Machine Learning for Fun and Profit

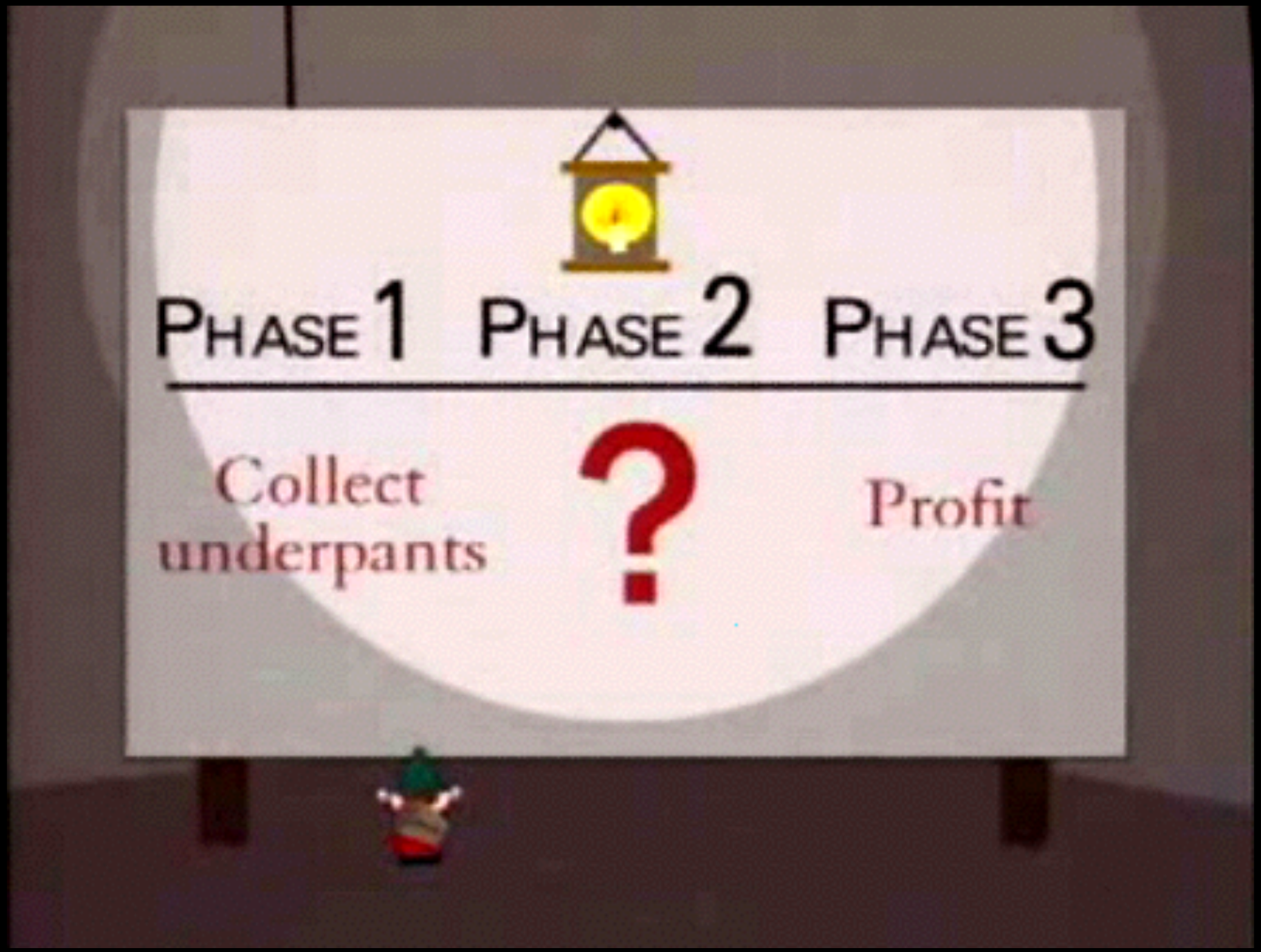
John Paul Ashenfelter

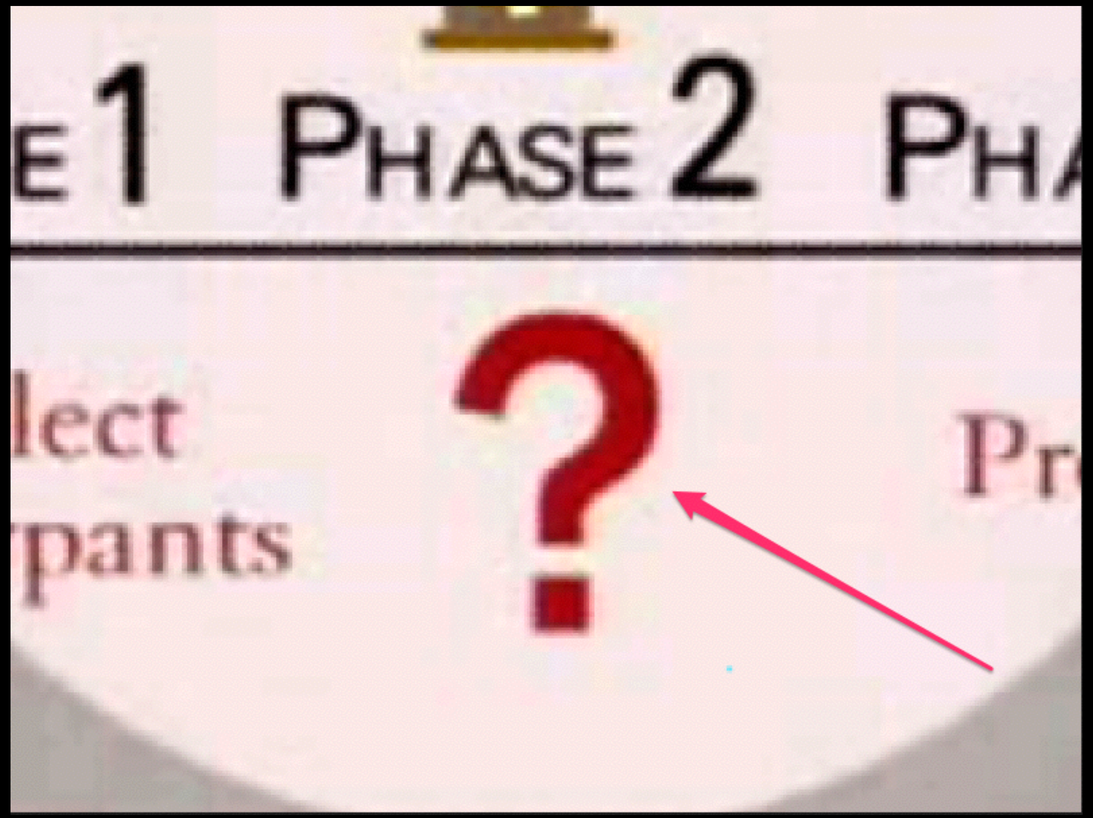
Prologue

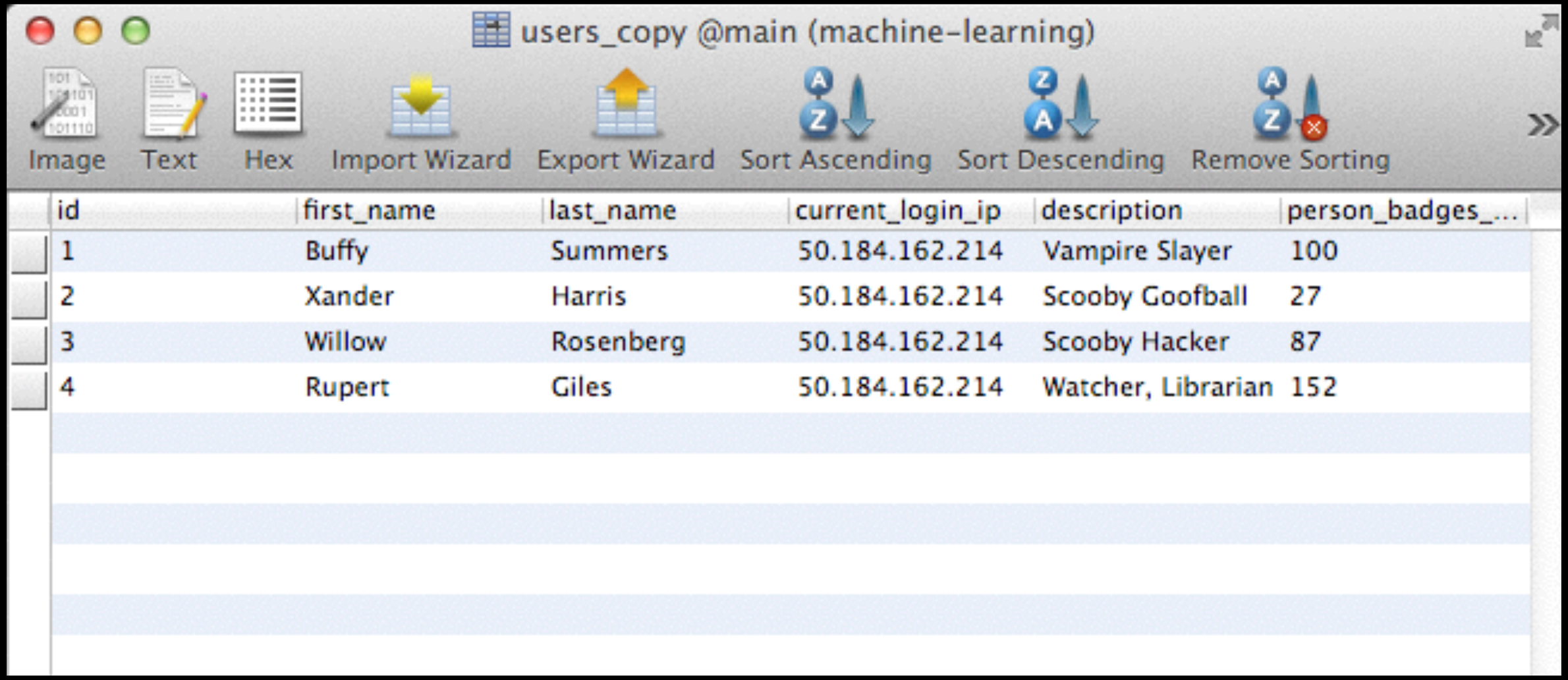
Our Goal

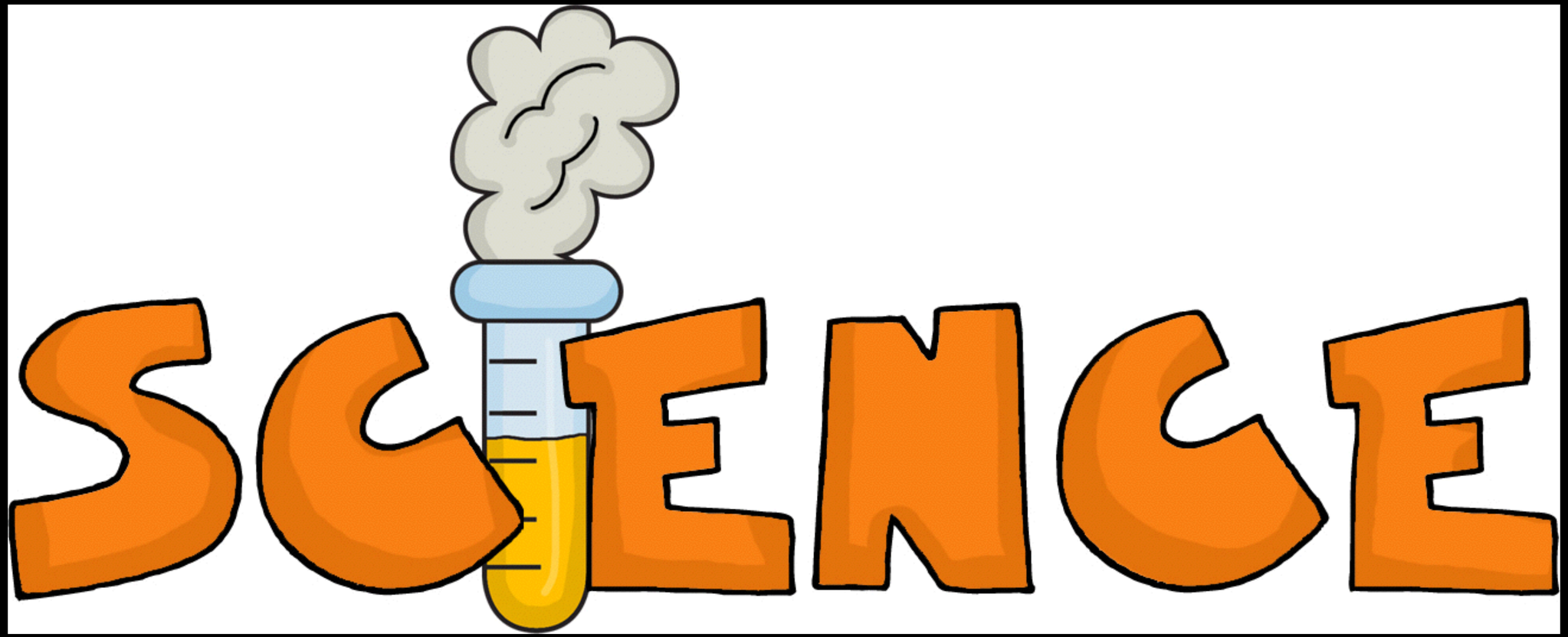
Use Ruby to answer questions about your users and your business

What’s Your Plan?


PHASE 1
PHASE 2
PHASE 3
Collect underpants
?
Profit


E 1 PHASE 2 PHA
lect
pants
?
Pr


users_copy @main (machine-learning)
Image
Text
Hex
Import Wizard
Export Wizard
Sort Ascending
Sort Descending
Remove Sorting
id
first_name
last_name
current_login_ip
description
person_badges_...
1
Buffy
Summers
50.184.162.214
Vampire Slayer
100
2
Xander
Harris
50.184.162.214
Scooby Goofball
27
3
Willow
Rosenberg
50.184.162.214
Scooby Hacker
87
4
Rupert
Giles
50.184.162.214
Watcher, Librarian
152


SCIENCE

GitHub

Search or type a command

Explore Features Enterprise Blog

Sign up Sign in

John Paul Ashenfelter

johnpaulashenfelter

Transitionpoint
Harrisonburg, VA
johnpaul@transitionpoint.com
http://www.transitionpoint.com
Joined on Aug 07, 2008

