Teaching data science as an interdisciplinary activity (with R)

John Mount

Wednesday, August 21, 2013
6:30 PM

Intuit Building 9
2600 Casey Ave, Mountain View, CA

http://www.meetup.com/R-Users/events/132378792/

http://www.meetup.com/R-Users/events/132378792/

Or …

- ~~The grand architecture of my opinions and why I am always right.~~
- What I have learned using R with others in short duration data science projects and how we are trying to distill it into a book.

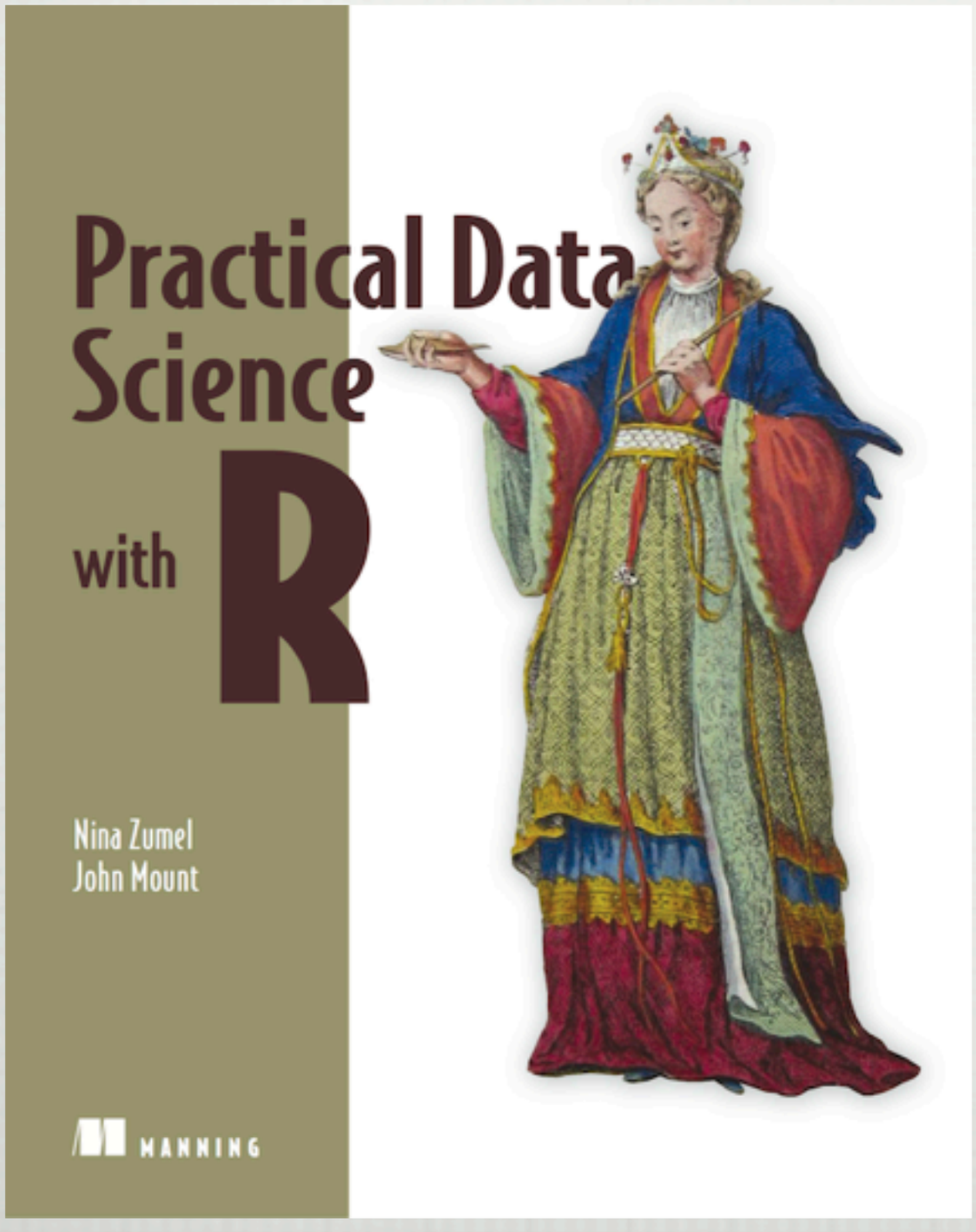
In my career I moved from a theoretical computer science and algorithms background to a data science short project “fire fighting” role.

A data scientist must interact competently with:

- ☐ Managers
- ☐ Business owners
- ☐ Project managers
- ☐ Analysts
- ☐ Database administrators / IT
- ☐ Programmers
- ☐ Statisticians
- ☐ Econometricians
- ☐ Machine learning experts

You need to be good (or at least not bad) in all of there roles. At the roles you are “at least not bad” at you need to interact and recruit expert partners in your organization. You can not expect a ready to go “de-normalized table” that you just have to waive a machine learning wand over.

