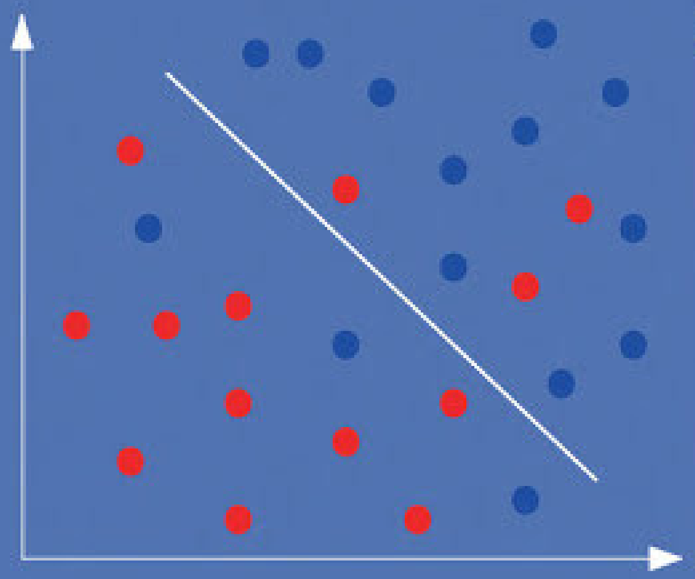
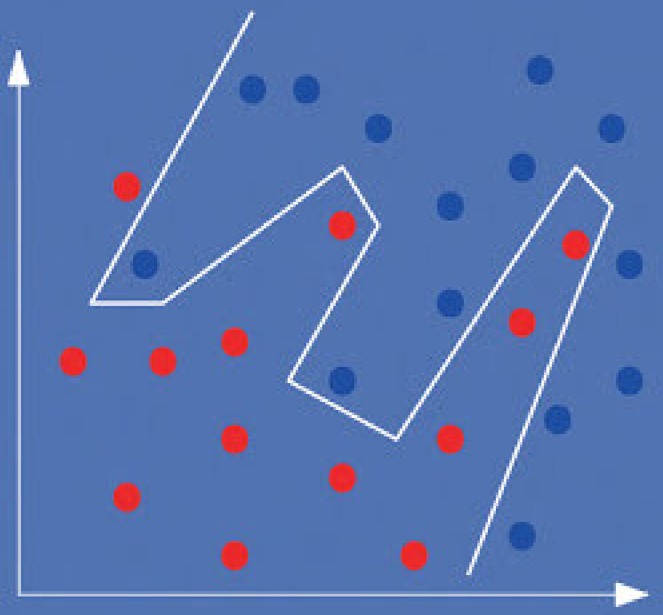
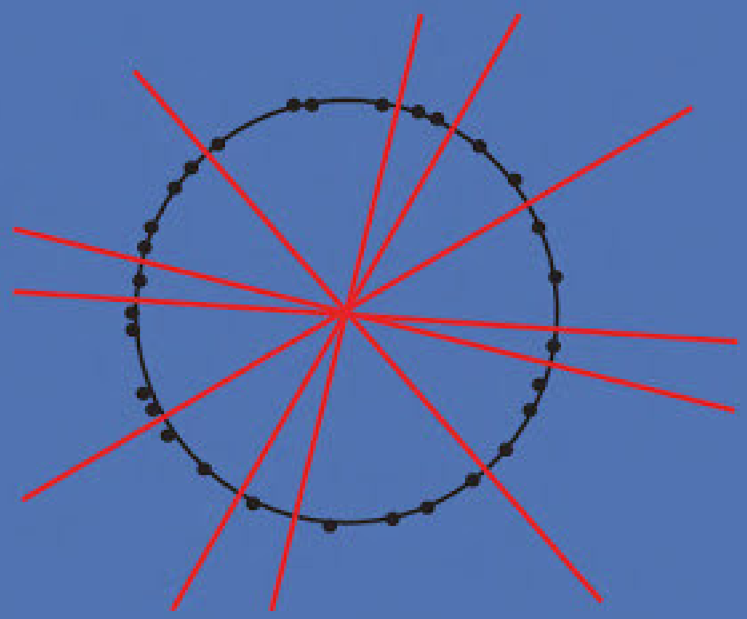
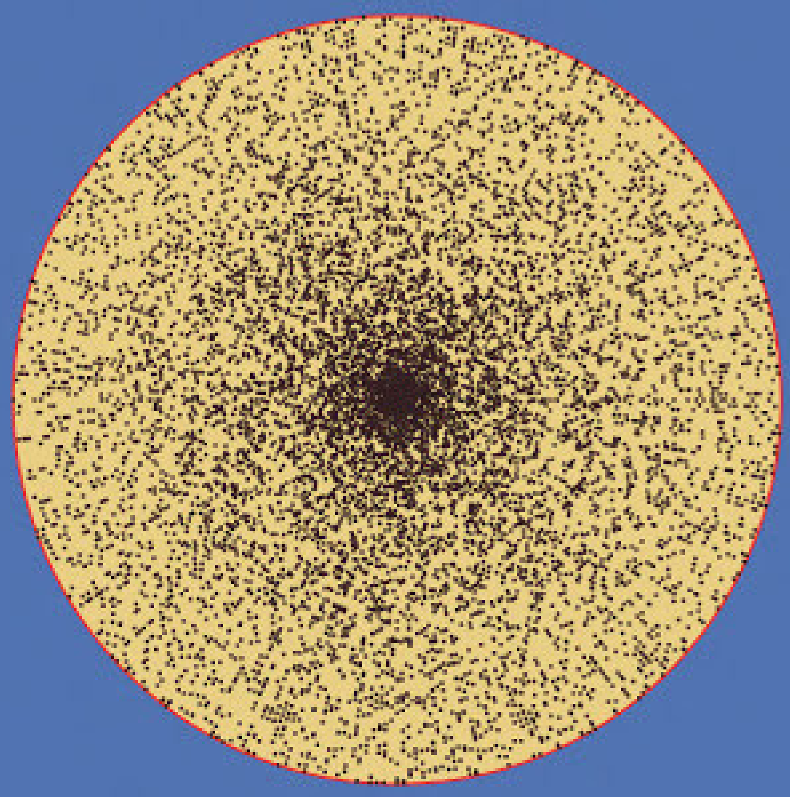
TEXTS IN COMPUTER SCIENCE

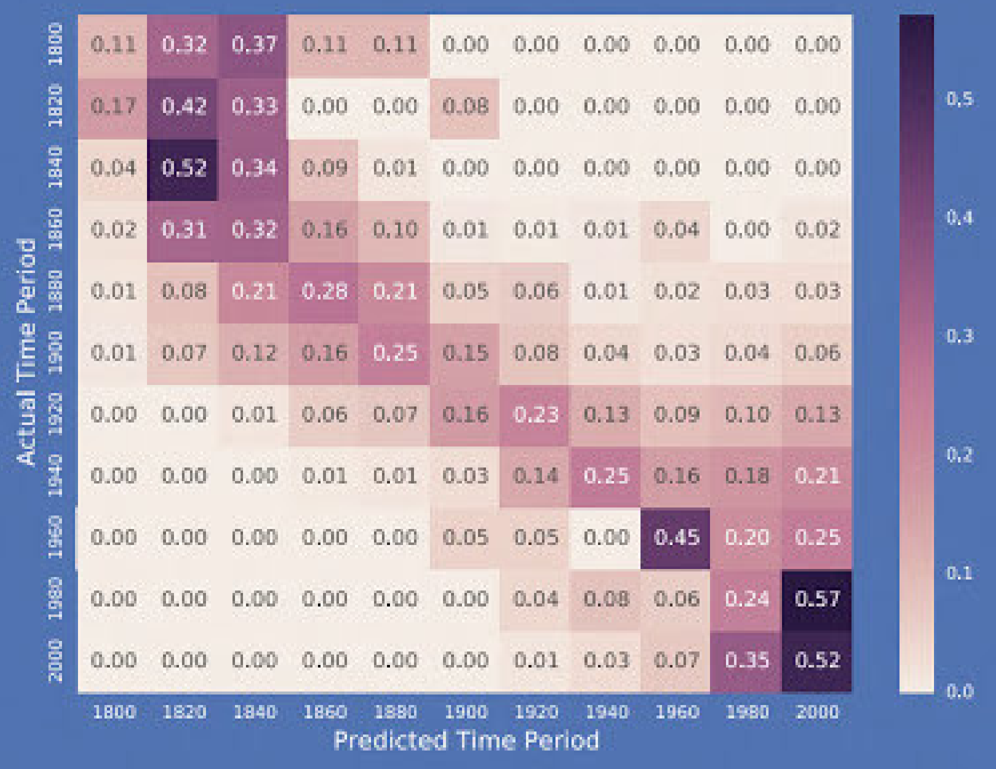
THE Data Science Design MANUAL

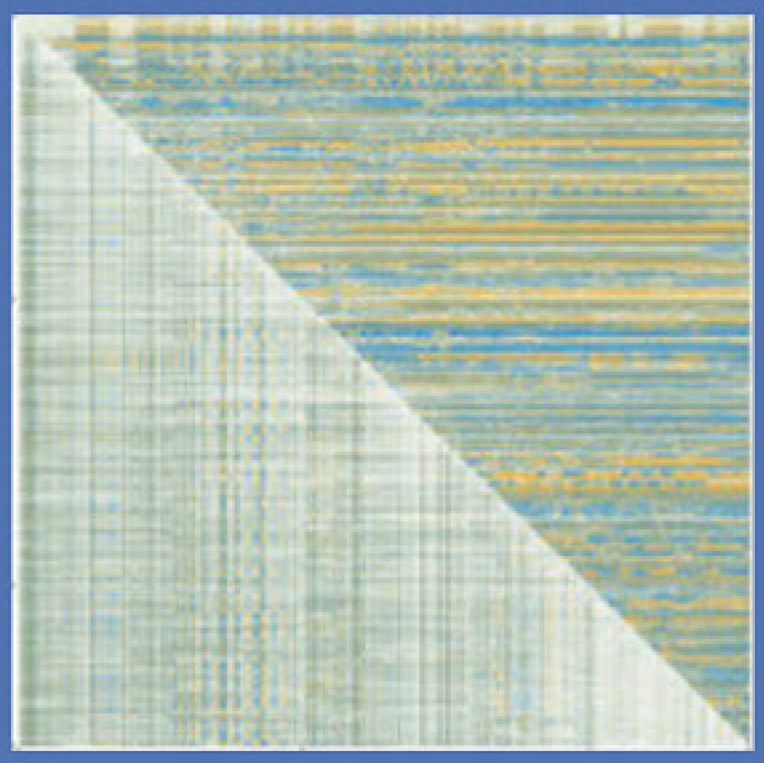












Steven S. Skiena

Springer

Steven S. Skiena
Computer Science Department
Stony Brook University
Stony Brook, NY
USA

ISSN 1868-0941 ISSN 1868-095X (electronic)
Texts in Computer Science
ISBN 978-3-319-55443-3 ISBN 978-3-319-55444-0 (eBook)
https://doi.org/10.1007/978-3-319-55444-0

Library of Congress Control Number: 2017943201

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

Chapter 1

What is Data Science?

> The purpose of computing is insight, not numbers.
>
> – Richard W. Hamming

What is data science? Like any emerging field, it hasn't been completely defined yet, but you know enough about it to be interested or else you wouldn't be reading this book.

I think of data science as lying at the intersection of computer science, statistics, and substantive application domains. From computer science comes machine learning and high-performance computing technologies for dealing with scale. From statistics comes a long tradition of exploratory data analysis, significance testing, and visualization. From application domains in business and the sciences comes challenges worthy of battle, and evaluation standards to assess when they have been adequately conquered.

But these are all well-established fields. Why data science, and why now? I see three reasons for this sudden burst of activity:

- New technology makes it possible to capture, annotate, and store vast amounts of social media, logging, and sensor data. After you have amassed all this data, you begin to wonder what you can do with it.

- Computing advances make it possible to analyze data in novel ways and at ever increasing scales. Cloud computing architectures give even the little guy access to vast power when they need it. New approaches to machine learning have lead to amazing advances in longstanding problems, like computer vision and natural language processing.

- Prominent technology companies (like Google and Facebook) and quantitative hedge funds (like Renaissance Technologies and TwoSigma) have proven the power of modern data analytics. Success stories applying data to such diverse areas as sports management (*Moneyball* [Lew04]) and election forecasting (Nate Silver [Sil12]) have served as role models to bring data science to a large popular audience.

S.S. Skiena, *The Data Science Design Manual,*
Texts in Computer Science, https://doi.org/10.1007/978-3-319-55444-0_1

This introductory chapter has three missions. First, I will try to explain how good data scientists think, and how this differs from the mindset of traditional programmers and software developers. Second, we will look at data sets in terms of the potential for what they can be used for, and learn to ask the broader questions they are capable of answering. Finally, I introduce a collection of data analysis challenges that will be used throughout this book as motivating examples.

1.1 Computer Science, Data Science, and Real Science

Computer scientists, by nature, don't respect data. They have traditionally been taught that the algorithm was the thing, and that data was just meat to be passed through a sausage grinder.

So to qualify as an effective data scientist, you must first learn to think like a real scientist. Real scientists strive to understand the natural world, which is a complicated and messy place. By contrast, computer scientists tend to build their own clean and organized virtual worlds and live comfortably within them. Scientists obsess about discovering things, while computer scientists invent rather than discover.

People's mindsets strongly color how they think and act, causing misunderstandings when we try to communicate outside our tribes. So fundamental are these biases that we are often unaware we have them. Examples of the cultural differences between computer science and real science include:

- *Data vs. method centrism*: Scientists are data driven, while computer scientists are algorithm driven. Real scientists spend enormous amounts of effort collecting data to answer their question of interest. They invent fancy measuring devices, stay up all night tending to experiments, and devote most of their thinking to how to get the data they need.

  By contrast, computer scientists obsess about methods: which algorithm is better than which other algorithm, which programming language is best for a job, which program is better than which other program. The details of the data set they are working on seem comparably unexciting.

- *Concern about results*: Real scientists care about answers. They analyze data to discover something about how the world works. Good scientists care about whether the results make sense, because they care about what the answers mean.