18	641	7
followers	starred	following

Contributions Repositories Public activity

Follow

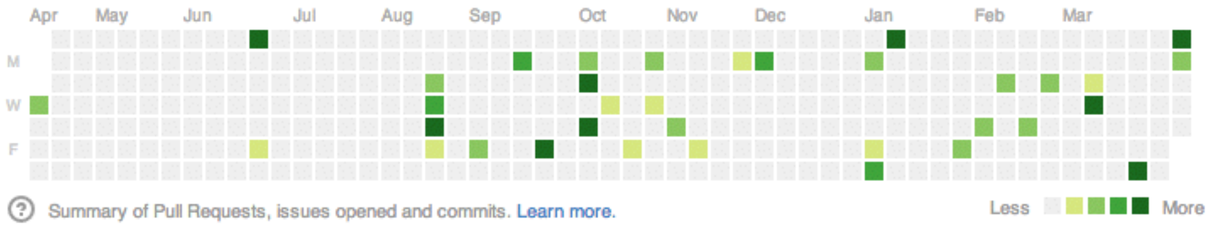
Popular repositories

dotfiles dotfiles and other config files	1 ★
firetower A command-line interface to Campfire c...	1 ★
functional-koans A set of common ideas for learning funct...	1 ★
missing-rubyconf a simple sibilant / node.js twitter scanner	1 ★
stars_align	1 ★

Repositories contributed to

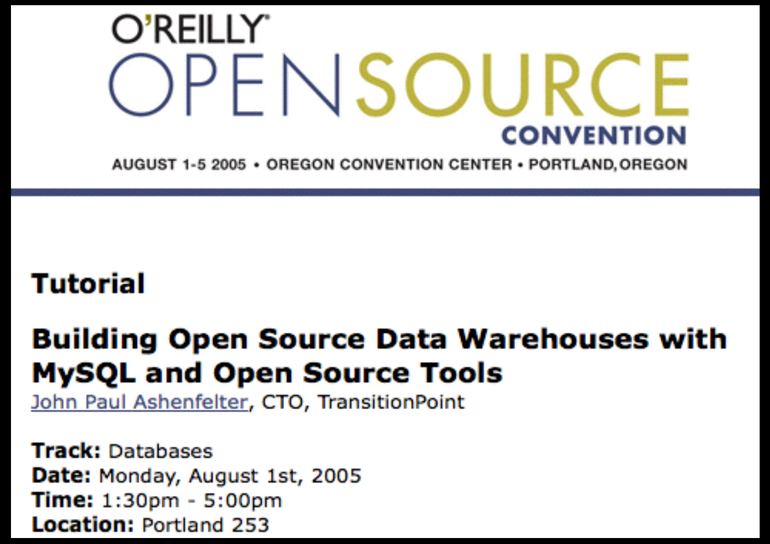
discourse/**discourse_api** Ruby API for Discourse	26 ★
discourse/**discourse** A platform for community discussion. Fr...	10,275 ★
ruby-conferen... /**ruby-confere...** A simple list of Ruby conferences	28 ★
triketora/**women-in-software-eng**	156 ★
SergioFierens/**ai4r** Artificial Intelligence for Ruby - A Ruby ...	339 ★

Public contributions

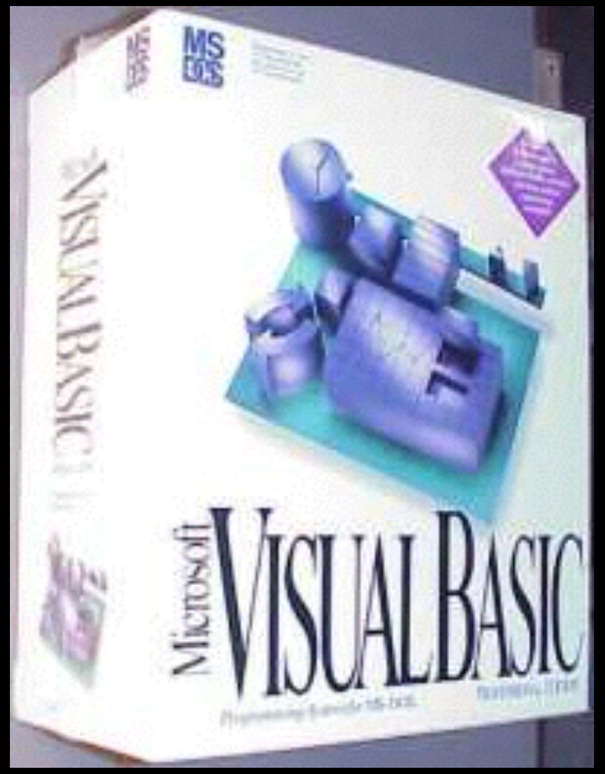


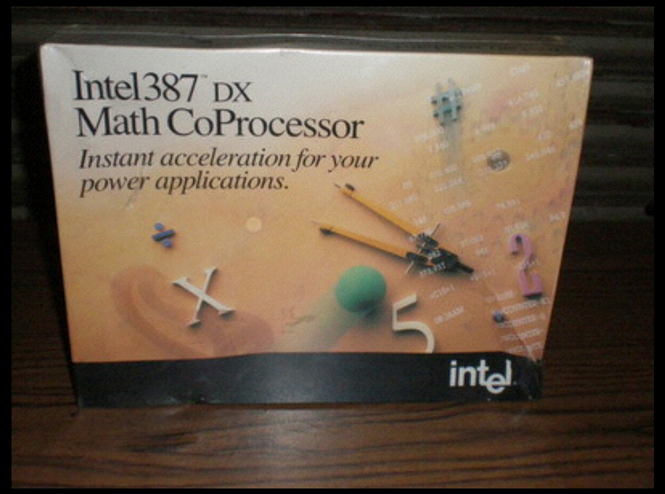
Summary of Pull Requests, issues opened and commits. Learn more.