We wanted to capture the flavor of this breadth of knowledge in our book



http://www.manning.com/zumel/

People are paying for this book, so we are writing about how to help them be better overall data scientists instead of writing about the mathematics of machine learning as we often do in the Win-Vector blog. Using R so we have a single language to work concrete examples with.

Outline of this talk

- ☐ Introduce the R and data science topics we were able to include in our book.
- ☐ Outline some of the difficulties in trying to produce the book.
- ☐ List broadly/shallowly the work methods and tools we advocate.
- ☐ Do a narrow/deep dive on SQL oriented data management.
- ☐ Wrap up with where we are on the book.

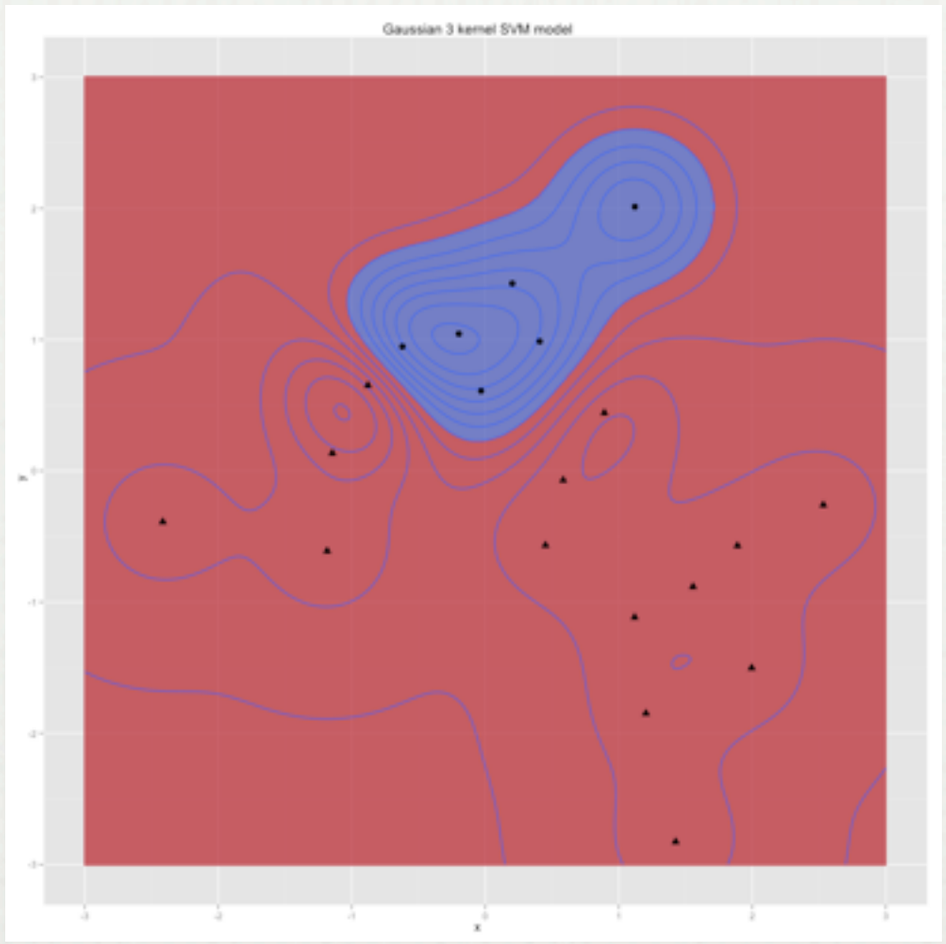
I apologize: I can't be both broad and deep at the same time. So we will mention a lot of topics in a shallow manner and then deep dive only into a simple data shaping problem. There are some serious digressions in store.

For scope: had to give up

- ☐ Non-R analysis platforms
- ☐ NoSQL
- ☐ Big Data
- ☐ Time series analysis
- ☐ Text/NLP processing
- ☐ Advanced machine learning
- ☐ Time series
- ☐ Econometric methods
- ☐ Traditional treatment of statistical concepts like analysis of variance (especially the compressed terminology)
- ☐ Teaching programming/ introducing R

Advanced machine learning: active learning, transductive learning, manifold regularization.
Signal analysis: compressed sensing. Still on the fence on big data: valuable curated data tends to be small to medium (as there is a cost per data row), very big data tends to be log or other un-vetted "data exhaust."

Yes, we can write about machine learning and draw pretty pictures



- It is just: we don't have room for that if the book has any chance of teaching the data science process.