  By contrast, bad computer scientists worry about producing plausible-looking numbers. As soon as the numbers stop looking grossly wrong, they are presumed to be right. This is because they are personally less invested in what can be learned from a computation, as opposed to getting it done quickly and efficiently.

- *Robustness*: Real scientists are comfortable with the idea that data has errors. In general, computer scientists are not. Scientists think a lot about possible sources of bias or error in their data, and how these possible problems can effect the conclusions derived from them. Good programmers use strong data-typing and parsing methodologies to guard against formatting errors, but the concerns here are different.

  Becoming aware that data can have errors is empowering. Computer scientists chant "garbage in, garbage out" as a defensive mantra to ward off criticism, a way to say *that's not my job*. Real scientists get close enough to their data to smell it, giving it the sniff test to decide whether it is likely to be garbage.

- *Precision*: Nothing is ever completely true or false in science, while *everything* is either true or false in computer science or mathematics.

  Generally speaking, computer scientists are happy printing floating point numbers to as many digits as possible: $8/13 = 0.61538461538$. Real scientists will use only two significant digits: $8/13 \approx 0.62$. Computer scientists care what a number is, while real scientists care what it means.

Aspiring data scientists must learn to think like real scientists. Your job is going to be to turn numbers into insight. It is important to understand the *why* as much as the *how*.

To be fair, it benefits real scientists to think like data scientists as well. New experimental technologies enable measuring systems on vastly greater scale than ever possible before, through technologies like full-genome sequencing in biology and full-sky telescope surveys in astronomy. With new breadth of view comes new levels of vision.

Traditional *hypothesis-driven* science was based on asking specific questions of the world and then generating the specific data needed to confirm or deny it. This is now augmented by *data-driven* science, which instead focuses on generating data on a previously unheard of scale or resolution, in the belief that new discoveries will come as soon as one is able to look at it. Both ways of thinking will be important to us:

- Given a problem, what available data will help us answer it?

- Given a data set, what interesting problems can we apply it to?

There is another way to capture this basic distinction between software engineering and data science. It is that software developers are hired to build systems, while data scientists are hired to produce insights.

This may be a point of contention for some developers. There exist an important class of engineers who wrangle the massive distributed infrastructures necessary to store and analyze, say, financial transaction or social media data on a full Facebook or Twitter-level of scale. Indeed, I will devote Chapter 12 to the distinctive challenges of big data infrastructures. These engineers are building tools and systems to support data science, even though they may not personally mine the data they wrangle. Do they qualify as data scientists?

This is a fair question, one I will finesse a bit so as to maximize the potential readership of this book. But I do believe that the better such engineers understand the full data analysis pipeline, the more likely they will be able to build powerful tools capable of providing important insights. A major goal of this book is providing big data engineers with the intellectual tools to think like big data scientists.

1.2 Asking Interesting Questions from Data

Good data scientists develop an inherent curiosity about the world around them, particularly in the associated domains and applications they are working on. They enjoy talking shop with the people whose data they work with. They ask them questions: What is the coolest thing you have learned about this field? Why did you get interested in it? What do you hope to learn by analyzing your data set? Data scientists always ask questions.

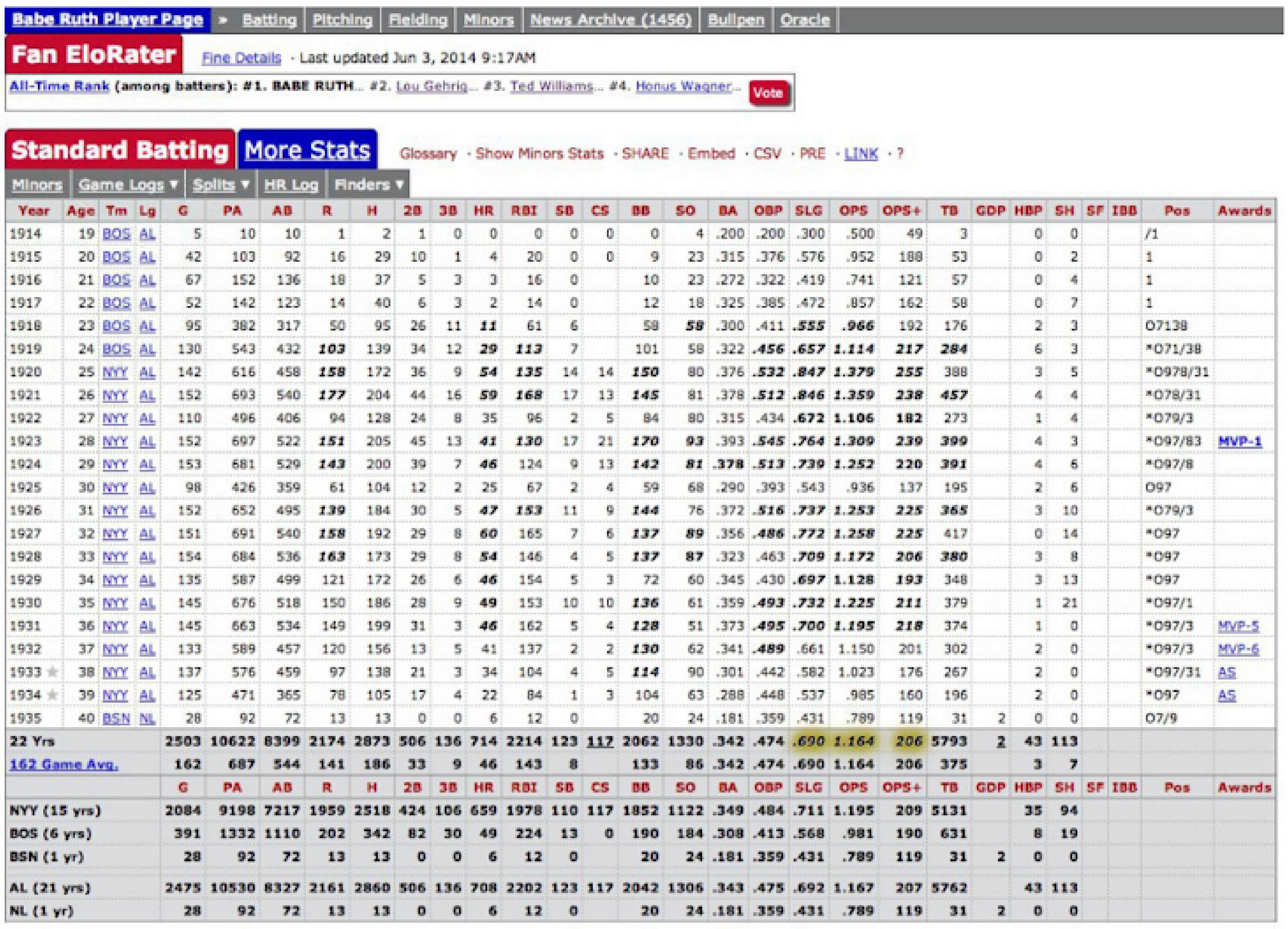
Good data scientists have wide-ranging interests. They read the newspaper every day to get a broader perspective on what is exciting. They understand that the world is an interesting place. Knowing a little something about everything equips them to play in other people's backyards. They are brave enough to get out of their comfort zones a bit, and driven to learn more once they get there.

Software developers are not really encouraged to ask questions, but data scientists are. We ask questions like:

- What things might you be able to learn from a given data set?
- What do you/your people really want to know about the world?
- What will it mean to you once you find out?

Computer scientists traditionally do not really appreciate data. Think about the way algorithm performance is experimentally measured. Usually the program is run on "random data" to see how long it takes. They rarely even look at the results of the computation, except to verify that it is correct and efficient. Since the "data" is meaningless, the results cannot be important. In contrast, real data sets are a scarce resource, which required hard work and imagination to obtain.

Becoming a data scientist requires learning to ask questions about data, so let's practice. Each of the subsections below will introduce an interesting data set. After you understand what kind of information is available, try to come up with, say, five interesting questions you might explore/answer with access to this data set.



Babe Ruth Player Page » Batting | Pitching | Fielding | Minors | News Archive (1456) | Bullpen | Oracle

Fan EloRater Fine Details · Last updated Jun 3, 2014 9:17AM

All-Time Rank (among batters): #1. BABE RUTH... #2. Lou Gehrig... #3. Ted Williams... #4. Honus Wagner... Vote

Standard Batting More Stats Glossary · Show Minors Stats · SHARE · Embed · CSV · PRE · LINK · ?

Minors | Game Logs ▾ | Splits ▾ | HR Log | Finders ▾

Year	Age	Tm	Lg	G	PA	AB	R	H	2B	3B	HR	RBI	SB	CS	BB	SO	BA	OBP	SLG	OPS	OPS+	TB	GDP	HBP	SH	SF	IBB	Pos	Awards
1914	19	BOS	AL	5	10	10	1	2	1	0	0	0	0	0	0	4	.200	.200	.300	.500	49	3		0	0			/1	
1915	20	BOS	AL	42	103	92	16	29	10	1	4	20	0	0	9	23	.315	.376	.576	.952	188	53		0	2			1	
1916	21	BOS	AL	67	152	136	18	37	5	3	3	16	0		10	23	.272	.322	.419	.741	121	57		0	4			1	
1917	22	BOS	AL	52	142	123	14	40	6	3	2	14	0		12	18	.325	.385	.472	.857	162	58		0	7			1	
1918	23	BOS	AL	95	382	317	50	95	26	11	11	61	6		58	58	.300	.411	.555	.966	192	176		2	3			O7138	
1919	24	BOS	AL	130	543	432	103	139	34	12	29	113	7		101	58	.322	.456	.657	1.114	217	284		6	3			*O71/38	
1920	25	NYY	AL	142	616	458	158	172	36	9	54	135	14	14	150	80	.376	.532	.847	1.379	255	388		3	5			*O978/31	
1921	26	NYY	AL	152	693	540	177	204	44	16	59	168	17	13	145	81	.378	.512	.846	1.359	238	457		4	4			*O78/31	
1922	27	NYY	AL	110	496	406	94	128	24	8	35	96	2	5	84	80	.315	.434	.672	1.106	182	273		1	4			*O79/3	
1923	28	NYY	AL	152	697	522	151	205	45	13	41	130	17	21	170	93	.393	.545	.764	1.309	239	399		4	3			*O97/83	MVP-1
1924	29	NYY	AL	153	681	529	143	200	39	7	46	124	9	13	142	81	.378	.513	.739	1.252	220	391		4	6			*O97/8	
1925	30	NYY	AL	98	426	359	61	104	12	2	25	67	2	4	59	68	.290	.393	.543	.936	137	195		2	6			O97	
1926	31	NYY	AL	152	652	495	139	184	30	5	47	153	11	9	144	76	.372	.516	.737	1.253	225	365		3	10			*O79/3	
1927	32	NYY	AL	151	691	540	158	192	29	8	60	165	7	6	137	89	.356	.486	.772	1.258	225	417		0	14			*O97	
1928	33	NYY	AL	154	684	536	163	173	29	8	54	146	4	5	137	87	.323	.463	.709	1.172	206	380		3	8			*O97	
1929	34	NYY	AL	135	587	499	121	172	26	6	46	154	5	3	72	60	.345	.430	.697	1.128	193	348		3	13			*O97	
1930	35	NYY	AL	145	676	518	150	186	28	9	49	153	10	10	136	61	.359	.493	.732	1.225	211	379		1	21			*O97/1	
1931	36	NYY	AL	145	663	534	149	199	31	3	46	162	5	4	128	51	.373	.495	.700	1.195	218	374		1	0			*O97/3	MVP-5
1932	37	NYY	AL	133	589	457	120	156	13	5	41	137	2	2	130	62	.341	.489	.661	1.150	201	302		2	0			*O97/3	MVP-6
1933 ★	38	NYY	AL	137	576	459	97	138	21	3	34	104	4	5	114	90	.301	.442	.582	1.023	176	267		2	0			*O97/31	AS
1934 ★	39	NYY	AL	125	471	365	78	105	17	4	22	84	1	3	104	63	.288	.448	.537	.985	160	196		2	0			*O97	AS
1935	40	BSN	NL	28	92	72	13	13	0	0	6	12	0		20	24	.181	.359	.431	.789	119	31	2	0	0			O7/9	
22 Yrs				2503	10622	8399	2174	2873	506	136	714	2214	123	117	2062	1330	.342	.474	.690	1.164	206	5793	2	43	113				
162 Game Avg.				162	687	544	141	186	33	9	46	143	8		133	86	.342	.474	.690	1.164	206	375		3	7				
				G	PA	AB	R	H	2B	3B	HR	RBI	SB	CS	BB	SO	BA	OBP	SLG	OPS	OPS+	TB	GDP	HBP	SH	SF	IBB	Pos	Awards
NYY (15 yrs)				2084	9198	7217	1959	2518	424	106	659	1978	110	117	1852	1122	.349	.484	.711	1.195	209	5131		35	94				
BOS (6 yrs)				391	1332	1110	202	342	82	30	49	224	13	0	190	184	.308	.413	.568	.981	190	631		8	19				
BSN (1 yr)				28	92	72	13	13	0	0	6	12	0		20	24	.181	.359	.431	.789	119	31	2	0	0				
AL (21 yrs)				2475	10530	8327	2161	2860	506	136	708	2202	123	117	2042	1306	.343	.475	.692	1.167	207	5762		43	113				
NL (1 yr)				28	92	72	13	13	0	0	6	12	0		20	24	.181	.359	.431	.789	119	31	2	0	0				

Figure 1.1: Statistical information on the performance of Babe Ruth can be found at `http://www.baseball-reference.com`.