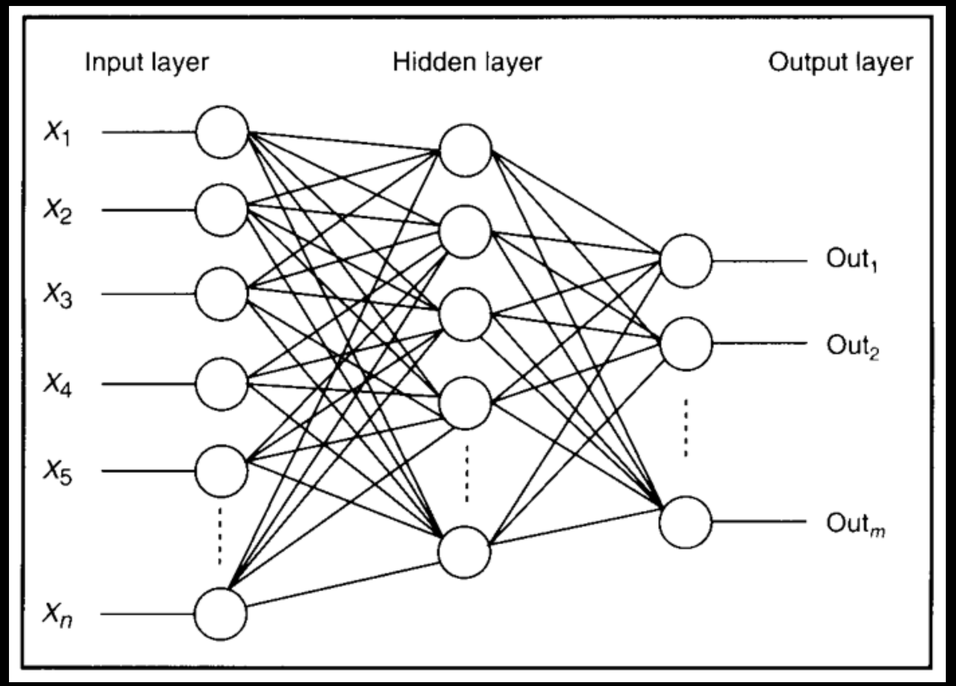
Less More



…the basics of building a data warehouse with MySQL, particularly in the **10-100 GB** range..


Microsoft
VISUAL BASIC


Intel387 DX
Math CoProcessor
Instant acceleration for your
power applications.
intel


Input layer
Hidden layer
Output layer
x_1
x_2
x_3
x_4
x_5
x_n
Out_1
Out_2
Out_m

SPECIAL TENTH ANNIVERSARY ISSUE

APR. 1989
$3.00

Inc.®

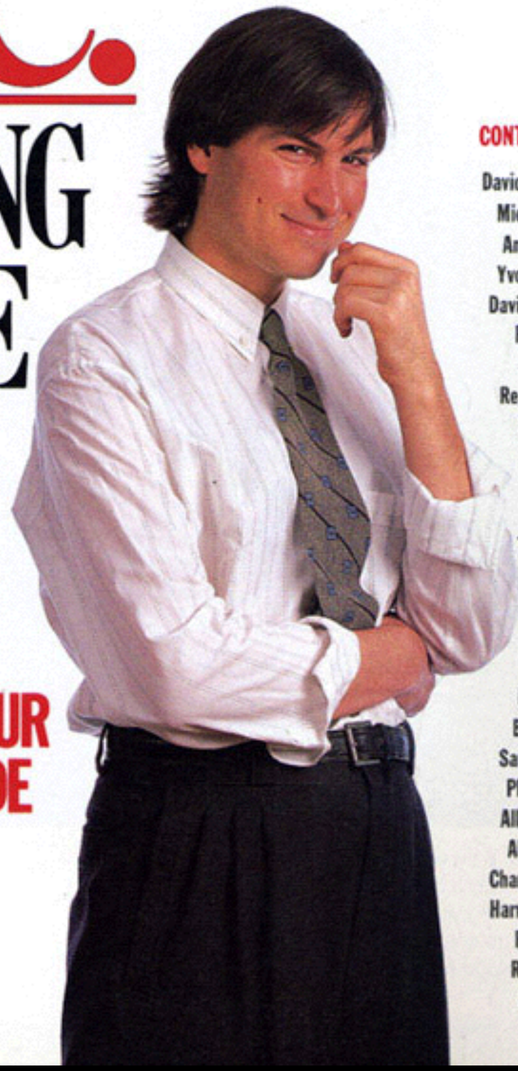
COMING OF AGE

How the Events of the Past Decade Have Changed the Way We Think About Business

PLUS

ENTREPRENEUR OF THE DECADE

An Interview with Steve Jobs



CONTRIBUTORS:

David Halberstam
Michael Kinsley
Anne Robinson
Yvon Chouinard
David Liederman
Philip Crosby
John Sculley
Regis McKenna
Paul Hawken
Dan Bricklin
Ken Iverson
Donald Burr
Tracy Kidder
Ben Cohen
Bill Gates
Tom Peters
David Birch
Ned Johnson
Bill McGowan
Sandra Kurtzig
Phillip Moffitt
Allen Neuharth
Allan Kennedy
Charlie Leighton
Harry Quadracci
Robert Reich
Robert Noyce
Fred DeLuca
Fred Smith

Format

- Start with a problem
- and some data
- Then write some code to
- get awesome results
- while learning a thing or two

Act I: Campfire Stories

Who are your users?