Win-Vector LLC
www.win-vector.com

http://www.win-vector.com/blog/2011/10/kernel-methods-and-support-vector-machines-de-mystified/

The book is what we could not cut

- ☐ Managing data
- ☐ Exploring, visualizing and summarizing data
- ☐ Use of core machine learning and statistical methods for building predictive models
- ☐ Goodness of fit tests and significance
- ☐ Guides to work, interaction, documentation and presentation
- ☐ Tools and methods from software engineering (version control, distributed communications, and project management)

One idea is that significance and goodness of fit are so important that it is much more important to do them correctly than to do them quickly. Having to re-think your criteria and tests is not bad (but re-implementing tests is very bad).

Our current draft table of contents

Parts 1 and 3 spend a lot of time on interactions (defining problems and presenting results). Almost all chapters have explicit examples worked through in R. Chapter 5 in particular runs through some exercises on how to convert customer interviews into model scoring criteria. Over half the book is about interaction; I tried to make that the topic of this talk, but it just doesn't distill down that small and remain specific and meaningful.

Major difficulties with the book

- Need to assume reader is somewhat familiar with script programming and/or R.
- Many programmers don't want to learn the whole data science process, they want to implement new machine learning algorithms or use new machine learning libraries.
- The traditional presentation of statistics is quite the right set of topics for a data scientist.
- Very hard to find unencumbered examples that support the activities we want to demonstrate.

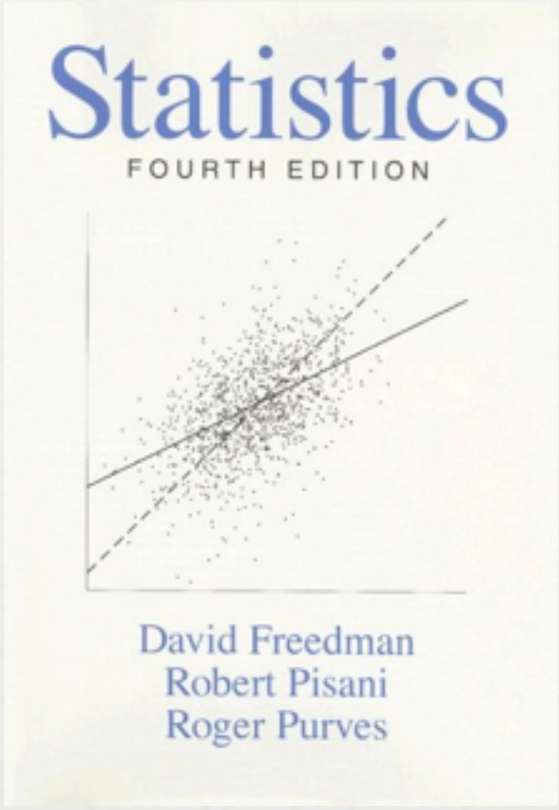
Strongly recommend: Kabacoff: "R in Action" and Freedman, Pisani, Purves: "Statistics." For deep knowledge on R-programming Matloff "The Art of R Programming" (it is just we don't suggest writing long programs in R). Topics that are not in the right emphasis are the various versions of ANOVA (easier to set up your experiment directly than to work through all of the variations), the central limit theorem (easier to teach concentration inequalities as probability theory), exploratory statistics (you can spend too long on mean and variance analysis). Also a lot more time on permutation tests and empirical re-sampling would be great.

