Welcome to Data Bootcamp

Joseph Adler, Drew Conway, Jake Hofman, Hilary Mason

February 1, 2011

Speakers

@jadler, @hmason, @drewconway, @jakehofman

Joseph Adler

LinkedIn

Joseph Adler has many years of experience in data mining and data analysis at companies including DoubleClick, American Express, and VeriSign. He graduated from MIT with an B.Sc. and M.Eng in Computer Science and Electrical Engineering. He is the inventor of several patents for computer security and cryptography, and the author of "Baseball Hacks" and "R in a Nutshell". Currently, he is a senior data scientist at LinkedIn.

Hilary Mason

bit.ly

Hilary is the lead scientist at bit.ly, where she is finding sense in vast data sets. She is a former computer science professor with a background in machine learning and data mining, has published numerous academic papers, and regularly releases code on her personal site, www.hilarymason.com. She has discovered two new species, loves to bake cookies, and asks way too many questions.

▶ Web site

Drew Conway

New York University

Drew Conway is a PhD student in political science at New York University. Drew studies terrorism and armed conflict; using tools from mathematics and computer science to gain a deeper understanding of these phenomena.

▶ Web site

Jake Hofman

Yahoo!

Jake Hofman is a member of the Human Social Dynamics group at Yahoo! Research. His work involves data-driven modeling of social data, focusing on applications of machine learning and statistical inference to large-scale data. He holds a B.S. in Electrical Engineering from Boston University and a Ph.D. in Physics from Columbia University.

Please *do* try this at home

All of the materials from today's tutorial are available on Github:

Clone the repository for data/code/slides

```
git clone https://github.com/drewconway/strata_bootcamp
```

Disclaimer

You may be bored if you already know how to ...

- Acquire data from APIs
- Clean/explore/visualize data
- Classify and cluster image and text data
- Work with large data sets
- Build simple mashups
- Work with unix tools on the command line, e.g.

```
$ sed -e 's/<[^>]*>//g' < page.html > page.txt
```

Data-dependent products

Data-dependent products

- Effective/practical systems that learn from experience impact our daily lives, e.g.:
  - Recommendation systems
  - Spam detection
  - Optical character recognition
  - Face recognition
  - Fraud detection
  - Machine translation
  - ...

Black[1]-boxified?

Google code labs e.g. "ajax apis" or "open source" Search

Watch Google I/O keynotes live on May 19 and 20!

Google Prediction API (Labs)

Home Docs FAQ Forum Terms

What is the Google Prediction API?

The Prediction API enables access to Google's machine learning algorithms to analyze your historic data and predict likely future outcomes. Upload your data to Google Storage for Developers, then use the Prediction API to make real-time decisions in your applications. The Prediction API implements supervised learning algorithms as a RESTful web service to let you leverage patterns in your data, providing more relevant information to your users. Run your predictions on Google's infrastructure and scale effortlessly as your data grows in size and complexity.

Features

- Lightweight RESTful API
- Asynchronous training
- Automatically selects from several available machine learning techniques
- Supported inputs: numeric data or unstructured text
- Outputs hundreds of discrete categories
- Accessible from many platforms: Google App Engine, Apps Script (Google Spreadsheets), web & desktop apps, and command line

Uses

- Language identification
- Customer sentiment analysis
- Product recommendations & upsell opportunities
- Message routing decisions
- Diagnostics
- Document and email classification
- Suspicious activity identification
- Churn analysis
- And many more...

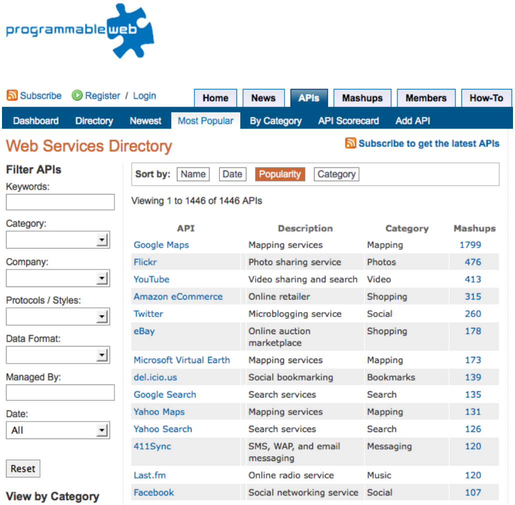
How do I start?

- Learn more about Google Prediction API.
- Request access.
- Try out the sample code.

[1] s/black/blue/g

Got data?