Google Analytics

My Account | Help | Contact Us | Sign Out

Analytics Settings | View Reports: Click to select Website Profile

- Dashboard
  - Saved Reports
- Visitors
- Traffic Sources
- Content
- Goals
- Ecommerce

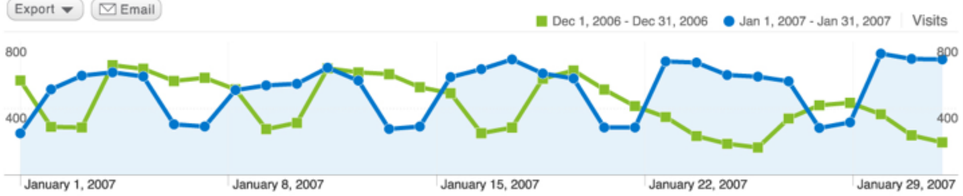
Settings
Email

Settings
- About this Report
- Conversion University
- Common Questions
- Report Finder

Dashboard

Jan 1, 2007 - Jan 31, 2007

Export | Email



Site Usage

16.107 Visits
Dec 1, 2006 - Dec 31, 2006: 13,209 (21.94%)

62.142 Pageviews
Dec 1, 2006 - Dec 31, 2006: 53,855 (15.39%)

3,86 Pages/Visit
Dec 1, 2006 - Dec 31, 2006: 4.08 (-5.37%)

00:03:22 Avg. Time on Site
Dec 1, 2006 - Dec 31, 2006: 00:03:31 (-4.27%)

36,81% Bounce Rate
Dec 1, 2006 - Dec 31, 2006: 35.98% (2.30%)

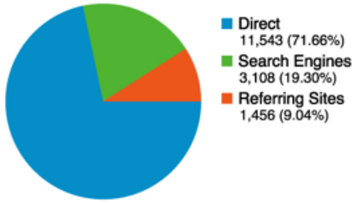
50.44% % New Visits
Dec 1, 2006 - Dec 31, 2006: 49.73% (1.42%)