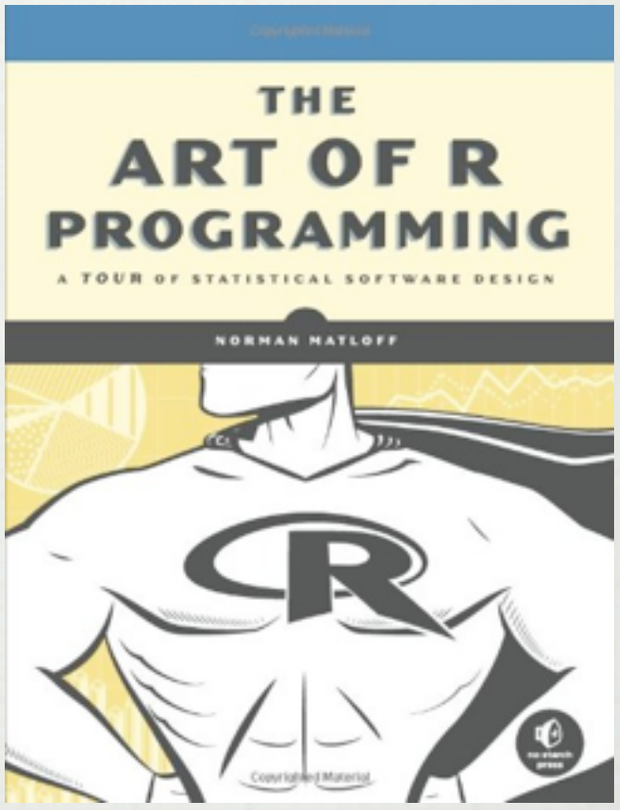
Difficulty: programming

- A big part of our target audience are analysts not familiar with programming.
- Learning programming for the first time is a major intellectual accomplishment.
  - Edsger W.Dijkstra: “... our intellectual powers are rather geared to master static relations and that our powers to visualize processes evolving in time are relatively poorly developed.”
  - I.e. program design involves simulating in the mind some parts of the state of a dynamic process. Therefor learning the programmer’s mindset is harder than learning a topic like logic.
- Programming is an essential part of data science.
  - In data science you are always on your penultimate analysis. You always have to re-run one more time (more data, change in data, need to cross-validate, bug fixes, changes in requirements and so on). Because you have repetition you *must* completely automate.

Fred Brooks in “No Silver Bullet” points out the distinctions between inessential difficulties (which can be eliminated) and essential difficulties which must be worked with.

Prerequisites: You need access to one or more of these





Difficulty: non-canon presentation of statistics

- Statistics is a specialized field that needs concise and specialized nomenclature.
- However, as Rota wrote: you have to go through a *lot* of references before you find one that lowers itself to saying why a non-statistician should be interested in a technique. An example from Rota (Indiscrete Thoughts):

  " "A company wishes to purchase one of five different machines: A, B, C, D, or E. In an experiment designed to test whether there is difference in the machines' performance, each of five experienced operators works on each of the machines for equal times. Table 126.11 shows the numbers of units produced per machine. Test the hypothesis that there is no difference between the machines at significance levels (a) 0.05 and (b) 0.01."

  After reading this paragraph, I knew what the analysis of variance is. I also realized why students of statistics in France, Italy, and other countries systematically shun the pompous and incomprehensible texts of the local professoriate in favor of Murray Siegel's [Schaums Outline of Statistics]."
- Some of the standard nomenclature is so specialized (independent variables for inputs) that it is not compatible with mathematical English or colloquial English.

Data scientists are hired to be broad optimists, statisticians are hired to be correct specialists (which can sound pessimistic).

Difficulty: Academic Priority has no limit on complexity

- Example: computing the degree of correlation of two taggers.
- If we were to be completely correct and insist on using a technique “by name” from the literature we have over 28 different calculations to choose from (“On similarity indices and correction for chance agreement” AN Albatineh, M Niewiadomska-Bugaj, and D Mihalko, Journal of classification, 2006 vol. 23 (2) pp. 301-313).
- Better to work out what one is trying to test and develop adapt a simple measure that fits the business application. As long as it works for the data at hand it is okay if it doesn’t fix a problem we have not yet encountered.

Lesson: keep score and significance issues separate when possible (things like AIC do attempt to combine them). With enough data significance can be estimate with empirical resampling, permutation tests or bootstrap style methods.

Example: 22 non-identical correspondence scores from 28 found in a literature review

No.	Symbol	Formula	Range	$\mathcal{L}$	$\mathcal{H}$
1	R	$\frac{a+d}{a+b+c+d}$	[0,1]	+	+
2	H	$\frac{(a+d)-(b+c)}{a+b+c+d}$	[-1,1]	+	+
3	CZ	$\frac{2a}{2a+b+c}$	[0,1]	+	+
4	K	$\frac{1}{2}\left(\frac{a}{a+b}+\frac{a}{a+c}\right)$	[0,1]	+	+
5	MC	$\frac{a^2-bc}{(a+b)(a+c)}$	[-1,1]	+	+
6	PE	$\frac{ad-bc}{(a+c)(b+d)}$	[-1,1]	+	+
7	FM	$\frac{a}{\sqrt{(a+b)(a+c)}}$	[0,1]	+	+
8	$W1$	$\frac{a}{a+b}$	[0,1]	+	+
9	$W2$	$\frac{a}{a+c}$	[0,1]	+	+
10	Γ	$\frac{ad-bc}{\sqrt{(a+b)(a+c)(c+d)(b+d)}}$	[-1,1]	+	+
11	$SS1$	$\frac{1}{4}\left(\frac{a}{a+b}+\frac{a}{a+c}+\frac{d}{d+b}+\frac{d}{d+c}\right)$	[0,1]	+	+
12	$B1$	$\frac{\binom{m}{2}^2-\binom{m}{2}(b+c)+(b-c)^2}{\binom{m}{2}^2}$	[0,1]	+	+
13	RR	$\frac{a}{a+b+c+d}$	[0,1]	+	−
14	FMG	$\frac{a}{\sqrt{(a+b)(a+c)}}-\frac{1}{2\sqrt{(a+b)}}$	$[-\frac{1}{2},1)$	+	−
15	P	$\frac{ad-bc}{(a+b)(a+c)(c+d)(b+d)}$	[-1,1]	+	−
16	$B2$	$\frac{ad-bc}{\binom{m}{2}^2}$	$[-\frac{1}{4},\frac{1}{4}]$	+	−
17	J	$\frac{a}{a+b+c}$	[0,1]	−	+
18	$SS2$	$\frac{a}{a+2(b+c)}$	[0,1]	−	+
19	$SS3$	$\frac{ad}{\sqrt{(a+b)(a+c)(d+b)(d+c)}}$	[0,1]	−	+
20	GL	$\frac{a+d}{a+\frac{1}{2}(b+c)+d}$	[0,1]	−	+
21	RT	$\frac{a+d}{a+2(b+c)+d}$	[0,1]	−	+
22	GK	$\frac{ad-bc}{ad+bc}$	[-1,1]	−	+