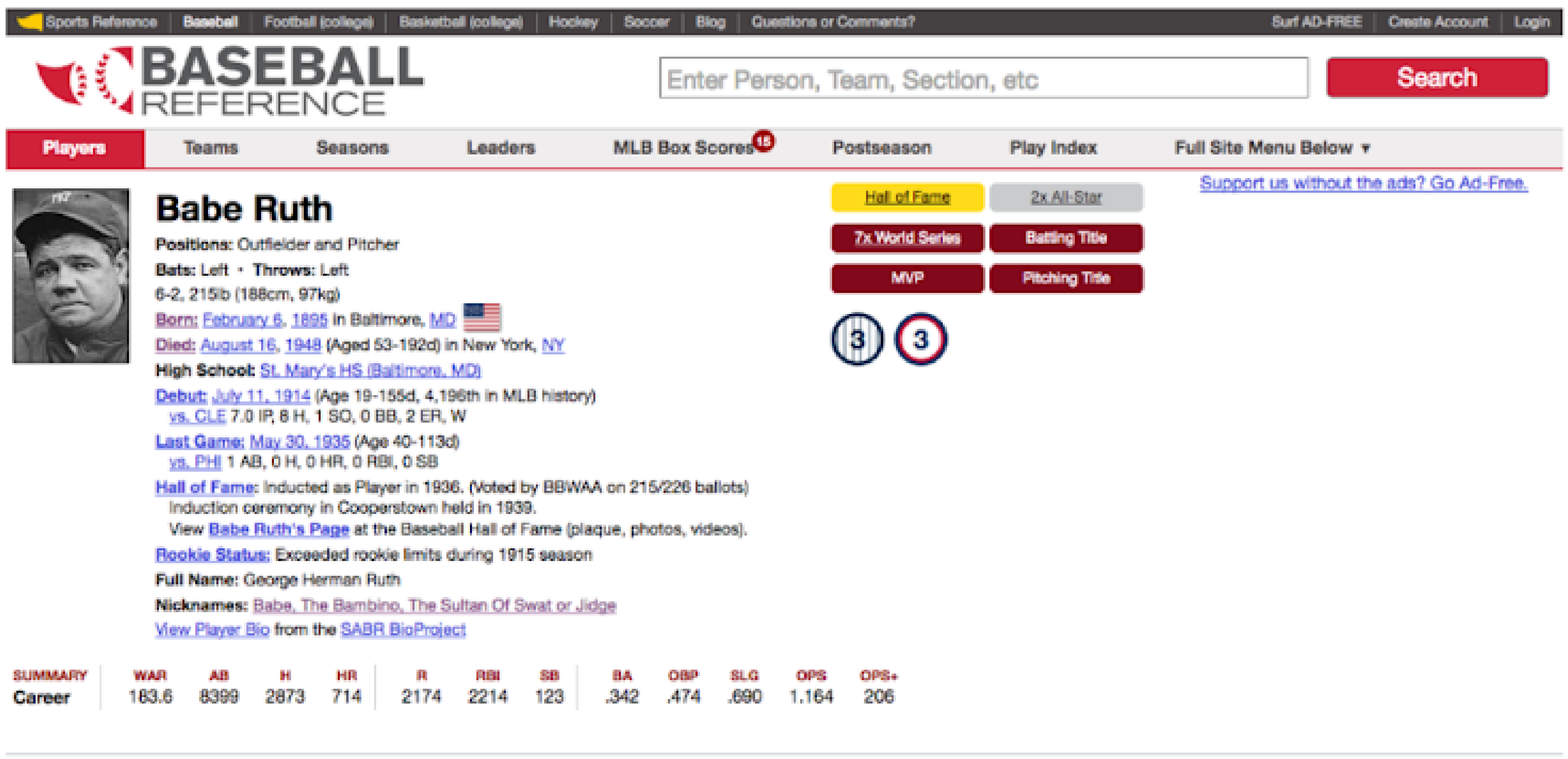
The key is thinking broadly: the answers to big, general questions often lie buried in highly-specific data sets, which were by no means designed to contain them.

1.2.1 The Baseball Encyclopedia

Baseball has long had an outsized importance in the world of data science. This sport has been called the national pastime of the United States; indeed, French historian Jacques Barzun observed that "Whoever wants to know the heart and mind of America had better learn baseball." I realize that many readers are not American, and even those that are might be completely disinterested in sports. But stick with me for a while.

What makes baseball important to data science is its extensive statistical record of play, dating back for well over a hundred years. Baseball is a sport of discrete events: pitchers throw balls and batters try to hit them – that naturally lends itself to informative statistics. Fans get immersed in these statistics as children, building their intuition about the strengths and limitations of quantitative analysis. Some of these children grow up to become data scientists. Indeed, the success of Brad Pitt's statistically-minded baseball team in the movie *Moneyball* remains the American public's most vivid contact with data science.

This historical baseball record is available at `http://www.baseball-reference.com`. There you will find complete statistical data on the performance of every player who even stepped on the field. This includes summary statistics of each season's batting, pitching, and fielding record, plus information about teams



Babe Ruth

Positions: Outfielder and Pitcher

Bats: Left • Throws: Left

6-2, 215lb (188cm, 97kg)

Born: February 6, 1895 in Baltimore, MD

Died: August 16, 1948 (Aged 53-192d) in New York, NY

High School: St. Mary's HS (Baltimore, MD)

Debut: July 11, 1914 (Age 19-155d, 4,196th in MLB history)
vs. CLE 7.0 IP, 8 H, 1 SO, 0 BB, 2 ER, W

Last Game: May 30, 1935 (Age 40-113d)
vs. PHI 1 AB, 0 H, 0 HR, 0 RBI, 0 SB

Hall of Fame: Inducted as Player in 1936. (Voted by BBWAA on 215/226 ballots)
Induction ceremony in Cooperstown held in 1939.
View Babe Ruth's Page at the Baseball Hall of Fame (plaque, photos, videos).

Rookie Status: Exceeded rookie limits during 1915 season

Full Name: George Herman Ruth

Nicknames: Babe, The Bambino, The Sultan Of Swat or Jidge

View Player Bio from the SABR BioProject

SUMMARY	WAR	AB	H	HR	R	RBI	SB	BA	OBP	SLG	OPS	OPS+
Career	183.6	8399	2873	714	2174	2214	123	.342	.474	.690	1.164	206

Figure 1.2: Personal information on every major league baseball player is available at `http://www.baseball-reference.com`.

and awards as shown in Figure 1.1.

But more than just statistics, there is metadata on the life and careers of all the people who have ever played major league baseball, as shown in Figure 1.2. We get the vital statistics of each player (height, weight, handedness) and their lifespan (when/where they were born and died). We also get salary information (how much each player got paid every season) and transaction data (how did they get to be the property of each team they played for).

Now, I realize that many of you do not have the slightest knowledge of or interest in baseball. This sport is somewhat reminiscent of cricket, if that helps. But remember that as a data scientist, it is your job to be interested in the world around you. Think of this as chance to learn something.

So what interesting questions can you answer with this baseball data set? Try to write down five questions before moving on. Don't worry, I will wait here for you to finish.

The most obvious types of questions to answer with this data are directly related to baseball:

- How can we best measure an individual player's skill or value?
- How fairly do trades between teams generally work out?
- What is the general trajectory of player's performance level as they mature and age?
- To what extent does batting performance correlate with position played? For example, are outfielders really better hitters than infielders?

These are interesting questions. But even more interesting are questions about demographic and social issues. Almost 20,000 major league baseball players have taken the field over the past 150 years, providing a large, extensively-documented cohort of men who can serve as a proxy for even larger, less well-documented populations. Indeed, we can use this baseball player data to answer questions like:

- Do left-handed people have shorter lifespans than right-handers? Handedness is not captured in most demographic data sets, but has been diligently assembled here. Indeed, analysis of this data set has been used to show that right-handed people live longer than lefties [HC88]!

- How often do people return to live in the same place where they were born? Locations of birth and death have been extensively recorded in this data set. Further, almost all of these people played at least part of their career far from home, thus exposing them to the wider world at a critical time in their youth.

- Do player salaries generally reflect past, present, or future performance?

- To what extent have heights and weights been increasing in the population at large?

There are two particular themes to be aware of here. First, the identifiers and reference tags (i.e. the metadata) often prove more interesting in a data set than the stuff we are supposed to care about, here the statistical record of play.

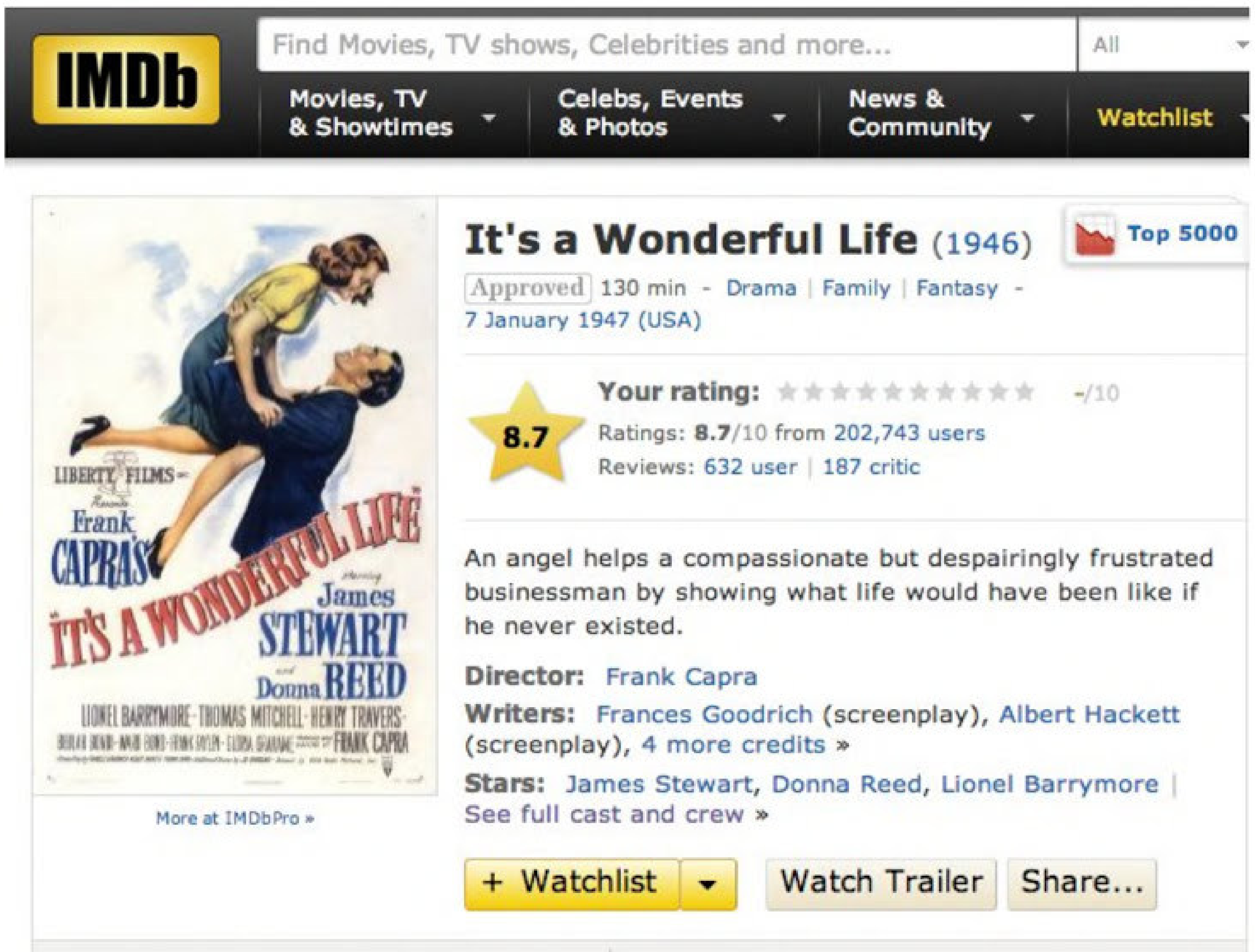
Second is the idea of a *statistical proxy*, where you use the data set you have to substitute for the one you really want. The data set of your dreams likely does not exist, or may be locked away behind a corporate wall even if it does. A good data scientist is a pragmatist, seeing what they can do with what they have instead of bemoaning what they cannot get their hands on.

1.2.2 The Internet Movie Database (IMDb)

Everybody loves the movies. The Internet Movie Database (IMDb) provides crowdsourced and curated data about all aspects of the motion picture industry, at `www.imdb.com`. IMDb currently contains data on over 3.3 million movies and TV programs. For each film, IMDb includes its title, running time, genres, date of release, and a full list of cast and crew. There is financial data about each production, including the budget for making the film and how well it did at the box office.

Finally, there are extensive ratings for each film from viewers and critics. This rating data consists of scores on a zero to ten stars scale, cross-tabulated into averages by age and gender. Written reviews are often included, explaining why a particular critic awarded a given number of stars. There are also links between films: for example, identifying which other films have been watched most often by viewers of *It's a Wonderful Life.*

Every actor, director, producer, and crew member associated with a film merits an entry in IMDb, which now contains records on 6.5 million people.