Visitors Overview

Visitors
10,160

Map Overlay world

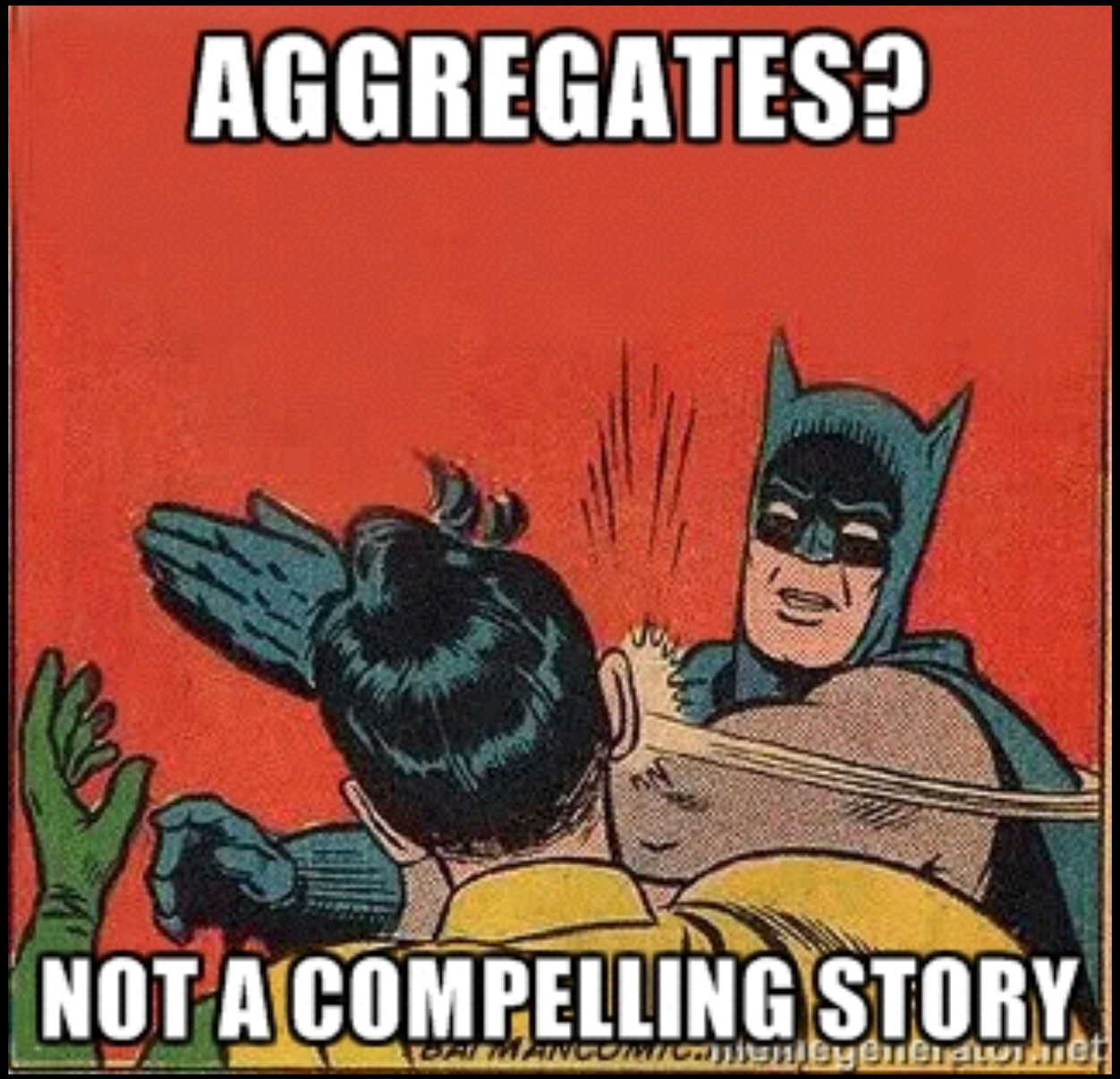
Traffic Sources Overview



- Direct
  11,543 (71.66%)
- Search Engines
  3,108 (19.30%)
- Referring Sites
  1,456 (9.04%)