This is where you end up if allow the discourse to degenerate to “how could you be so stupid to use Pearson’s product moment correlation, oh I guess you don’t know about Cohen’s kappa.” All counts number of pairs of items: a both say in same cluster; b,c one groups other does not; d both do not group. Later papers reduce these to a smaller number of clusters that share a monotone relation.

Difficulty: statistics is right

- Data science is, compared to statistics obsessed with prediction and not fully engaged in issues of inference.
- Data science has the luxury of assuming medium to large data sets, allow us to ignore many issues of statistical efficiency and also estimate significance through empirical re-sampling.
- Finally: you must study statistics to get treatments of confidence, significance, credibility and posteriors that are sufficiently rigorous to support serious work.

Difficulty: finding relevant examples

- Classic machine learning problem archives (such as the UCI Machine Learning Repository) concentrate on examples specifically chosen to justify specific machine learning techniques. Some of the examples are even synthetic data (such as “Car Evaluation,” though it is not completely obvious from its description).

- Most prominent sources of examples (Kaggle, KDDCup) abstract out exactly the steps we would call data science (leaving only the much more specific machine learning steps undone).

- Many public sources of data (Census, CDC, Terrorism, Airline, Crime, Property Tax, and so on) don’t come with obvious example business problems (i.e. they have all the “x”s and none of the “y”s).

Difficulty: examples cont.

- http://www.kdd.org/kdd-cup-2009-customer-relationship-prediction
  - Description: “large marketing databases from the French Telecom company Orange to predict the propensity of customers to switch provider (churn), buy new products or services (appetency), or buy upgrades or add-ons proposed to them to make the sale more profitable (up-selling).”
  - Data: 50,000 rows of 230 columns named “Var1” .. “Var230”. Most values null. No clue which missing values are fair to use and which are not. No data dictionary, explanation of levels in categorical variables, or explanation of conditions of validity of variables.
  - Goal: maximize AUC, a machine learning metric- but not a metric that typically maps directly to a business need (like precision, recall, sensitivity or specificity).
- For all that, one of the better data sets available. We were happy to use it on our book.

Enough negativity, what tools do we have available to help us?

Common tools

- Tools we insist on:
  - R (R, RStudio, or Revolution R)
  - SQL (Postgresql, H2, SquirreL SQL)
  - git
  - an editor capable of repairing data files (emacs, TextWrangler)
- Tools clients insist on:
  - noSQL
  - Jira
  - Confluence
  - Excel
  - Hadoop
- Tools nobody insists on, but that tend to dominate if not reigned in
  - Word
  - email

Notable R packages

- data import
  - RJDBC/DBI
  - xlsx
- data manipulation
  - reshape2
  - sqldf
- graphics
  - ggplot2
- machine learning “magic bullets”
  - lm
  - glm
  - gam
  - rpart
  - randomForest

One task/tool in detail

- We often say 90% of data science is data tubing and feature engineering.
- I am going to break tradition and actually spend some time describing how to move data around (instead of talking about machine learning).

A fundamental data operation is moving between fat and thin forms

- Analysts: Excel and Pivot tables
- R: reshape2 melt() and cast()
- SQL (don't bother, SQL *hates* data determined columns)
  - Oracle: Pivot/UnPivot
  - Postgresql: crosstab

Example Fat Data

	A	B	C	D	E
1	Number of reservations on the book				
2		reservations			
3	date	day of stay	1 before	2 before	3 before
4	7/1/13	105	98	95	96
5	7/2/13	103	100	98	95
6	7/3/13	105	95	90	80
7	7/4/13	105	105	107	98

	A	B	C	D	E
1	Published price				
2		prices			
3	date	day of stay	1 before	2 before	3 before
4	7/1/13	$250.00	$200.00	$280.00	$300.00
5	7/2/13	$200.00	$250.00	$290.00	$250.00
6	7/3/13	$200.00	$200.00	$250.00	$275.00
7	7/4/13	$250.00	$300.00	$300.00	$200.00

We have our data, we would like to explore how price affects booking pickups.