It's a Wonderful Life (1946)

Approved 130 min - Drama | Family | Fantasy - 7 January 1947 (USA)

Your rating: ★★★★★★★★★★ -/10

8.7 Ratings: **8.7**/10 from 202,743 users

Reviews: 632 user | 187 critic

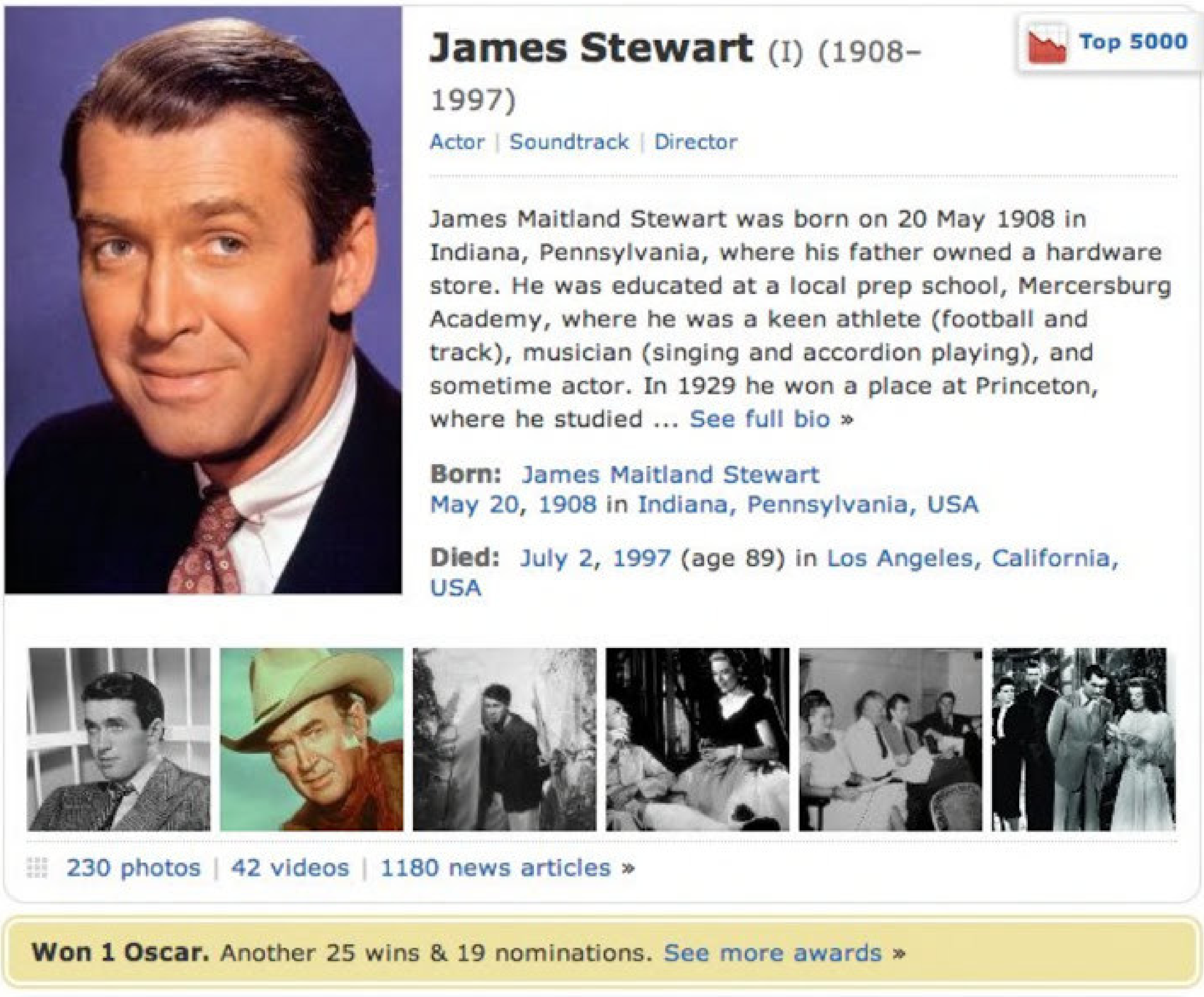
An angel helps a compassionate but despairingly frustrated businessman by showing what life would have been like if he never existed.

Director: Frank Capra

Writers: Frances Goodrich (screenplay), Albert Hackett (screenplay), 4 more credits »

Stars: James Stewart, Donna Reed, Lionel Barrymore | See full cast and crew »

Figure 1.3: Representative film data from the Internet Movie Database.



James Stewart (I) (1908–1997)

Actor | Soundtrack | Director

James Maitland Stewart was born on 20 May 1908 in Indiana, Pennsylvania, where his father owned a hardware store. He was educated at a local prep school, Mercersburg Academy, where he was a keen athlete (football and track), musician (singing and accordion playing), and sometime actor. In 1929 he won a place at Princeton, where he studied ... See full bio »

Born: James Maitland Stewart May 20, 1908 in Indiana, Pennsylvania, USA

Died: July 2, 1997 (age 89) in Los Angeles, California, USA

230 photos | 42 videos | 1180 news articles »

Won 1 Oscar. Another 25 wins & 19 nominations. See more awards »

Figure 1.4: Representative actor data from the Internet Movie Database.

These happen to include my brother, cousin, and sister-in-law. Each actor is linked to every film they appeared in, with a description of their role and their ordering in the credits. Available data about each personality includes birth/death dates, height, awards, and family relations.

So what kind of questions can you answer with this movie data?

Perhaps the most natural questions to ask IMDb involve identifying the extremes of movies and actors:

- Which actors appeared in the most films? Earned the most money? Appeared in the lowest rated films? Had the longest career or the shortest lifespan?

- What was the highest rated film each year, or the best in each genre? Which movies lost the most money, had the highest-powered casts, or got the least favorable reviews.

Then there are larger-scale questions one can ask about the nature of the motion picture business itself:

- How well does movie gross correlate with viewer ratings or awards? Do customers instinctively flock to trash, or is virtue on the part of the creative team properly rewarded?

- How do Hollywood movies compare to Bollywood movies, in terms of ratings, budget, and gross? Are American movies better received than foreign films, and how does this differ between U.S. and non-U.S. reviewers?

- What is the age distribution of actors and actresses in films? How much younger is the actress playing the wife, on average, than the actor playing the husband? Has this disparity been increasing or decreasing with time?

- Live fast, die young, and leave a good-looking corpse? Do movie stars live longer or shorter lives than bit players, or compared to the general public?

Assuming that people working together on a film get to know each other, the cast and crew data can be used to build a social network of the movie business. What does the social network of actors look like? The Oracle of Bacon (`https://oracleofbacon.org/`) posits Kevin Bacon as the center of the Hollywood universe and generates the shortest path to Bacon from any other actor. Other actors, like Samuel L. Jackson, prove even more central.

More critically, can we analyze this data to determine the probability that someone will like a given movie? The technique of *collaborative filtering* finds people who liked films that I also liked, and recommends other films that *they* liked as good candidates for me. The 2007 Netflix Prize was a $1,000,000 competition to produce a ratings engine 10% better than the proprietary Netflix system. The ultimate winner of this prize (BellKor) used a variety of data sources and techniques, including the analysis of links [BK07].

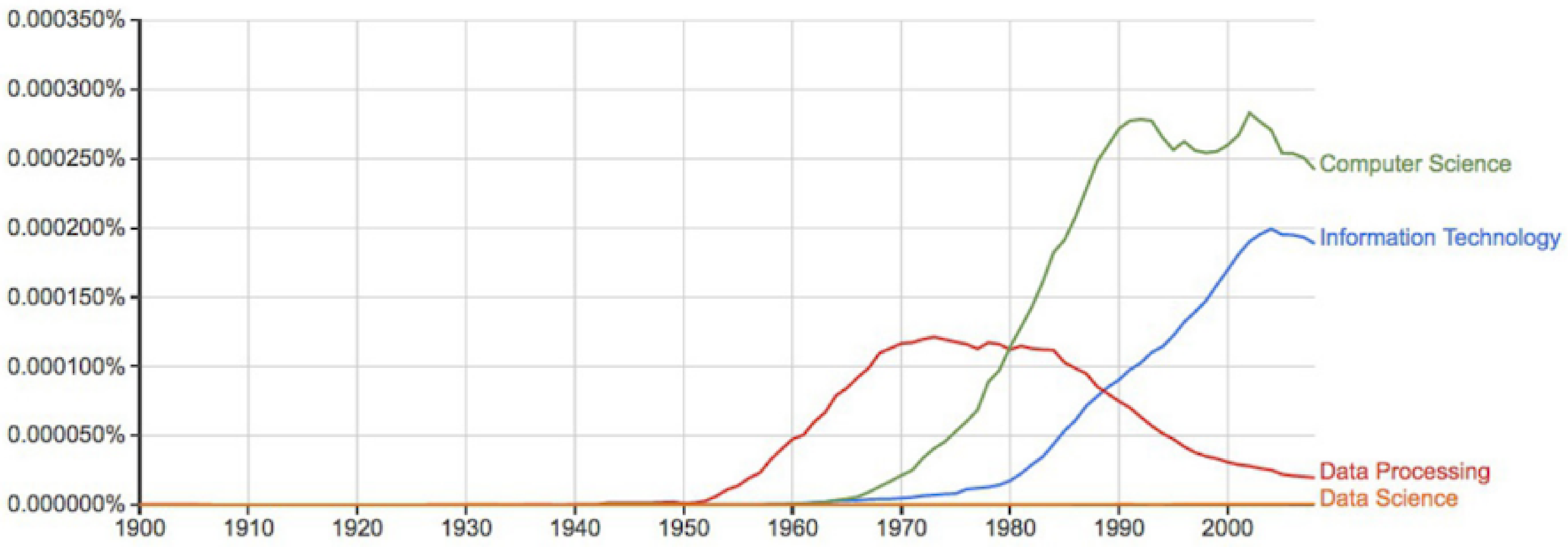


Figure 1.5: The rise and fall of data processing, as witnessed by Google Ngrams.

1.2.3 Google Ngrams

Printed books have been the primary repository of human knowledge since Gutenberg's invention of movable type in 1439. Physical objects live somewhat uneasily in today's digital world, but technology has a way of reducing everything to data. As part of its mission to organize the world's information, Google undertook an effort to scan all of the world's published books. They haven't quite gotten there yet, but the 30 million books thus far digitized represent over 20% of all books ever published.

Google uses this data to improve search results, and provide fresh access to out-of-print books. But perhaps the coolest product is *Google Ngrams*, an amazing resource for monitoring changes in the cultural zeitgeist. It provides the frequency with which short phrases occur in books published each year. Each phrase must occur at least forty times in their scanned book corpus. This eliminates obscure words and phrases, but leaves over two billion time series available for analysis.

This rich data set shows how language use has changed over the past 200 years, and has been widely applied to cultural trend analysis [MAV$^+$11]. Figure 1.5 uses this data to show how the word *data* fell out of favor when thinking about computing. *Data processing* was the popular term associated with the computing field during the punched card and spinning magnetic tape era of the 1950s. The Ngrams data shows that the rapid rise of *Computer Science* did not eclipse *Data Processing* until 1980. Even today, *Data Science* remains almost invisible on this scale.

Check out Google Ngrams at `http://books.google.com/ngrams`. I promise you will enjoy playing with it. Compare *hot dog* to *tofu*, *science* against *religion*, *freedom* to *justice*, and *sex* vs. *marriage*, to better understand this fantastic telescope for looking into the past.

But once you are done playing, think of bigger things you could do if you got your hands on this data. Assume you have access to the annual number of references for *all* words/phrases published in books over the past 200 years.

Google makes this data freely available. So what are you going to do with it?

Observing the time series associated with particular words using the Ngrams Viewer is fun. But more sophisticated historical trends can be captured by aggregating multiple time series together. The following types of questions seem particularly interesting to me:

- How has the amount of cursing changed over time? Use of the four-letter words I am most familiar with seem to have exploded since 1960, although it is perhaps less clear whether this reflects increased cussing or lower publication standards.

- How often do new words emerge and get popular? Do these words tend to stay in common usage, or rapidly fade away? Can we detect when words change meaning over time, like the transition of *gay* from *happy* to *homosexual*?

- Have standards of spelling been improving or deteriorating with time, especially now that we have entered the era of automated spell checking? Rarely-occurring words that are only one character removed from a commonly-used word are likely candidates to be spelling errors (e.g. *algorithm* vs. *algorthm*). Aggregated over many different misspellings, are such errors increasing or decreasing?

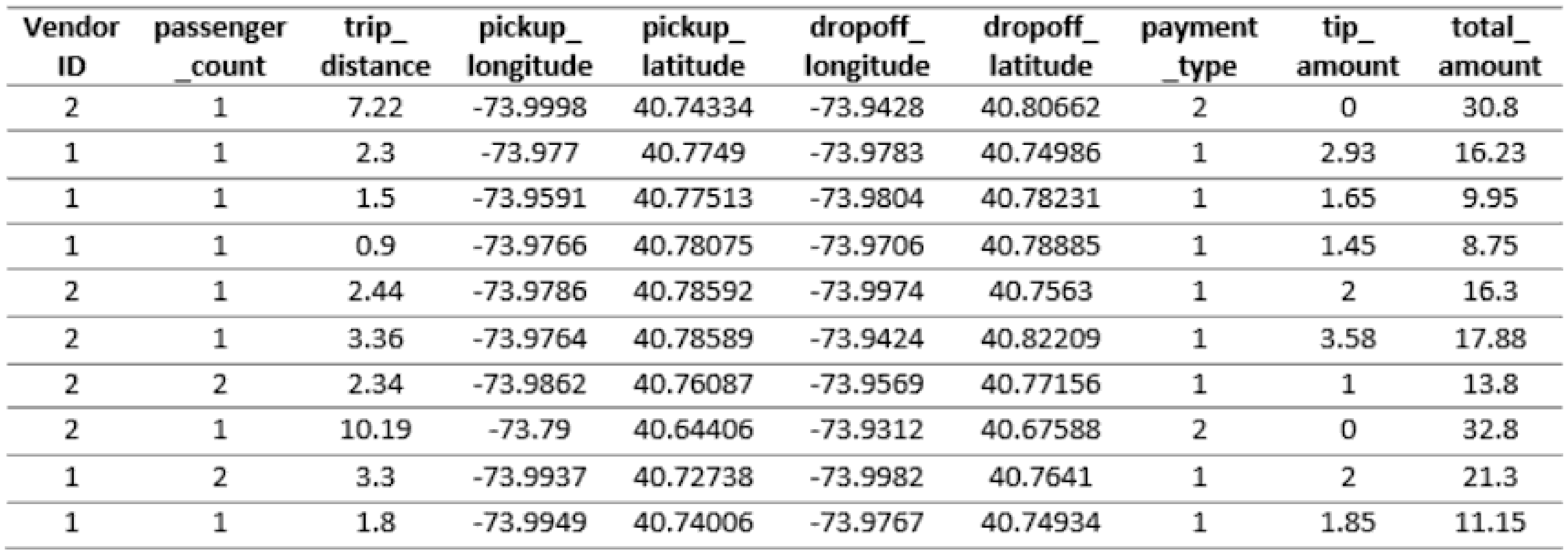
You can also use this Ngrams corpus to build a language model that captures the meaning and usage of the words in a given language. We will discuss word embeddings in Section 11.6.3, which are powerful tools for building language models. Frequency counts reveal which words are most popular. The frequency of word pairs appearing next to each other can be used to improve speech recognition systems, helping to distinguish whether the speaker said *that's too bad* or *that's to bad*. These millions of books provide an ample data set to build representative models from.

1.2.4 New York Taxi Records

Every financial transaction today leaves a data trail behind it. Following these paths can lead to interesting insights.