- Web service APIs expose lots of data



programmableweb

Subscribe Register / Login

Home News APIs Mashups Members How-To

Dashboard Directory Newest Most Popular By Category API Scorecard Add API

Web Services Directory

Subscribe to get the latest APIs

Filter APIs

Keywords:

Category:

Company:

Protocols / Styles:

Data Format:

Managed By:

Date:

All

Reset

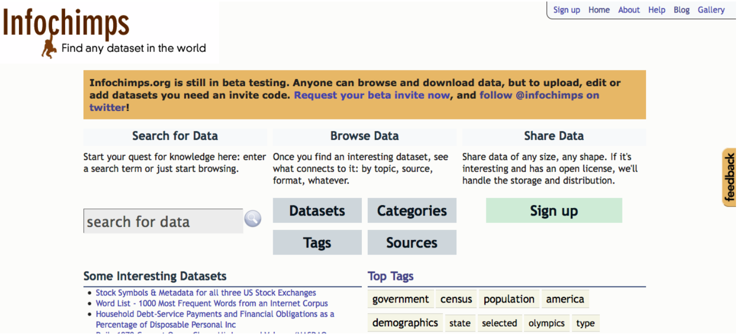
View by Category

Sort by: Name Date Popularity Category

Viewing 1 to 1446 of 1446 APIs

API	Description	Category	Mashups
Google Maps	Mapping services	Mapping	1799
Flickr	Photo sharing service	Photos	476
YouTube	Video sharing and search	Video	413
Amazon eCommerce	Online retailer	Shopping	315
Twitter	Microblogging service	Social	260
eBay	Online auction marketplace	Shopping	178
Microsoft Virtual Earth	Mapping services	Mapping	173
del.icio.us	Social bookmarking	Bookmarks	139
Google Search	Search services	Search	135
Yahoo Maps	Mapping services	Mapping	131
Yahoo Search	Search services	Search	126
411Sync	SMS, WAP, and email messaging	Messaging	120
Last.fm	Online radio service	Music	120
Facebook	Social networking service	Social	107

- Many free, public data sets available online

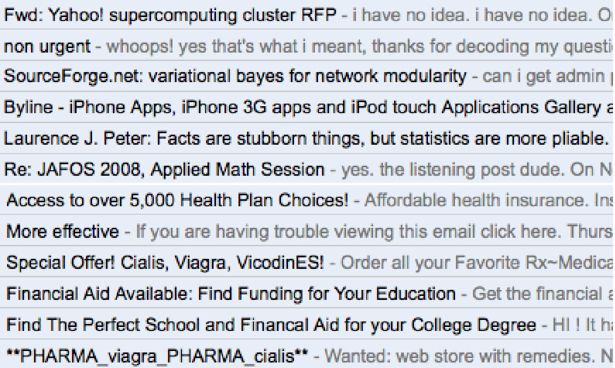


Step 1: Have data

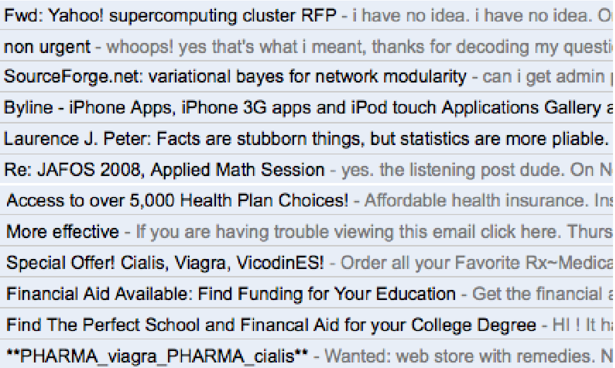
Step 2: ???

Step 3: Profit

Learning by example



Learning by example



- How did you solve this problem?
- Can you make this process explicit (e.g. write code to do so)?

- We learn quickly from few, relatively unstructured examples ... but we don't understand *how* we accomplish this

Everything old is new again[2]

- Many fields ...
  - Statistics
  - Pattern recognition
  - Data mining
  - Machine learning
- ... similar goals
  - Extract and recognize patterns in data
  - Interpret or explain observations
  - Test validity of hypotheses
  - Efficiently search the space of hypotheses
  - Design efficient algorithms enabling machines to learn from data

[2] http://cbcl.mit.edu/publications/theses/thesis-rifkin.pdf

Statistics vs. machine learning[3]

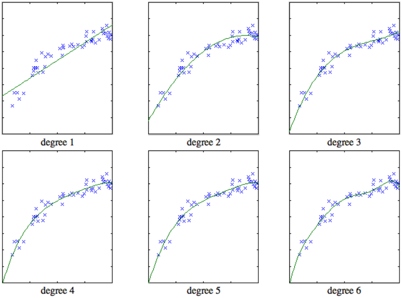
Glossary

Machine learning	Statistics
network, graphs	model
weights	parameters
learning	fitting
generalization	test set performance
supervised learning	regression/classification
unsupervised learning	density estimation, clustering
large grant = $1,000,000	large grant= $50,000
nice place to have a meeting: Snowbird, Utah, French Alps	nice place to have a meeting: Las Vegas in August

[3] http://anyall.org/blog/2008/12/statistics-vs-machine-learning-fight/

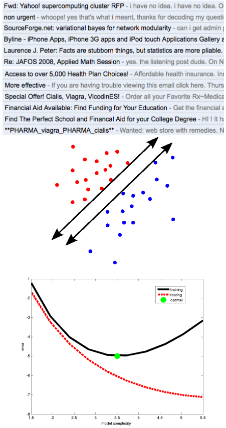
Philosophy

- We would like models that:
  - Provide predictive and explanatory power
  - Are complex enough to describe observed phenomena
  - Are simple enough to generalize to future observations