I also call this kind of spreadsheet “ascii art” because a lot of the meaning of the sheet is encoded in the relative arrangement of the cells (beyond just representing an orderly data frame).

R-steps to load and move to a thin form

```
library('xlsx')
bookings <- read.xlsx('Workbook1.xlsx',1,startRow=3)
prices <- read.xlsx('Workbook1.xlsx',2,startRow=3)

library('reshape2')
bthin <- melt(bookings,id.vars=c('date'),
   variable.name='daysBefore',value.name='bookings')
pthin <- melt(prices,id.vars=c('date'),
   variable.name='daysBefore',value.name='price')
```

```
> print(bookings)
        date day.of.stay X1.before X2.before X3.before
1 2009-06-30         105        98        95        96
2 2009-07-01         103       100        98        95
3 2009-07-02         105        95        90        80
4 2009-07-03         105       105       107        98
> print(prices)
        date day.of.stay X1.before X2.before X3.before
1 2009-06-30         250       200       280       300
2 2009-07-01         200       250       290       250
3 2009-07-02         200       200       250       275
4 2009-07-03         250       300       300       200
```

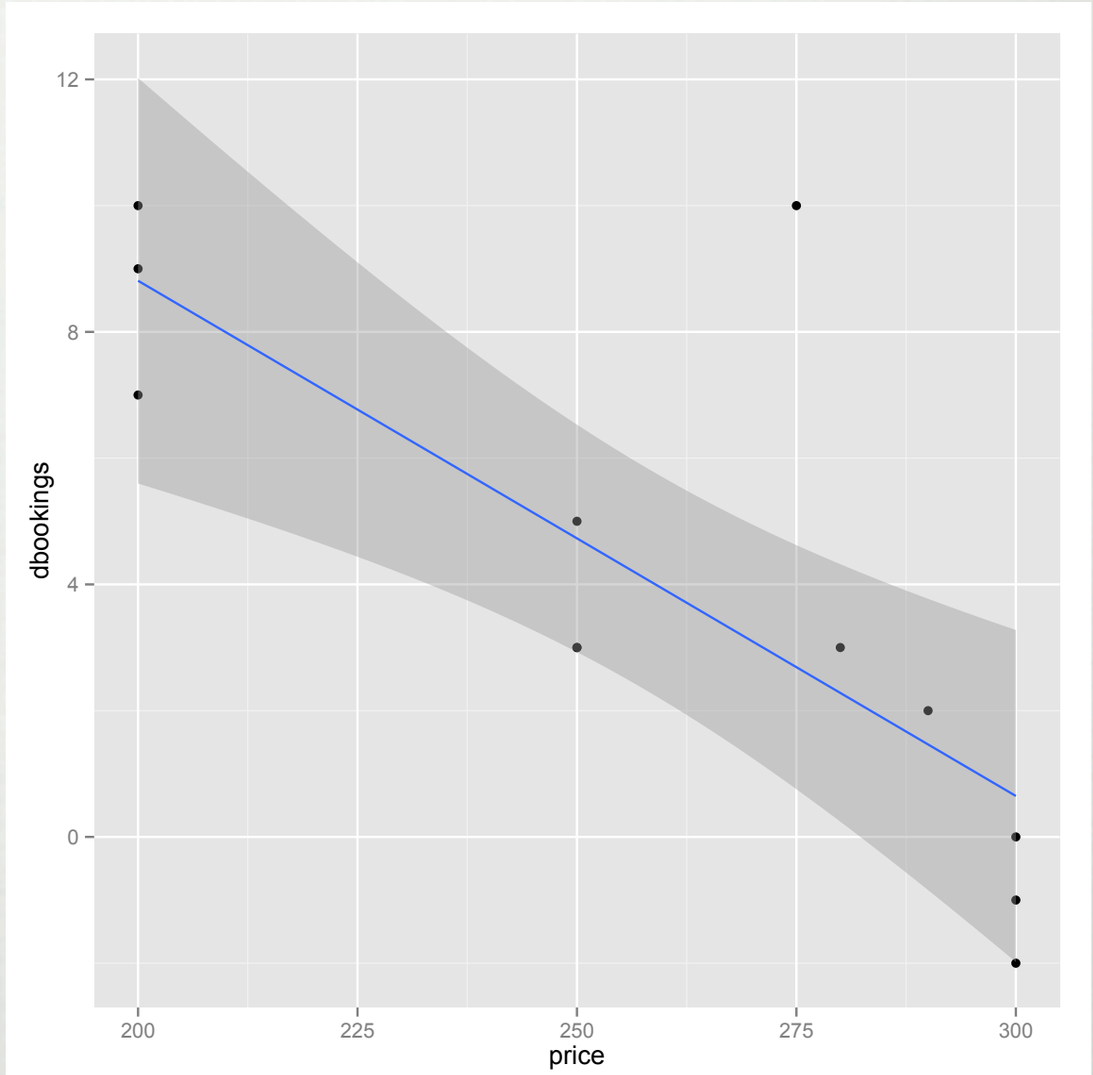
```
> print(pthin)
         date  daysBefore price
1  2009-06-30 day.of.stay   250
2  2009-07-01 day.of.stay   200
3  2009-07-02 day.of.stay   200
4  2009-07-03 day.of.stay   250
5  2009-06-30   X1.before   200
6  2009-07-01   X1.before   250
7  2009-07-02   X1.before   200
8  2009-07-03   X1.before   300
9  2009-06-30   X2.before   280
10 2009-07-01   X2.before   290
11 2009-07-02   X2.before   250
12 2009-07-03   X2.before   300
13 2009-06-30   X3.before   300
14 2009-07-01   X3.before   250
15 2009-07-02   X3.before   275
16 2009-07-03   X3.before   200
> print(bthin)
         date  daysBefore bookings
1  2009-06-30 day.of.stay      105
2  2009-07-01 day.of.stay      103
3  2009-07-02 day.of.stay      105
4  2009-07-03 day.of.stay      105
5  2009-06-30   X1.before       98
6  2009-07-01   X1.before      100
7  2009-07-02   X1.before       95
8  2009-07-03   X1.before      105
9  2009-06-30   X2.before       95
10 2009-07-01   X2.before       98
11 2009-07-02   X2.before       90
12 2009-07-03   X2.before      107
13 2009-06-30   X3.before       96
14 2009-07-01   X3.before       95
15 2009-07-02   X3.before       80
16 2009-07-03   X3.before       98
```

