I will absolutely love, but how do I find them. I want to watch a streaming movie from Netflix tonight, what should I watch. I want to download a movie using P2P, but which movie. And the problem is getting worse. Every minute terabytes of media are added to the net. Every minute 100 new files are available on usenet. Every minute 24 hours of video is uploaded to YouTube. Every hour 180 new books are published. Every day there are more and more options of stuff to buy in the real world. It gets more and more difficult to find the relevant stuff in this ocean of possibilities.

If you are a producer of media—say Zee Avi from Malaysia—the danger isn't someone downloading your music illegally—the danger is obscurity.



## But how to find stuff?

Years ago, in that small town, our **friends** helped us find stuff. That bolt of fabric that would be perfect for us; that new novel at the bookstore; that new 33 1/3 LP at the record store. Even today we rely on friends to help us find some relevant stuff.

We used **experts** to help us find stuff. Years ago Consumer Reports could evaluate all the washing machines sold—all 20 of them—or all the rice cookers sold-- all 10 of them and make recommendations. Today there are hundreds of different rice cookers available on Amazon and it is unlikely that a single expert source can rate all of them. Years ago, Roger Ebert would review virtually all the movies available. Today about 25,000 movies are made each year worldwide. Plus, we now have access to video from a variety of sources. Roger Ebert, or any single expert, cannot review all the movies that are available to us.

We also use **the thing itself** to help us find stuff. For example, I owned a Sears washing machine that lasted 30 years, I am going to buy another Sears washing machine. I liked one album by the Beatles—I will buy another thinking chances are good I will like that too.

**These methods of finding relevant stuff—friends, experts, the thing itself—are still present today but we need some computational help to transform them into the 21st century where we have billions of choices.**

These methods of finding relevant stuff—friends, experts, the thing itself—are still present today but we need some computational help to transform them into the 21st century where we have billions of choices. In this book we will explore methods of aggregating people's likes and dislikes, their purchasing history, and other data—exploiting the power of social net (friends)—to help us mine for relevant stuff. We will examine methods that use attributes of the thing itself. For example, I like the band Phoenix. The system might know attributes of Phoenix—that it uses electric rock instrumentation, has punk influences, has a subtle use of vocal harmony. It might recommend to me a similar band that has similar attributes, for example, The Strokes.

### **It's just not stuff...**

Data mining is just not about recommending stuff to us, or having merchants sell more stuff. Consider these examples.

The mayor of that small town of 100 years ago, knew everybody. When he ran for re-election he knew how to tailor what he said to each individual.



Martha, I know you are interested in schools and I will do everything in my power to bring another teacher to town.

John, how is your bakery doing? I promise to get more parking in your area of downtown.

My father belonged to the United Auto Workers' Union. Around election time I remember the union representative coming to our house to remind my father what candidates to vote for:



Frank Zeidler was the Socialist mayor of Milwaukee from 1948 to 1960.

*Hey Syl, how are the wife and kids? ... Now let me tell you why you should vote for Frank Zeidler, the Socialist candidate for mayor...*

This individualized political message changed to the homogenous ads during the rise of television. Everyone got the exact same message. A good example of this is the famous Daisy television ad in support of

Lyndon Johnson ( a young girl pulling petals off a daisy while a nuclear bomb goes off in the background). Now, with elections determined by small margins and the growing use of data mining, individualization has returned. You are interested in a women's right to chose? You might get a robo-call directed at that very issue.

**T**he sheriff of that small town knew who the trouble makers were. Now, threats seem to be hidden, terrorists can be anywhere. On October 11, 2001 the US government passed the USA Patriot Act (short for **U**niting and **S**trengthening **A**merica by **P**roviding **A**ppropriate **T**ools **R**equired to **I**ntercept and **O**bstruct **T**errorism). In part this bill enables investigators to obtain records for a variety of sources including libraries (what books we read), hotels (who stayed where and for how long), credit card companies, toll roads registering that we passed by. For the most part the government uses private companies to keep data on us. Companies like Seisint have data on almost all of us, photos of us, where we live, what we drive, our income, our buying behavior, our friends. Seisint owns supercomputers that use data mining techniques to make predictions about people. Their product by the way is called...