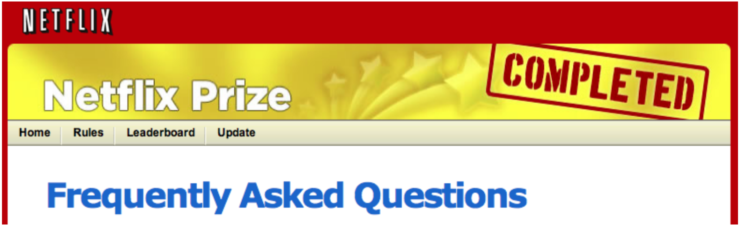


Roadmap, take 2[4]

1. Get data
2. Visualize/perform sanity checks
3. Clean/filter observations
4. Choose features to represent data
5. Specify model
6. Specify loss function
7. Develop algorithm to minimize loss
8. Choose performance measure
9. "Train" to minimize loss
10. "Test" to evaluate generalization



[4] http://www.dataists.com/2010/09/a-taxonomy-of-data-science/



How does Cinematch do it?

Straightforward statistical linear models with a lot of data conditioning. But a real-world system is much more than an algorithm, and Cinematch does a lot more than just optimize for RMSE. After all, we have a website to support. In production we have to worry about system scaling and performance, and we have additional sources to data we can use to guide our recommendations. But, as mentioned in the Rules and just to be perfectly clear, for the purposes of the Prize the RMSE values we report here do not use any of this extra data.

Shipping = Feature

Add an asymmetric frequency feature $\mathbf{y}^{(3)}_{j,f_{ut}}$: **SBRAMF-UTB-UTF-MTF-ATF-MFF-AFF**

$$\widehat{r_{uit}} = \mu_i + \mu_u + \mu_{u,t} + \mu_{i,\text{bin}(t)} + \left(\mathbf{p}_i^{(1)} + \mathbf{p}_{i,\text{bin}(t)}^{(2)} + \mathbf{p}_{i,f_{ut}}^{(3)}\right)^T \left(\mathbf{q}_u^{(1)} + \mathbf{q}_{u,t}^{(2)} + \frac{1}{\sqrt{|N(u)|}} \sum_{j \in N(u)} \left(\mathbf{y}_j^{(1)} + \mathbf{y}_{j,\text{bin}(t)}^{(2)} + \mathbf{y}_{j,f_{ut}}^{(3)}\right)\right) \tag{34}$$

Model extension (+)	epoch time	#epochs	probeRMSE, $k = 50$ features
SBRMF - SVD with biases	17[s]	69	0.9054
SBRAMF - asymmetric part	50[s]	30	0.8974
+UTB - user time bias	61[s]	50	0.8919
+UTF - user time feature	62[s]	38	0.8911
+MTF - movie time feature	74[s]	37	0.8908
+ATF - asymmetric time feature	74[s]	44	0.8905
+MFF - movie frequency feature	149[s]	46	0.8900
+AFF - asymmetric frequency feature	206[s]	45	0.8886 (0.8846 with $k = 1000$)

Themes

Data jeopardy

Regardless of scale, it's difficult to find the right questions to ask of the data

Themes

Data hacking

Cleaning and normalizing data is a substantial amount of the work (and likely impacts results)

Themes

Data hacking

The ability to iterate quickly, asking and answering many questions, is crucial

Data hacking

Hacks happen: sed/awk/grep are useful, and scale

Themes

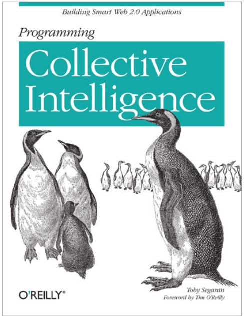
Data "science"

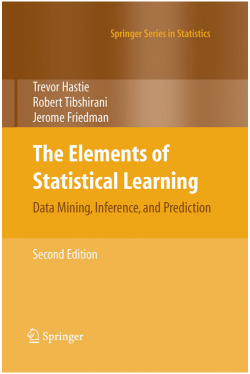
Simple methods (e.g., linear models) work surprisingly well, especially with lots of data

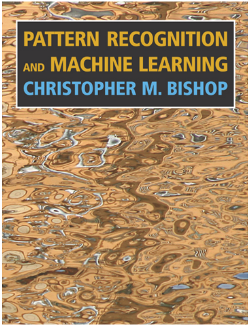
Data "science"

It's easy to cover your tracks—things are often much more complicated than they appear

References


Building Smart Web 2.0 Applications
Programming
Collective Intelligence
O'REILLY
Toby Segaran
Foreword by Tim O'Reilly


Springer Series in Statistics
Trevor Hastie
Robert Tibshirani
Jerome Friedman
The Elements of Statistical Learning
Data Mining, Inference, and Prediction
Second Edition
Springer


PATTERN RECOGNITION AND MACHINE LEARNING
CHRISTOPHER M. BISHOP