R-steps continued: join the data and perform an analysis

```
daysCodes <- c('day.of.stay', 'X1.before', 'X2.before',
'X3.before')
bthin$nDayBefore <- match(bthin$daysBefore,daysCodes)
pthin$nDayBefore <- match(pthin$daysBefore,daysCodes)

library('sqldf')
joined <- sqldf('
  select
     b1.date,
     b1.daysBefore as daysBefore1,
     b1.bookings as bookings1,
     b2.daysBefore as daysBefore2,
     b2.bookings as bookings2,
     p.price
  from
     bthin b1
  join
     bthin b2
  on
     b1.date=b2.date
     and b1.nDayBefore+1=b2.nDayBefore
  join
     pthin p
  on
     b1.date=p.date
     and b2.nDayBefore=p.nDayBefore
')

library('ggplot2')
joined$dbookings <- joined$bookings1 - joined$bookings2
ggplot(data=joined,aes(x=price,y=dbookings)) +
  geom_point() + geom_smooth(method='lm')
```



Overall strategy

- Load data into R with minimal hand steps (prefer reading Excel worksheets to exporting and loading).
- Do not attempt any work on “fat tables” which are tables where many facts about the same dimension/key are in a single row.
- Use reshape2 melt() to convert to thin tables.
- Use sqldf joins to build the tables you want (in preference to reshape2 cast(), aggregate, table and other commands).
- Notice we spent almost all of our time on data manipulation (as to spending time on characterization, visualization, and modeling).

Designing SQL queries

- SQL queries are very verbose and formidable looking. The most important operator join can be used to:
  - Intersect tables
  - Align tables to add extra columns
  - Perform cross products that create many new rows from original tables.
- Unfortunately it is not always easy to read the intent from the join syntax.
  - Best to design them up in terms of smaller steps that describe the intent of the join
  - Strongly suggest building a small cheat sheet of useful join patterns and also document join intent before writing SQL.

Example plan

- ☐ Want to plot picked up bookings (change in bookings) as a function of price.
- ☐ Need to self-join bookings from one day to next to be able to compute deltas.
- ☐ Need to join bookings to price to see the relation

Things to check on each join: what keys define rows we expect, are we okay with rows disappearing, are we okay with rows expanding in a cross product?

NoServer instead of NoSQL

- A lot of data transformations that would seem to require truly horrific code are expressed very naturally in SQL. The SQL may be verbose but concepts like join, self-join are natural steps.
- You don't need a database server to use SQL. sqldf supplies SQL functionality (using SQLite). We also suggest using Postgresql or H2 for tens of gigabytes scale data. With H2 you can use SQLScrewdriver to load large tables and SQuirreL SQL to practice ad-hoc database queries.
- This strategy works great in the tens of gigabytes range. We have (for the book) posted conversions of many public databases from ad-hoc to machine readable form on github: https://github.com/WinVector/zmPDSwR .