## The Matrix.



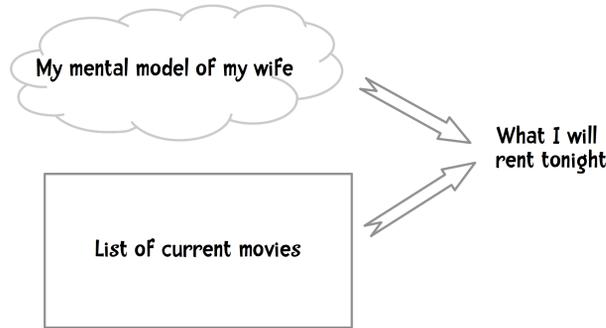
## Data Mining Extends what we already do!

Stephen Baker begins his book *The Numerati* this way:

Imagine you are in a café, perhaps the noisy one I'm sitting in at this moment. A young women at a table to your right is typing on her laptop. You turn your head and look at her screen. She surfs the Internet. You watch.

Hours pass. She reads an online paper. You notice that she reads three articles about China. She scouts movies for Friday night and watches the trailer for Kung Fu Panda. She clicks on an ad that promises to connect her to old high school classmates. You sit there taking notes. With each passing minute, you're learning more about her. Now imagine that you could watch 150 million people surfing at the same time.

Data mining is focused on finding patterns in data. At the small scale, we are expert at building mental models and finding patterns. I want to watch a movie tonight with my wife. I have a mental model of what she likes. I know she dislikes violent movies (she didn't like District 9 for that reason). She likes movies by Charlie Kaufman. I can use that mental model I have of her movie preferences to predict what movies she may or may not like.



A friend is visiting from Europe. I know she is a vegetarian and I can use that information to predict she would not like the local rib joint. People are good at making models and making predictions. Data mining expands this ability and enables us to handle large quantities of information—the 150 million people in the Baker quote above. It enables the Pandora Music Service to tailor a music station to your specific musical preferences. It enables Netflix to make specific personalized movie recommendations for you.

---

## Tera-mining is not something from Starcraft II

At the end of the 20th century a million word data set was considered large. When I was a graduate student in the 1990s (yes, I am that ancient) I worked as a programmer for a year on the Greek New Testament. It's only around 200,000 words but the analysis was too large to fit into the mainframe's memory necessitating spooling results off to magnetic tapes, which I had to request to be mounted.

The book resulting from this work is the Analytical Greek New Testament by Timothy and Barbara Friberg (available on Amazon). I was just one of three programmers on this project done at the University of Minnesota.

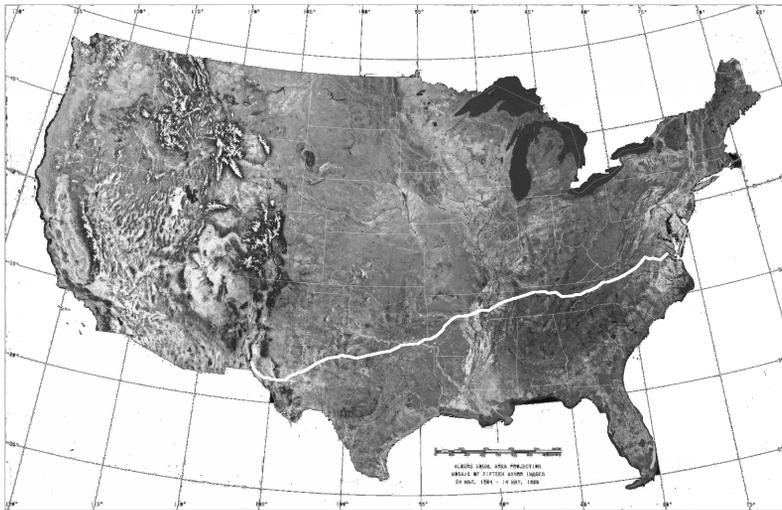


Today it is not unusual to be doing data mining on terabytes of information. Google has over 5 petabytes (that's 5,000 terabytes) of web data. In 2006 Google released a dataset to the research community based on one trillion words. The National Security Agency has call records for trillions of phone calls. Acxiom, a company that collects information (credit card purchases, telephone records, medical records, car registrations, etc) on 200 million adults in the US, has amassed over 1 petabyte of data.



a 1 petabyte server shipping container

Robert O'Harrow, Jr., author of No Place to Hide, in an effort to help us grasp how much information is 1 petabyte says it is the equivalent of 50,000 miles of stacked King James Bibles. I frequently drive 2,000 between New Mexico and Virginia. When I try to imagine bibles stacked along the entire way that seems like an unbelievable amount of data.