Taxi cabs form an important part of the urban transportation network. They roam the streets of the city looking for customers, and then drive them to their destination for a fare proportional to the length of the trip. Each cab contains a metering device to calculate the cost of the trip as a function of time. This meter serves as a record keeping device, and a mechanism to ensure that the driver charges the proper amount for each trip.

The taxi meters currently employed in New York cabs can do many things beyond calculating fares. They act as credit card terminals, providing a way



Vendor ID	passenger _count	trip_ distance	pickup_ longitude	pickup_ latitude	dropoff_ longitude	dropoff_ latitude	payment _type	tip_ amount	total_ amount
2	1	7.22	-73.9998	40.74334	-73.9428	40.80662	2	0	30.8
1	1	2.3	-73.977	40.7749	-73.9783	40.74986	1	2.93	16.23
1	1	1.5	-73.9591	40.77513	-73.9804	40.78231	1	1.65	9.95
1	1	0.9	-73.9766	40.78075	-73.9706	40.78885	1	1.45	8.75
2	1	2.44	-73.9786	40.78592	-73.9974	40.7563	1	2	16.3
2	1	3.36	-73.9764	40.78589	-73.9424	40.82209	1	3.58	17.88
2	2	2.34	-73.9862	40.76087	-73.9569	40.77156	1	1	13.8
2	1	10.19	-73.79	40.64406	-73.9312	40.67588	2	0	32.8
1	2	3.3	-73.9937	40.72738	-73.9982	40.7641	1	2	21.3
1	1	1.8	-73.9949	40.74006	-73.9767	40.74934	1	1.85	11.15

Figure 1.6: Representative fields from the New York city taxi cab data: pick up and dropoff points, distances, and fares.

for customers to pay for rides without cash. They are integrated with global positioning systems (GPS), recording the exact location of every pickup and drop off. And finally, since they are on a wireless network, these boxes can communicate all of this data back to a central server.

The result is a database documenting every single trip by all taxi cabs in one of the world's greatest cities, a small portion of which is shown in Figure 1.6. Because the New York Taxi and Limousine Commission is a public agency, its non-confidential data is available to all under the Freedom of Information Act (FOA).

Every ride generates two records: one with data on the trip, the other with details of the fare. Each trip is keyed to the medallion (license) of each car coupled with the identifier of each driver. For each trip, we get the time/date of pickup and drop-off, as well as the GPS coordinates (longitude and latitude) of the starting location and destination. We do not get GPS data of the route they traveled between these points, but to some extent that can be inferred by the shortest path between them.

As for fare data, we get the metered cost of each trip, including tax, surcharge and tolls. It is traditional to pay the driver a tip for service, the amount of which is also recorded in the data.

So I'm talking to you. This taxi data is readily available, with records of over 80 million trips over the past several years. What are you going to do with it?

Any interesting data set can be used to answer questions on many different scales. This taxi fare data can help us better understand the transportation industry, but also how the city works and how we could make it work even better. Natural questions with respect to the taxi industry include:

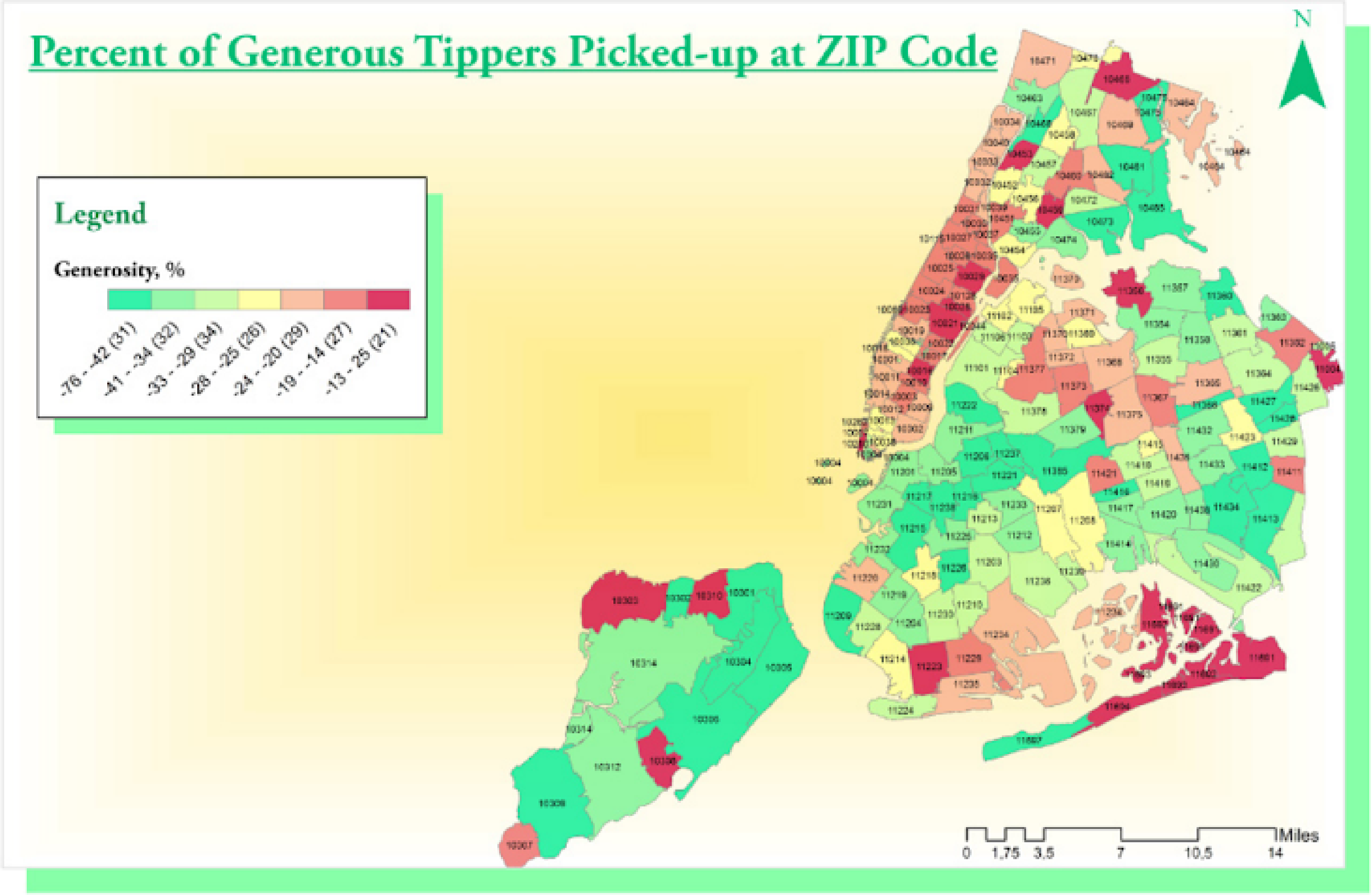


Figure 1.7: Which neighborhoods in New York city tip most generously? The relatively remote outer boroughs of Brooklyn and Queens, where trips are longest and supply is relatively scarce.

- How much money do drivers make each night, on average? What is the distribution? Do drivers make more on sunny days or rainy days?

- Where are the best spots in the city for drivers to cruise, in order to pick up profitable fares? How does this vary at different times of the day?

- How far do drivers travel over the course of a night's work? We can't answer this exactly using this data set, because it does not provide GPS data of the route traveled between fares. But we do know the last place of drop off, the next place of pickup, and how long it took to get between them. Together, this should provide enough information to make a sound estimate.

- Which drivers take their unsuspecting out-of-town passengers for a "ride," running up the meter on what should be a much shorter, cheaper trip?

- How much are drivers tipped, and why? Do faster drivers get tipped better? How do tipping rates vary by neighborhood, and is it the rich neighborhoods or poor neighborhoods which prove more generous?

  I will confess we did an analysis of this, which I will further describe in the war story of Section 9.3. We found a variety of interesting patterns [SS15]. Figure 1.7 shows that Manhattanites are generally cheapskates relative to large swaths of Brooklyn, Queens, and Staten Island, where trips are longer and street cabs a rare but welcome sight.

- *Simple models do not require massive data to fit or evaluate*: A typical data science task might be to make a decision (say, whether I should offer this fellow life insurance?) on the basis of a small number of variables: say age, gender, height, weight, and the presence or absence of existing medical conditions.

  If I have this data on 1 million people with their associated life outcomes, I should be able to build a good general model of coverage risk. It probably wouldn't help me build a substantially better model if I had this data on hundreds of millions of people. The decision criteria on only a few variables (like age and martial status) cannot be too complex, and should be robust over a large number of applicants. Any observation that is so subtle it requires massive data to tease out will prove irrelevant to a large business which is based on volume.

Big data is sometimes called *bad data*. It is often gathered as the by-product of a given system or procedure, instead of being purposefully collected to answer your question at hand. The result is that we might have to go to heroic efforts to make sense of something just because we have it.

Consider the problem of getting a pulse on voter preferences among presidential candidates. The big data approach might analyze massive Twitter or Facebook feeds, interpreting clues to their opinions in the text. The small data approach might be to conduct a poll, asking a few hundred people this specific question and tabulating the results. Which procedure do you think will prove more accurate? The right data set is the one most directly relevant to the tasks at hand, not necessarily the biggest one.

> *Take-Home Lesson*: Do not blindly aspire to analyze large data sets. Seek the *right* data to answer a given question, not necessarily the biggest thing you can get your hands on.

1.4 Classification and Regression

Two types of problems arise repeatedly in traditional data science and pattern recognition applications, the challenges of classification and regression. As this book has developed, I have pushed discussions of the algorithmic approaches to solving these problems toward the later chapters, so they can benefit from a solid understanding of core material in data munging, statistics, visualization, and mathematical modeling.

Still, I will mention issues related to classification and regression as they arise, so it makes sense to pause here for a quick introduction to these problems, to help you recognize them when you see them.

- *Classification*: Often we seek to assign a label to an item from a discrete set of possibilities. Such problems as predicting the winner of a particular sporting contest (team A or team B?) or deciding the genre of a given movie (comedy, drama, or animation?) are *classification* problems, since each entail selecting a label from the possible choices.

- *Regression*: Another common task is to forecast a given numerical quantity. Predicting a person's weight or how much snow we will get this year is a *regression* problem, where we forecast the future value of a numerical function in terms of previous values and other relevant features.

Perhaps the best way to see the intended distinction is to look at a variety of data science problems and label (classify) them as regression or classification. Different algorithmic methods are used to solve these two types of problems, although the same questions can often be approached in either way:

- Will the price of a particular stock be higher or lower tomorrow? (classification)
- What will the price of a particular stock be tomorrow? (regression)
- Is this person a good risk to sell an insurance policy to? (classification)
- How long do we expect this person to live? (regression)

Keep your eyes open for classification and regression problems as you encounter them in your life, and in this book.

1.5 Data Science Television: The Quant Shop

I believe that hands-on experience is necessary to internalize basic principles. Thus when I teach data science, I like to give each student team an interesting but messy forecasting challenge, and demand that they build and evaluate a predictive model for the task.

These forecasting challenges are associated with events where the students must make testable predictions. They start from scratch: finding the relevant data sets, building their own evaluation environments, and devising their model. Finally, I make them watch the event as it unfolds, so as to witness the vindication or collapse of their prediction.

As an experiment, we documented the evolution of each group's project on video in Fall 2014. Professionally edited, this became *The Quant Shop*, a television-like data science series for a general audience. The eight episodes of this first season are available at `http://www.quant-shop.com`, and include:

- *Finding Miss Universe* – The annual Miss Universe competition aspires to identify the most beautiful woman in the world. Can computational models predict who will win a beauty contest? Is beauty just subjective, or can algorithms tell who is the fairest one of all?

- *Modeling the Movies* – The business of movie making involves a lot of high-stakes data analysis. Can we build models to predict which film will gross the most on Christmas day? How about identifying which actors will receive awards for their performance?

- *Winning the Baby Pool* – Birth weight is an important factor in assessing the health of a newborn child. But how accurately can we predict junior's weight before the actual birth? How can data clarify environmental risks to developing pregnancies?

- *The Art of the Auction* – The world's most valuable artworks sell at auctions to the highest bidder. But can we predict how many millions a particular J.W. Turner painting will sell for? Can computers develop an artistic sense of what's worth buying?

- *White Christmas* – Weather forecasting is perhaps the most familiar domain of predictive modeling. Short-term forecasts are generally accurate, but what about longer-term prediction? What places will wake up to a snowy Christmas this year? And can you tell one month in advance?

- *Predicting the Playoffs* – Sports events have winners and losers, and bookies are happy to take your bets on the outcome of any match. How well can statistics help predict which football team will win the Super Bowl? Can Google's PageRank algorithm pick the winners on the field as accurately as it does on the web?

- *The Ghoul Pool* – Death comes to all men, but when? Can we apply actuarial models to celebrities, to decide who will be the next to die? Similar analysis underlies the workings of the life insurance industry, where accurate predictions of lifespan are necessary to set premiums which are both sustainable and affordable.

Figure 1.8: Exciting scenes from data science television: *The Quant Shop*.

- *Playing the Market* – Hedge fund quants get rich when guessing right about tomorrow's prices, and poor when wrong. How accurately can we predict future prices of gold and oil using histories of price data? What other information goes into building a successful price model?

I encourage you to watch some episodes of *The Quant Shop* in tandem with reading this book. We try to make it fun, although I am sure you will find plenty of things to cringe at. Each show runs for thirty minutes, and maybe will inspire you to tackle a prediction challenge of your own.

These programs will certainly give you more insight into these eight specific challenges. I will use these projects throughout this book to illustrate important lessons in how to do data science, both as positive and negative examples. These projects provide a laboratory to see how intelligent but inexperienced people not wildly unlike yourself thought about a data science problem, and what happened when they did.

1.5.1 Kaggle Challenges

Another source of inspiration are challenges from Kaggle (`www.kaggle.com`), which provides a competitive forum for data scientists. New challenges are posted on a regular basis, providing a problem definition, training data, and a scoring function over hidden evaluation data. A leader board displays the scores of the strongest competitors, so you can see how well your model stacks up in comparison with your opponents. The winners spill their modeling secrets during post-contest interviews, to help you improve your modeling skills.