Valuable table tends to be small to medium (in the tens of gigabytes range) as for non-log data there is usually a non-negligible cost per row to make the data. SQL may seem ugly, but the sorting, indexing, mapping and iteration code you would have to write to get similar effect is far worse.

Aside: our tools can be brittle

- Each of the tools we used in our example breaks under a small variation of our data treatment.
  - POSIXlt (list style) dates break melt()
  - Date types can drift (and change day) when put through sqldf().
  - xlsx can fail on moderate sized spreadsheets.
  - ggplot2 loess curve hates our example data.
- The thing is: not just one of our tools nearly broke, all of our tools nearly broke.

This sort of thing makes one a small bit guilty about leaving R code maintenance to junior partners.

Aside: dates and melt()

- Problem: (lists hiding in a data frame column)

```
> d2 <- data.frame(Date=c('6/1 Sat 2013'),X0=c(99),X1=c(99))
> d2$Date <- c(as.POSIXlt(strptime(d2$Date,'%m/%d %a %Y')))
> melt(d2,id.vars=c('Date'),measure.vars=c('X0','X1'))
Error in data.frame(ids, variable, value, stringsAsFactors = FALSE) :
  arguments imply differing number of rows: 1, 2
```

- Solution: use POSIXct (non-list) or your own version of melt()

```
# for each set of values of id.vars plus name from measure.vars
# create a new data row.  This row contains columns id.vars, measure.name
# (taking the column name from measure.vars) and value.name (
# (taking the value from the measure.vars column matching value.name)
# returned data frame has number of rows equal to original data frame
# times the number of columns selected as measure vars.
# example:
# > print(d2)
#         Date X0 X1
# 1 2013-06-01 99 99
# > print(toThinForm(data=d2,id.vars=c('Date'),measure.vars=c('X0','X1'),
# +    measure.name='XVal',value.name='xunits'))
#         Date XVal xunits
# 1 2013-06-01   X0     99
# 2 2013-06-01   X1     99

toThinForm <- function(data,id.vars,measure.vars,measure.name,value.name) {
   dm <- c()
   cols <- c(id.vars,c(measure.name,value.name))
   for(mv in measure.vars) {
      di <- data[,c(id.vars),drop=F]
      di$MKEY=mv
      di$MV=data[,mv]
      colnames(di) <- cols
      dm <- rbind(dm,di)
   }
   dm
}
```

Aside: dates and sqldf()

- Problem: dates can drift with taken through sqldf.
  - POSIXct and POSIXlt do not represent dates. They represent date plus time with respect to an unspecified time zone.
- Solution: Do not use true date types with sqldf. The conversion to and for R representation to database representation involves a lot of details you probably don't want to bother with. Convert dates, as we did, into strings and integer offsets to safely cary the information.

This is the same position as in my TSV work (http://www.win-vector.com/blog/2012/12/please-stop-using-excel-like-formats-to-exchange-data/). It is prohibitively expensive to fully model all of the incompatible quoting and conversion conventions needed to prevent data damage when moving arbitrary data among a large number of tools. Instead simplify the data by introducing an early small amount of damage that obviates the need for complicated quoting and escaping.

Aside: there are spreadsheets that stump xlsx.

- Problem: Modern Excel formats are hideously complicated, so packages like xlsx perform a lot of work to extract tabular data from Excel files.
  - Eventually the complexity overcomes reasonable implementations.
  - Example: http://www.start.umd.edu/gtd/ gtd_201210dist.zip

```
> library('xlsx')
> .jinit(parameters="-Xmx2G")
> gtd <- read.xlsx('globalterrorismdb_1012dist.xlsx',1)
Error in .jcall("RJavaTools", "Ljava/lang/Object;", "invokeMethod", cl,  :
  java.lang.OutOfMemoryError: Java heap space
```

- Solution: move away form Excel like formats, especially for machine generated data extracts. Prefer DB dump formats, JSON, and strong TSV. Or better yet: DB connections.

Aside: conclusion

- Users of open source packages can be ungrateful (sorry).
- Fragility of tools does affect value of delivered projects.
- Because R is not a locked-down environment you can fix or work around any such problems.

Back to the book

Where we are with the book

- Eight chapters submitted to Manning Publications for review.
- Book front-mater freely available on the Win-Vector blog:
  - http://www.win-vector.com/blog/2013/06/what-is-practical-data-science-with-r/
- One chapter available for free review.
- Four more chapters available for subscription reading.
- Book to go into production at the end of 2013 and be available in print form early 2014.

Who the book is for

- ☐ People who want to work with or work as data scientists.
- ☐ Anyone interested in working through a number of data analysis examples using R (and the packages I have mentioned).

Thank you