New Mexico to Virginia

The Library of Congress has around 20 terabytes of text. You could store the entire collection of the Library of Congress on a few thousand dollar's worth of hard drives! In contrast, Walmart has over 570 terabytes of data. All this data just doesn't sit there—it is constantly being mined, new associations made, patterns identified. Tera-mining.

Throughout this book we will be dealing with small datasets. It's good thing. We don't want our algorithm to run for a week only to discover we have some error in our logic. The biggest dataset we will use is under 100MB; the smallest just tens of lines of data.

## The format of the book.

This book follows a learn-by-doing approach. Instead of passively reading the book, I encourage you to work through the exercises and experiment with the Python code I provide. Experimenting around, code hacking, and trying out methods with different data sets is the key to really gaining an understanding for the techniques.



I try to strike a balance between hands-on, nuts-and-bolts discussion of Python data mining code that you can use and modify, and the theory behind the data mining techniques. To try to prevent the brain freeze associated with reading theory, math, and Python code, I tried to stimulate a different part of your brain by adding drawings and pictures.

Peter Norvig, Director of Research at Google, had this to say in his great Udacity course. *Design of a Computer Program*:

**"I'll show you and discuss my solution. It's important to note, there is more than one way to approach a problem. And I don't mean that my solution is the ONLY way or the BEST way. My solutions are there to help you learn a style and some techniques for programming. If you solve problems a different way, that's fine. Good for you.**

**All the learning that goes on happens inside of your head. Not inside of my head. So what's important is that you understand the relation between your code and my code, that you get the right answer by writing out the solution yourself and then you can examine my code and maybe pick out some pointers and techniques that you can use later."**

**I couldn't agree more!**



This book is not a comprehensive textbook on data mining techniques. There are textbooks, like *Introduction to Data Mining* by Pang-Ning Tan, Michael Steinbach, and Vipin Kumar that provide significantly better coverage of data mining methods and provide more in-depth analysis of the mathematic underpinnings of these methods. This book—the one you are holding—is intended more as a quick, gritty, hands-on introduction designed to give you a basic foundation of data mining techniques. Later, you can pick up a more comprehensive book to fill in any gaps that you wish.

Part of the usefulness of this book is the accompanying Python code and the datasets. I think the inclusion of both these make it easier for the learner to understand key concepts, but at the same time, not shoe-horn the learner into a scripted exploration.

## What will you be able to do when you finish this book?

When you finish this book you will be able to design and implement recommendation systems for websites using Python or any language you know. For example, when you look at a product on Amazon, or a tune on Pandora, you are presented with a list of recommendations (You might also like ...). You will learn how to develop such systems. In addition, the book should provide you with the necessary vocabulary to enable you to work in development teams on data mining efforts.

As part of this goal, this book should help shed the mystery of recommendation systems, terrorist identification systems, and other data mining systems. You should at least have a rough idea of how they work.

## Why – why does this matter?

Why should you use your time reading (and working through) this book on data mining? At the beginning of this chapter I gave examples related to the importance of data mining. The summary of that section would go as follows. There's lots of stuff out there (movies, music, books, rice cookers). There's going to be a huge growth in the amount of stuff out there. The problem with having all this stuff available is finding the stuff that is relevant to us. Of all the movies out there, what movie should I watch. What's the next book I should read? This problem of identifying relevant stuff is what data mining is about. Most websites will have some component dealing with 'finding stuff'. In addition to the movies, music, books, and rice cookers mentioned above, you might want recommendations about what friends to follow. How about a personalized newspaper showing just the news you are most interested in? If you are a programmer, particularly a web developer, it would be useful to know data mining techniques.