Performing well on Kaggle challenges is an excellent credential to put on your resume to get a good job as a data scientist. Indeed, potential employers will track you down if you are a real Kaggle star. But the real reason to participate is that the problems are fun and inspiring, and practice helps make you a better data scientist.

The exercises at the end of each chapter point to expired Kaggle challenges, loosely connected to the material in that chapter. Be forewarned that Kaggle provides a misleading glamorous view of data science as applied machine learning, because it presents extremely well-defined problems with the hard work of data collection and cleaning already done for you. Still, I encourage you to check it out for inspiration, and as a source of data for new projects.

1.6 About the War Stories

Genius and wisdom are two distinct intellectual gifts. *Genius* shows in discovering the right answer, making imaginative mental leaps which overcome obstacles and challenges. *Wisdom* shows in avoiding obstacles in the first place, providing a sense of direction or guiding light that keeps us moving soundly in the right direction.

Genius is manifested in technical strength and depth, the ability to see things and do things that other people cannot. In contrast, wisdom comes from experience and general knowledge. It comes from listening to others. Wisdom comes from humility, observing how often you have been wrong in the past and figuring out why you were wrong, so as to better recognize future traps and avoid them.

Data science, like most things in life, benefits more from wisdom than from genius. In this book, I seek to pass on wisdom that I have accumulated the hard way through *war stories*, gleaned from a diverse set of projects I have worked on:

- *Large-scale text analytics and NLP*: My Data Science Laboratory at Stony Brook University works on a variety of projects in big data, including sentiment analysis from social media, historical trends analysis, deep learning approaches to natural language processing (NLP), and feature extraction from networks.

- *Start-up companies*: I served as co-founder and chief scientist to two data analytics companies: General Sentiment and Thrivemetrics. General Sentiment analyzed large-scale text streams from news, blogs, and social media to identify trends in the sentiment (positive or negative) associated with people, places, and things. Thrivemetrics applied this type of analysis to internal corporate communications, like email and messaging systems.

  Neither of these ventures left me wealthy enough to forgo my royalties from this book, but they did provide me with experience on cloud-based computing systems, and insight into how data is used in industry.

- *Collaborating with real scientists*: I have had several interesting collaborations with biologists and social scientists, which helped shape my understanding of the complexities of working with real data. Experimental data is horribly noisy and riddled with errors, yet you must do the best you can with what you have, in order to discover how the world works.

- *Building gambling systems*: A particularly amusing project was building a system to predict the results of jai-alai matches so we could bet on them, an experience recounted in my book *Calculated Bets: Computers, Gambling, and Mathematical Modeling to Win* [Ski01]. Our system relied on web scraping for data collection, statistical analysis, simulation/modeling, and careful evaluation. We also have developed and evaluated predictive models for movie grosses [ZS09], stock prices [ZS10], and football games [HS10] using social media analysis.

- *Ranking historical figures*: By analyzing Wikipedia to extract meaningful variables on over 800,000 historical figures, we developed a scoring function to rank them by their strength as historical memes. This ranking does a great job separating the greatest of the great (Jesus, Napoleon, Shakespeare, Mohammad, and Lincoln round out the top five) from lesser

The potential of ride-sharing systems in New York was studied by Santi et. al. [SRS$^+$14], who showed that almost 95% of the trips could have been shared with no more than five minutes delay per trip.

The Lydia system for sentiment analysis is described in [GSS07]. Methods to identify changes in word meaning through analysis of historical text corpora like Google Ngram are reported in [KARPS15].

1.9 Exercises

Identifying Data Sets

1-1. *[3]* Identify where interesting data sets relevant to the following domains can be found on the web:

(a) Books.

(b) Horse racing.

(c) Stock prices.

(d) Risks of diseases.

(e) Colleges and universities.

(f) Crime rates.

(g) Bird watching.

For each of these data sources, explain what you must do to turn this data into a usable format on your computer for analysis.

1-2. *[3]* Propose relevant data sources for the following *The Quant Shop* prediction challenges. Distinguish between sources of data that you are sure *somebody* must have, and those where the data is clearly available to you.

(a) *Miss Universe.*

(b) *Movie gross.*

(c) *Baby weight.*

(d) *Art auction price.*

(e) *White Christmas.*

(f) *Football champions.*

(g) *Ghoul pool.*

(h) *Gold/oil prices.*

1-3. *[3]* Visit `http://data.gov`, and identify five data sets that sound interesting to you. For each write a brief description, and propose three interesting things you might do with them.

Asking Questions

1-4. *[3]* For each of the following data sources, propose three interesting questions you can answer by analyzing them:

(a) Credit card billing data.

(b) Click data from `http://www.Amazon.com`.

(c) White Pages residential/commercial telephone directory.

1-5. *[5]* Visit Entrez, the National Center for Biotechnology Information (NCBI) portal. Investigate what data sources are available, particularly the Pubmed and Genome resources. Propose three interesting projects to explore with each of them.

1-6. *[5]* You would like to conduct an experiment to establish whether your friends prefer the taste of regular Coke or Diet Coke. Briefly outline a design for such a study.

1-7. *[5]* You would like to conduct an experiment to see whether students learn better if they study without any music, with instrumental music, or with songs that have lyrics. Briefly outline the design for such a study.

1-8. *[5]* Traditional polling operations like Gallup use a procedure called random digit dialing, which dials random strings of digits instead of picking phone numbers from the phone book. Suggest why such polls are conducted using random digit dialing.

Implementation Projects

1-9. *[5]* Write a program to scrape the best-seller rank for a book on Amazon.com. Use this to plot the rank of all of Skiena's books over time. Which one of these books should be the next item that you purchase? Do you have friends for whom they would make a welcome and appropriate gift? :-)

1-10. *[5]* For your favorite sport (baseball, football, basketball, cricket, or soccer) identify a data set with the historical statistical records for all major participants. Devise and implement a ranking system to identify the best player at each position.

Interview Questions

1-11. *[3]* For each of the following questions: (1) produce a quick guess based only on your understanding of the world, and then (2) use Google to find supportable numbers to produce a more principled estimate from. How much did your two estimates differ by?

(a) How many piano tuners are there in the entire world?

(b) How much does the ice in a hockey rink weigh?

(c) How many gas stations are there in the United States?

(d) How many people fly in and out of LaGuardia Airport every day?

(e) How many gallons of ice cream are sold in the United States each year?

(f) How many basketballs are purchased by the National Basketball Association (NBA) each year?

(g) How many fish are there in all the world's oceans?

(h) How many people are flying in the air right now, all over the world?

(i) How many ping-pong balls can fit in a large commercial jet?

(j) How many miles of paved road are there in your favorite country?

(k) How many dollar bills are sitting in the wallets of all people at Stony Brook University?

(l) How many gallons of gasoline does a typical gas station sell per day?

(m) How many words are there in this book?

(n) How many cats live in New York city?

(o) How much would it cost to fill a typical car's gas tank with Starbuck's coffee?

(p) How much tea is there in China?

(q) How many checking accounts are there in the United States?

1-12. *[3]* What is the difference between regression and classification?

1-13. *[8]* How would you build a data-driven recommendation system? What are the limitations of this approach?

1-14. *[3]* How did you become interested in data science?

1-15. *[3]* Do you think data science is an art or a science?

Kaggle Challenges

1-16. Who survived the shipwreck of the Titanic?
`https://www.kaggle.com/c/titanic`

1-17. Where is a particular taxi cab going?
`https://www.kaggle.com/c/pkdd-15-predict-taxi-service-trajectory-i`

1-18. How long will a given taxi trip take?
`https://www.kaggle.com/c/pkdd-15-taxi-trip-time-prediction-ii`

Chapter 2

Mathematical Preliminaries

> A data scientist is someone who knows more statistics than a computer scientist and more computer science than a statistician.
>
> – Josh Blumenstock

You must walk before you can run. Similarly, there is a certain level of mathematical maturity which is necessary before you should be trusted to do anything meaningful with numerical data.

In writing this book, I have assumed that the reader has had some degree of exposure to probability and statistics, linear algebra, and continuous mathematics. I have also assumed that they have probably forgotten most of it, or perhaps didn't always see the forest (why things are important, and how to use them) for the trees (all the details of definitions, proofs, and operations).

This chapter will try to refresh your understanding of certain basic mathematical concepts. Follow along with me, and pull out your old textbooks if necessary for future reference. Deeper concepts will be introduced later in the book when we need them.

2.1 Probability

Probability theory provides a formal framework for reasoning about the likelihood of events. Because it is a formal discipline, there are a thicket of associated definitions to instantiate exactly what we are reasoning about:

- An *experiment* is a procedure which yields one of a set of possible outcomes. As our ongoing example, consider the experiment of tossing two six-sided dice, one red and one blue, with each face baring a distinct integer $\{1, \ldots, 6\}$.

- A *sample space* S is the set of possible outcomes of an experiment. In our

S.S. Skiena, *The Data Science Design Manual,*
Texts in Computer Science, https://doi.org/10.1007/978-3-319-55444-0_2

dice example, there are 36 possible outcomes, namely

$$\begin{aligned} S = \{&(1,1),(1,2),(1,3),(1,4),(1,5),(1,6),(2,1),(2,2),(2,3),(2,4),(2,5),(2,6),\\ &(3,1),(3,2),(3,3),(3,4),(3,5),(3,6),(4,1),(4,2),(4,3),(4,4),(4,5),(4,6),\\ &(5,1),(5,2),(5,3),(5,4),(5,5),(5,6),(6,1),(6,2),(6,3),(6,4),(6,5),(6,6)\}.\end{aligned}$$

- An *event* E is a specified subset of the outcomes of an experiment. The event that the sum of the dice equals 7 or 11 (the conditions to win at craps on the first roll) is the subset

  $$E = \{(1,6),(2,5),(3,4),(4,3),(5,2),(6,1),(5,6),(6,5)\}.$$

- The *probability of an outcome* s, denoted $p(s)$ is a number with the two properties:

  - For each outcome s in sample space S, $0 \leq p(s) \leq 1$.
  - The sum of probabilities of all outcomes adds to one: $\sum_{s \in S} p(s) = 1$.

  If we assume two distinct fair dice, the probability $p(s) = (1/6) \times (1/6) = 1/36$ for all outcomes $s \in S$.

- The *probability of an event* E is the sum of the probabilities of the outcomes of the experiment. Thus

  $$p(E) = \sum_{s \in E} p(s).$$

  An alternate formulation is in terms of the *complement* of the event $\bar{E}$, the case when E does not occur. Then

  $$P(E) = 1 - P(\bar{E}).$$

  This is useful, because often it is easier to analyze $P(\bar{E})$ than $P(E)$ directly.

- A *random variable* V is a numerical function on the outcomes of a probability space. The function "sum the values of two dice" $(V((a,b)) = a+b)$ produces an integer result between 2 and 12. This implies a probability distribution of the values of the random variable. The probability $P(V(s) = 7) = 1/6$, as previously shown, while $P(V(s) = 12) = 1/36$.

- The *expected value* of a random variable V defined on a sample space S, $E(V)$ is defined

  $$E(V) = \sum_{s \in S} p(s) \cdot V(s).$$

All this you have presumably seen before. But it provides the language we will use to connect between probability and statistics. The data we see usually comes from measuring properties of observed events. The theory of probability and statistics provides the tools to analyze this data.

Probability theorists love independent events, because it simplifies their calculations. But data scientists generally don't. When building models to predict the likelihood of some future event B, given knowledge of some previous event A, we want as strong a dependence of B on A as possible.

Suppose I always use an umbrella if and only if it is raining. Assume that the probability it is raining here (event B) is, say, $p = 1/5$. This implies the probability that I am carrying my umbrella (event A) is $q = 1/5$. But even more, if you know the state of the rain you know exactly whether I have my umbrella. These two events are perfectly *correlated.*

By contrast, suppose the events were independent. Then

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A)P(B)}{P(B)} = P(A)$$

and whether it is raining has absolutely no impact on whether I carry my protective gear.

Correlations are the driving force behind predictive models, so we will discuss how to measure them and what they mean in Section 2.3.

2.1.3 Conditional Probability

When two events are correlated, there is a dependency between them which makes calculations more difficult. The *conditional probability* of A given B, $P(A|B)$ is defined:

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Recall the dice rolling events from Section 2.1.2, namely:

- Event A is that at least one of two dice be an even number.
- Event B is the sum of the two dice is either a 7 or an 11.

Observe that $P(A|B) = 1$, because *any* roll summing to an odd value must consist of one even and one odd number. Thus $A \cap B = B$, analogous to the umbrella case above. For $P(B|A)$, note that $P(A \cap B) = 9/36$ and $P(A) = 25/36$, so $P(B|A) = 9/25$.

Conditional probability will be important to us, because we are interested in the likelihood of an event A (perhaps that a particular piece of email is spam) as a function of some evidence B (perhaps the distribution of words within the document). Classification problems generally reduce to computing conditional probabilities, in one way or another.