Okay, so you can see the reason to devote some of your time to learning data mining, but why this book? There are books that give you a non-technical overview of data mining. They are a quick read, entertaining, inexpensive, and can be read late at night (no hairy technical bits). A great example of this is *The Numerati* by Stephen Baker. I recommend this book—I listened to the audio version of it while driving between Virginia and New Mexico. It was engrossing. On the other extreme are college textbooks on data mining. They are

comprehensive and provide an in-depth analysis of data mining theory and practice. Again, I recommend books in this category. I wrote this book to fill a gap. It's a book designed for people who love to program—hackers.



The book is intended to be read at a computer so the reader can participate and mess with code.

**Eeeks!**

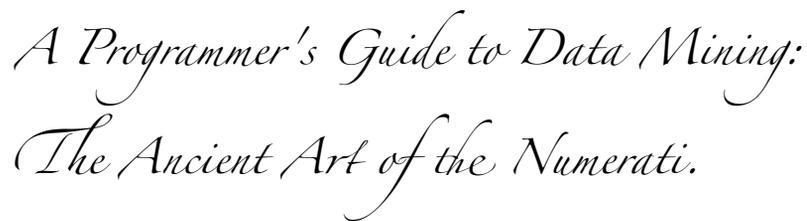
The book has math formulas but I try to explain them in a way that is intelligible to average programmers, who may have forgotten a hunk of the math they took in college.

$$s(i, j) = \frac{\sum_{u \in U} (R_{u,i} - \bar{R}_u)(R_{u,j} - \bar{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \bar{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \bar{R}_u)^2}}$$

If that doesn't convince you, this book is also free (as in no cost) and free as in you can share it.

## What's with the 'Ancient Art of the Numerati' part of the title

In June of 2010 I was trying to come up with a title for this book. I like clever titles, but unfortunately, I have no talent in the area. I recently published a paper titled *Linguistic Dumpster Diving: Geographical Classification of Arabic Text* (yep, a data mining paper). I like the title and it is clever because it fits with the content of the paper, but I have to confess my wife came up with the title. I co-wrote a paper *Mood and Modality: Out of the theory and into the fray*. My co-author Marjorie McShane came up with the title. Anyway, back to June, 2010. All my clever title ideas were so vague that you wouldn't have a clue what the book was about. I finally settled on *A Programmer's Guide to Data Mining* as part of the title. I believe that bit is a concise description of the content of the book—I intend the book be a guide for the working programmer. You might wonder what is the meaning of the part after the colon:



*A Programmer's Guide to Data Mining:  
The Ancient Art of the Numerati.*

The Numerati is a term coined by Stephen Baker. Each one of us generates an amazing amount of digital data everyday. credit card purchases, Twitter posts, Gowalla posts, Foursquare check-ins, cell phone calls, email messages, text messages, etc.

You get up. The Matrix knows you boarded the subway at the Foggy Bottom Station at 7:10 and departed the Westside Station at 7:32. The Matrix knows you got a venti latte and a blueberry scone at the Starbucks on 5th and Union at 7:45; you used Gowalla to check-in at work at 8:05; you made an Amazon purchase for the P90X Extreme Home Fitness Workout Program 13 DVD set and a chin-up bar at 9:35; you had lunch at the Golden Falafel.

Stephen Baker writes:

The only folks who can make sense of the data we create are crack mathematicians, computer scientists, and engineers. What will these Numerati learn about us as they run us into dizzying combinations of numbers? First they need to find us. Say you're a potential SUV shopper in the northern suburbs of New York, or a churchgoing, antiabortion Democrat in Albuquerque. Maybe you're a Java programmer ready to relocate to Hyderabad, or a jazz-loving, Chianti-sipping Sagittarius looking for walks in the country and snuggles by the fireplace in Stockholm, or—heaven help us—maybe you're eager to strap bombs to your waist and climb onto a bus. Whatever you are—and each of us is a lot of things—companies and governments want to identify and locate you.

Baker

As you can probably guess, I like this term *Numerati* and Stephen Baker's description of it.