Our primary tool to compute conditional probabilities will be *Bayes theorem*, which reverses the direction of the dependencies:

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

Often it proves easier to compute probabilities in one direction than another, as in this problem. By Bayes theorem $P(B|A) = (1 \cdot 9/36)/(25/36) = 9/25$, exactly

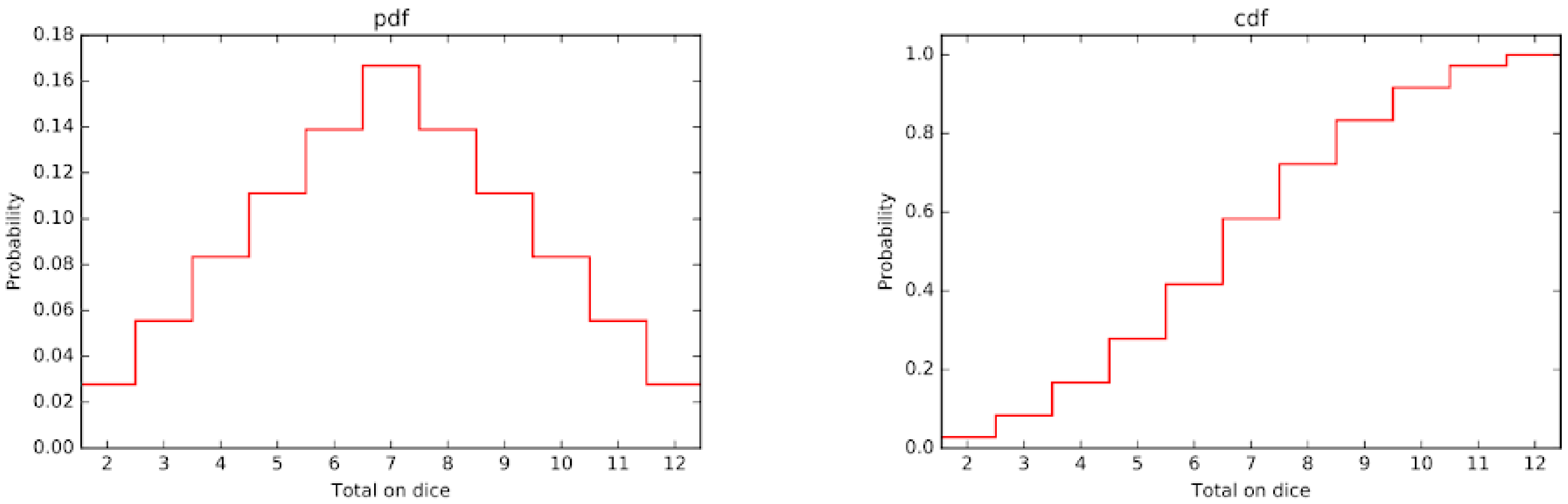


Figure 2.2: The probability density function (pdf) of the sum of two dice contains exactly the same information as the cumulative density function (cdf), but looks very different.

what we got before. We will revisit Bayes theorem in Section 5.6, where it will establish the foundations of computing probabilities in the face of evidence.

2.1.4 Probability Distributions

Random variables are numerical functions where the values are associated with probabilities of occurrence. In our example where $V(s)$ the sum of two tossed dice, the function produces an integer between 2 and 12. The probability of a particular value $V(s) = X$ is the sum of the probabilities of all the outcomes which add up to X.

Such random variables can be represented by their *probability density function*, or pdf. This is a graph where the x-axis represents the range of values the random variable can take on, and the y-axis denotes the probability of that given value. Figure 2.2 (left) presents the pdf of the sum of two fair dice. Observe that the peak at $X = 7$ corresponds to the most frequent dice total, with a probability of $1/6$.

Such pdf plots have a strong relationship to histograms of data frequency, where the x-axis again represents the range of value, but y now represents the observed frequency of exactly how many event occurrences were seen for each given value X. Converting a histogram to a pdf can be done by dividing each bucket by the total frequency over all buckets. The sum of the entries then becomes 1, so we get a probability distribution.

Histograms are statistical: they reflect actual observations of outcomes. In contrast, pdfs are probabilistic: they represent the underlying chance that the next observation will have value X. We often use the histogram of observations $h(x)$ in practice to estimate the probabilities[2] by normalizing counts by the total

[2] A technique called *discounting* offers a better way to estimate the frequency of rare events, and will be discussed in Section 11.1.2.

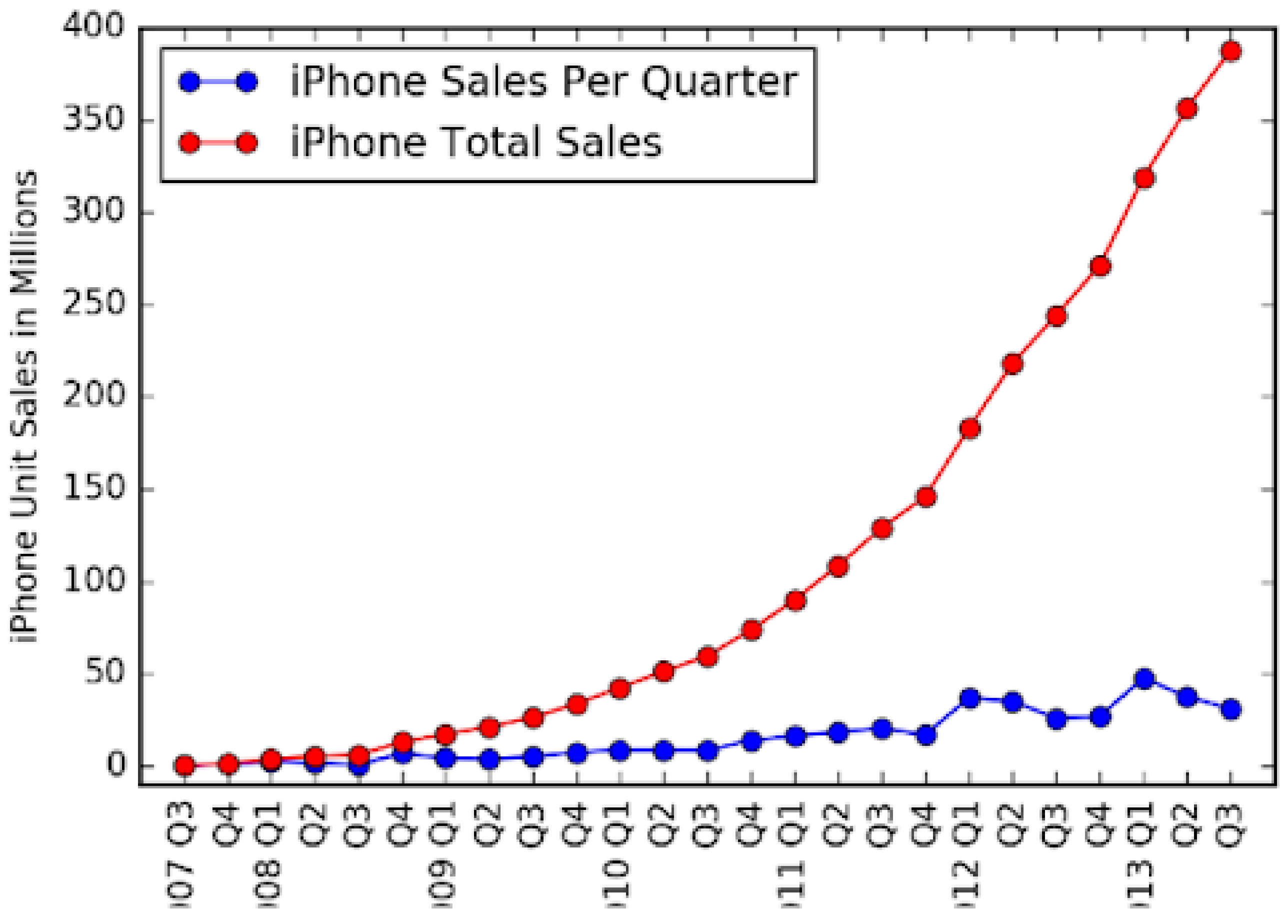


Figure 2.3: iPhone quarterly sales data presented as cumulative and incremental (quarterly) distributions. Which curve did Apple CEO Tim Cook choose to present?

number of observations:

$$P(k = X) = \frac{h(k = X)}{\sum_x h(x = X)}$$

There is another way to represent random variables which often proves useful, called a *cumulative density function* or cdf. The cdf is the running sum of the probabilities in the pdf; as a function of k, it reflects the probability that $X \leq k$ instead of the probability that $X = k$. Figure 2.2 (right) shows the cdf of the dice sum distribution. The values increase monotonically from left to right, because each term comes from adding a positive probability to the previous total. The rightmost value is 1, because all outcomes produce a value no greater than the maximum.

It is important to realize that the pdf $P(V)$ and cdf $C(V)$ of a given random variable V contain *exactly* the same information. We can move back and forth between them because:

$$P(k = X) = C(X \leq k + \delta) - C(X \leq k),$$

where $\delta = 1$ for integer distributions. The cdf is the running sum of the pdf, so

$$C(X \leq k) = \sum_{x \leq k} P(X = x).$$

Just be aware of which distribution you are looking at. Cumulative distributions always get higher as we move to the right, culminating with a probability of $C(X \leq \infty) = 1$. By contrast, the total area under the curve of a pdf equals 1, so the probability at any point in the distribution is generally substantially less.

An amusing example of the difference between cumulative and incremental distributions is shown in Figure 2.3. Both distributions show exactly the same data on Apple iPhone sales, but which curve did Apple CEO Tim Cook choose to present at a major shareholder event? The cumulative distribution (red) shows that sales are exploding, right? But it presents a misleading view of growth rate, because incremental change is the derivative of this function, and hard to visualize. Indeed, the sales-per-quarter plot (blue) shows that the rate of iPhone sales actually had declined for the last two periods before the presentation.

2.2 Descriptive Statistics

Descriptive statistics provide ways of capturing the properties of a given data set or sample. They summarize observed data, and provide a language to talk about it. Representing a group of elements by a new derived element, like mean, min, count, or sum reduces a large data set to a small summary statistic: aggregation as data reduction.

Such statistics can become features in their own right when taken over natural groups or clusters in the full data set. There are two main types of descriptive statistics:

- *Central tendency measures*, which capture the center around which the data is distributed.

- *Variation* or *variability measures*, which describe the data spread, i.e. how far the measurements lie from the center.

Together these statistics tell us an enormous amount about our distribution.

2.2.1 Centrality Measures

The first element of statistics we are exposed to in school are the basic centrality measures: mean, median, and mode. These are the right place to start when thinking of a single number to characterize a data set.

- *Mean*: You are probably quite comfortable with the use of the *arithmetic mean*, where we sum values and divide by the number of observations:

  $$\mu_X = \frac{1}{n} \sum_{i=1}^{n} x_i$$

  We can easily maintain the mean under a stream of insertions and deletions, by keeping the sum of values separate from the frequency count, and divide only on demand.

  The mean is very meaningful to characterize symmetric distributions without outliers, like height and weight. That it is symmetric means the number of items above the mean should be roughly the same as the number below. That it is without outliers means that the range of values is reasonably tight. Note that a single MAXINT creeping into an otherwise sound set of observations throws the mean wildly off. The median is a centrality measure which proves more appropriate with such ill-behaved distributions.

- *Geometric mean*: The *geometric mean* is the nth root of the product of n values:

  $$\left(\prod_{i=1}^{n} a_i\right)^{1/n} = \sqrt[n]{a_1 a_2 \ldots a_n}$$

  The geometric mean is always less than or equal to the arithmetic mean. For example, the geometric mean of the sums of 36 dice rolls is 6.5201, as opposed to the arithmetic mean of 7. It is very sensitive to values near zero. A single value of zero lays waste to the geometric mean: no matter what other values you have in your data, you end up with zero. This is somewhat analogous to having an outlier of ∞ in an arithmetic mean.

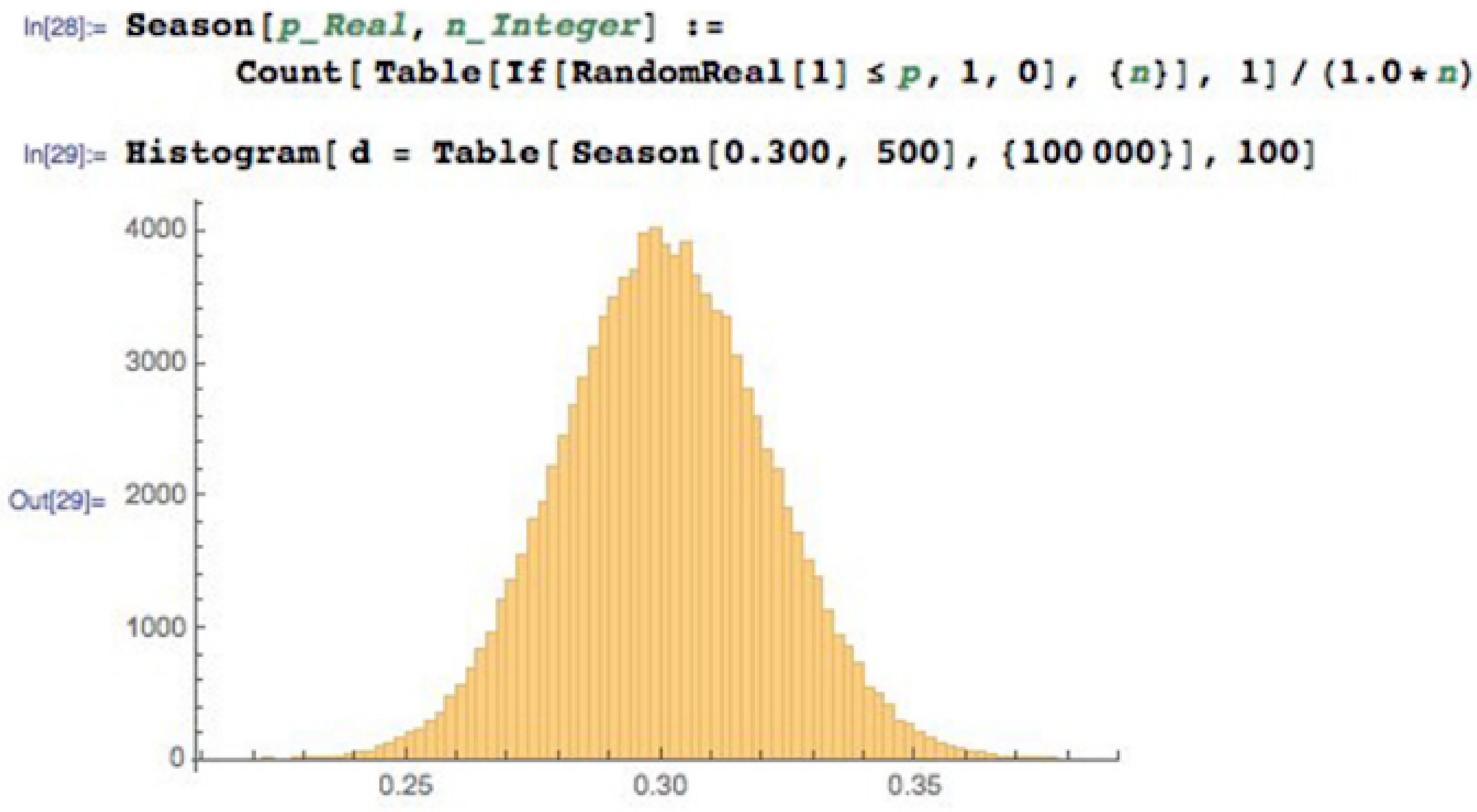
  But geometric means prove their worth when averaging ratios. The geometric mean of 1/2 and 2/1 is 1, whereas the mean is 1.25. There is less available “room” for ratios to be less than 1 than there is for ratios above 1, creating an asymmetry that the arithmetic mean overstates. The geometric mean is more meaningful in these cases, as is the arithmetic mean of the *logarithms* of the ratios.

- *Median*: The *median* is the exact middle value among a data set; just as many elements lie above the median as below it. There is a quibble about what to take as the median when you have an even number of elements. You can take either one of the two central candidates: in any reasonable data set these two values should be about the same. Indeed in the dice example, both are 7.

  A nice property of the median as so defined is that it must be a genuine value of the original data stream. There actually is someone of median height to you can point to as an example, but presumably no one in the world is of *exactly* average height. You lose this property when you average the two center elements.

  Which centrality measure is best for applications? The median typically lies pretty close to the arithmetic mean in symmetrical distributions, but it is often interesting to see how far apart they are, and on which side of the mean the median lies.

  The median generally proves to be a better statistic for skewed distributions or data with outliers: like wealth and income. Bill Gates adds $250 to the mean per capita wealth in the United States, but nothing to the median. If he makes you personally feel richer, then go ahead and use the mean. But the median is the more informative statistic here, as it will be for any power law distribution.



```
In[28]:= Season[p_Real, n_Integer] :=
            Count[ Table[If[RandomReal[1] ≤ p, 1, 0], {n}], 1] / (1.0 * n)

In[29]:= Histogram[ d = Table[ Season[0.300, 500], {100 000}], 100]
```

Figure 2.5: Sample variance on hitters with a real 30% success rate results in a wide range of observed performance even over 500 trials per season.

year usually underperforms the market the year after, which shouldn't happen if this outstanding performance was due to skill rather than luck.

The fund managers themselves are quick to credit profitable years to their own genius, but losses to unforeseeable circumstances. However, several studies have shown that the performance of professional investors is essentially random, meaning there is little real difference in skill. Most investors are paying managers for previously-used luck. So why do these entrail-readers get paid so much money?

- *Sports performance*: Students have good semesters and bad semesters, as reflected by their grade point average (GPA). Athletes have good and bad seasons, as reflected by their performance and statistics. Do such changes reflect genuine differences in effort and ability, or are they just variance?

  In baseball, .300 hitters (players who hit with a 30% success rate) represent consistency over a full season. Batting .275 is not a noteworthy season, but hit .300 and you are a star. Hit .325 and you are likely to be the batting champion.

  Figure 2.5 shows the results of a simple simulation, where random numbers were used to decide the outcome of each at-bat over a 500 at-bats/season. Our synthetic player is a *real* .300 hitter, because we programmed it to report a hit with probability 300/1000 (0.3). The results show that a real .300 hitter has a 10% chance of hitting .275 or below, just by chance. Such a season will typically be explained away by injuries or maybe the inevitable effects of age on athletic performance. But it could just be natural variance. Smart teams try to acquire a good hitter after a lousy season, when the price is cheaper, trying to take advantage of this variance.

  Our .300 hitter also has a 10% chance of batting above .325, but you can be pretty sure that they will ascribe such a breakout season to their improved conditioning or training methods instead of the fact they just got lucky. Good or bad season, or lucky/unlucky: it is hard to tell the signal from the noise.

- *Model performance*: As data scientists, we will typically develop and evaluate several models for each predictive challenge. The models may range from very simple to complex, and vary in their training conditions or parameters.

  Typically the model with the best accuracy on the training corpus will be paraded triumphantly before the world as the right one. But small differences in the performance between models is likely explained by simple variance rather than wisdom: which training/evaluation pairs were selected, how well parameters were optimized, etc.

  Remember this when it comes to training machine learning models. Indeed, when asked to choose between models with small performance differences between them, I am more likely to argue for the simplest model than the one with the highest score. Given a hundred people trying to predict heads and tails on a stream of coin tosses, one of them is guaranteed to end up with the most right answers. But there is no reason to believe that this fellow has any better predictive powers than the rest of us.

2.2.4 Characterizing Distributions

Distributions do not necessarily have much probability mass exactly at the mean. Consider what your wealth would look like after you borrow \$100 million, and then bet it all on an even money coin flip. Heads you are now \$100 million in clear, tails you are \$100 million in hock. Your expected wealth is zero, but this mean does not tell you much about the shape of your wealth distribution.

However, taken together the mean and standard deviation do a decent job of characterizing *any* distribution. Even a relatively small amount of mass positioned far from the mean would add a lot to the standard deviation, so a small value of σ implies the bulk of the mass must be near the mean.

To be precise, regardless of how your data is distributed, at least $(1 - (1/k^2))$th of the mass must lie within $\pm k$ standard deviations of the mean. This means that at least 75% of all the data must lie within 2σ of the mean, and almost 89% within 3σ for any distribution.

We will see that even tighter bounds hold when we know the distribution is well-behaved, like the Gaussian or normal distribution. But this is why it is a great practice to report both μ and σ whenever you talk about averages. The average height of adult women in the United States is 63.7 ± 2.7 inches, meaning $\mu = 63.7$ and $\sigma = 2.7$. The average temperature in Orlando, Fl is 60.3 degrees Fahrenheit. However, there have been many more 100 degree days at Disney World than 100 inch (8.33 foot) women visiting to enjoy them.

Take-Home Lesson: Report both the mean and standard deviation to characterize your distribution, written as $\mu \pm \sigma$.

2.3 Correlation Analysis

Suppose we are given two variables x and y, represented by a sample of n points of the form (x_i, y_i), for $1 \leq i \leq n$. We say that x and y are *correlated* when the value of x has some predictive power on the value of y.

The *correlation coefficient* $r(X, Y)$ is a statistic that measures the degree to which Y is a function of X, and vice versa. The value of the correlation coefficient ranges from -1 to 1, where 1 means fully correlated and 0 implies no relation, or independent variables. Negative correlations imply that the variables are *anti-correlated*, meaning that when X goes up, Y goes down.

Perfectly anti-correlated variables have a correlation of -1. Note that negative correlations are just as good for predictive purposes as positive ones. That you are less likely to be unemployed the more education you have is an example of a negative correlation, so the level of education can indeed help predict job status. Correlations around 0 are useless for forecasting.

Observed correlations drives many of the predictive models we build in data science. Representative strengths of correlations include:

- Are taller people more likely to remain lean? The observed correlation between height and BMI is $r = -0.711$, so height is indeed negatively correlated with body mass index (BMI).[3]

- Do standardized tests predict the performance of students in college? The observed correlation between SAT scores and freshmen GPA is $r = 0.47$, so yes, there is some degree of predictive power. But social economic status is just as strongly correlated with SAT scores ($r = 0.42$).[4]

- Does financial status affect health? The observed correlation between household income and the prevalence of coronary artery disease is $r = -0.717$, so there is a strong negative correlation. So yes, the wealthier you are, the lower your risk of having a heart attack.[5]

- Does smoking affect health? The observed correlation between a group's propensity to smoke and their mortality rate is $r = 0.716$, so for G-d's sake, don't smoke.[6]

[3] `https://onlinecourses.science.psu.edu/stat500/node/60`

[4] `https://research.collegeboard.org/sites/default/files/publications/2012/9/researchreport-2009-1-socioeconomic-status-sat-freshman-gpa-analysis-data.pdf`

[5] `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3457990/`.

[6] `http://lib.stat.cmu.edu/DASL/Stories/SmokingandCancer.html`.

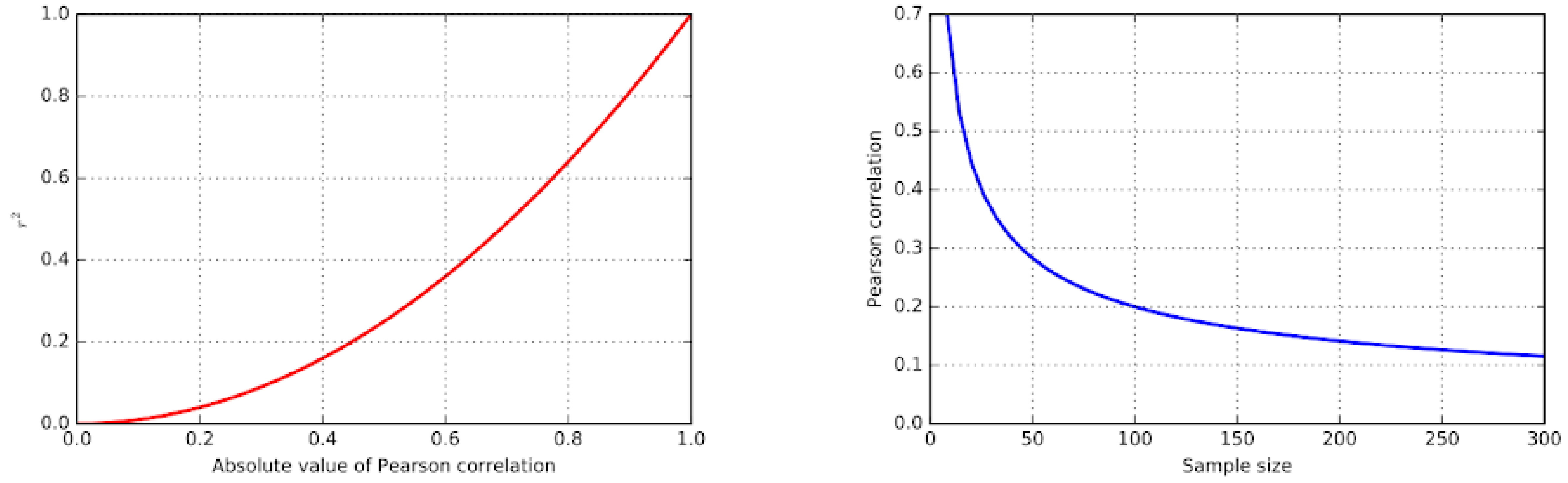


Figure 2.8: Limits in interpreting significance. The r^2 value shows that weak correlations explain only a small fraction of the variance (left). The level of correlation necessary to be statistically significance decreases rapidly with sample size n (right).

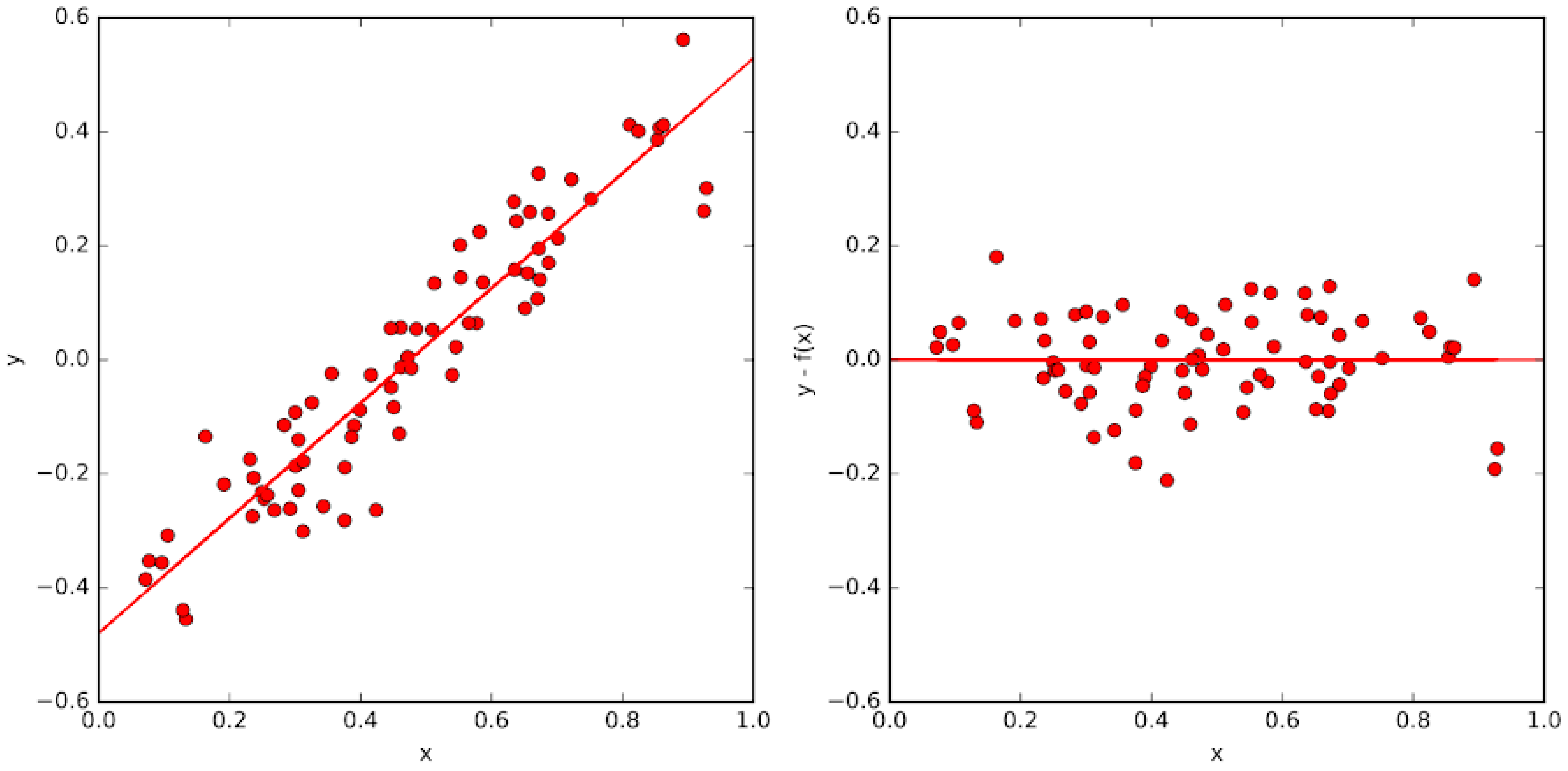


Figure 2.9: Plotting $r_i = y_i - f(x_i)$ shows that the residual values have lower variance and mean zero. The original data points are on the left, with the corresponding residuals on the right.