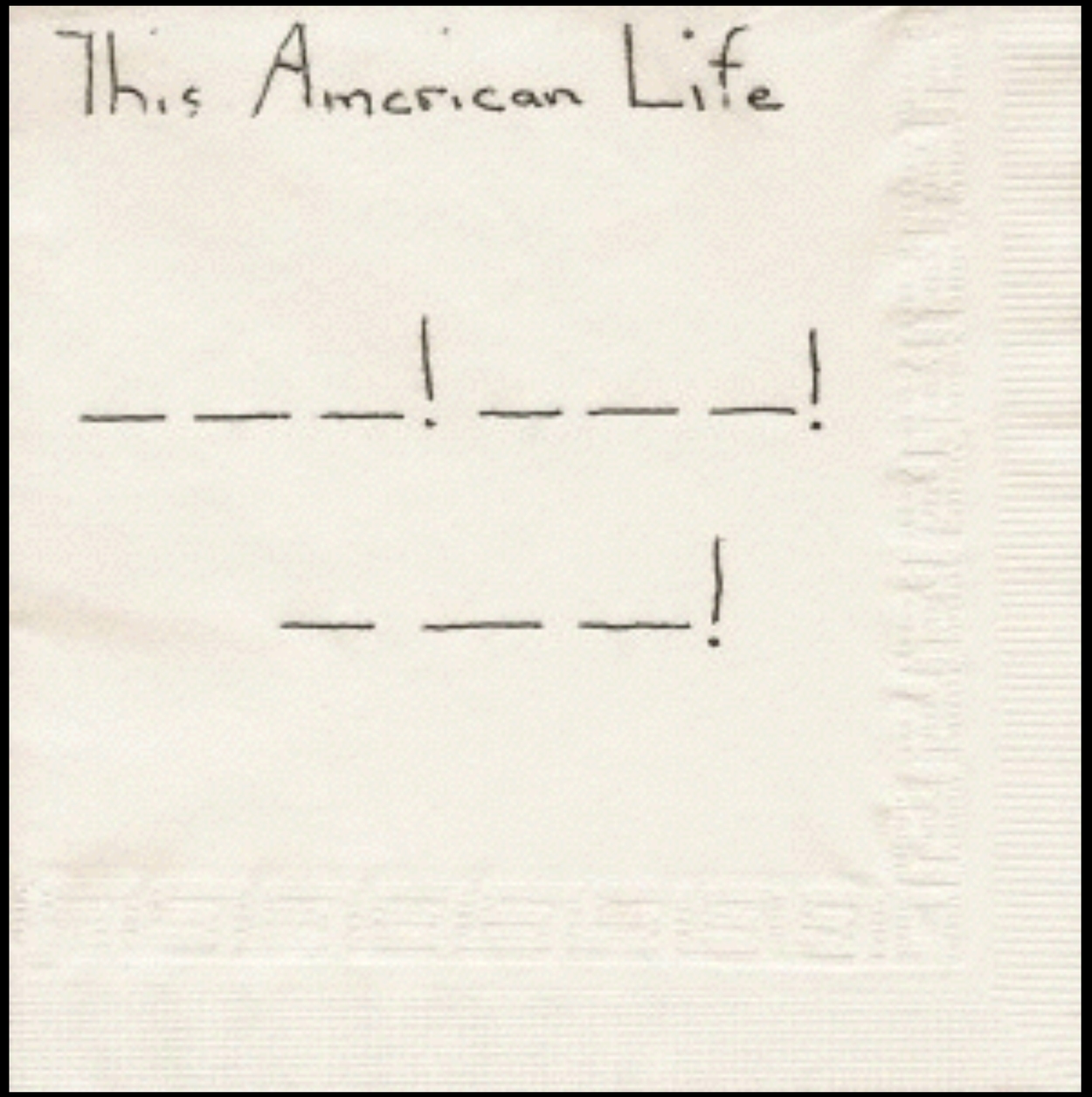
Browsers

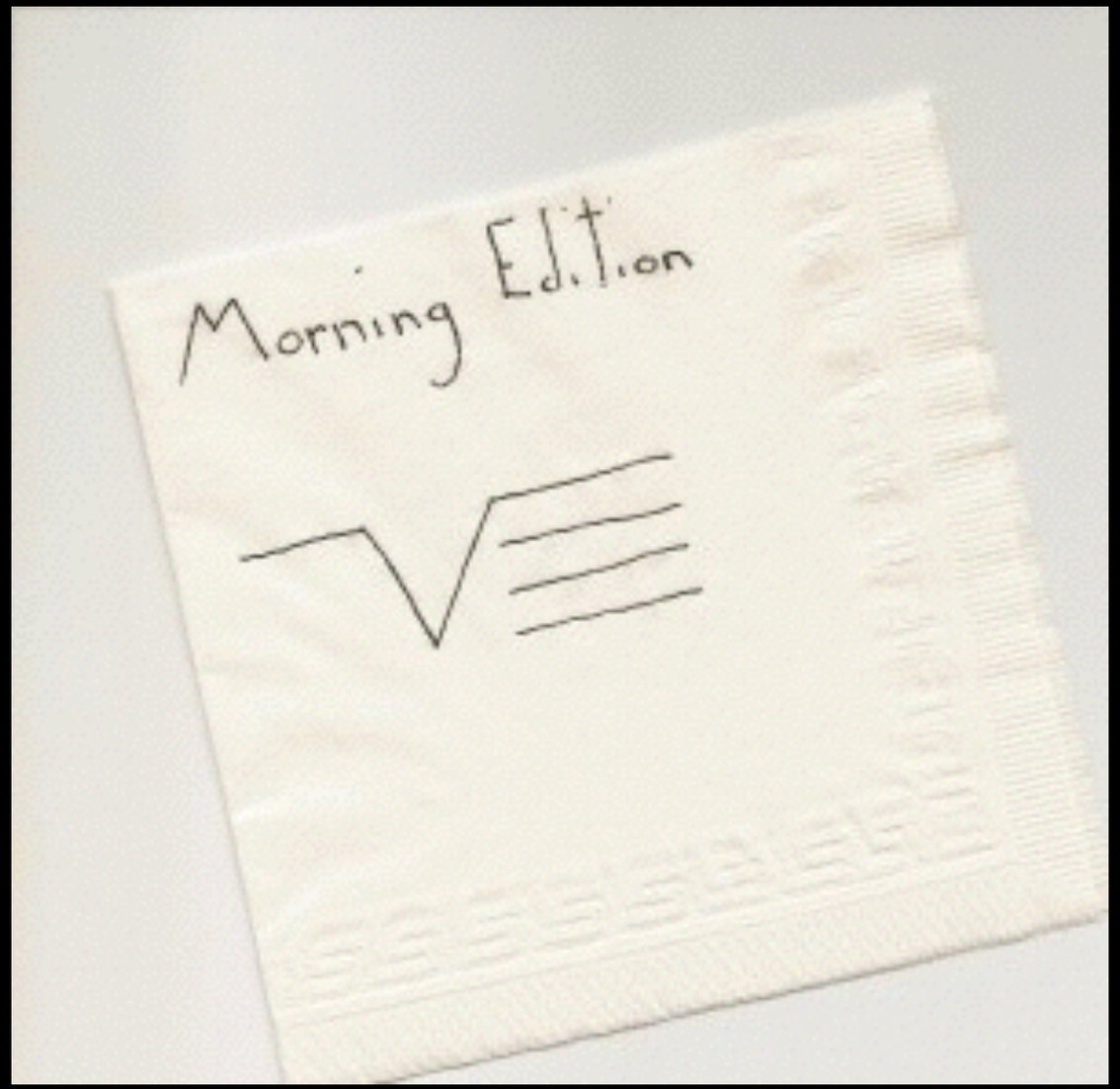
Browser	Visits	% visits
Internet Explorer	13,136	81.55%
Firefox	2,349	14.58%
Safari	393	2.44%
Netscape	116	0.72%
Opera	55	0.34%


AGGREGATES?
NOT A COMPELLING STORY

Tell good stories

- Aggregates are boring
- Events in motion are interesting
- Context makes it more interesting


This American Life


Morning Edition

7 Unbelievable Facts About Your Users!

Who are YOUR users?

- What do you know?
- How do you know it?
- Are you sure?
- What’s missing?

Who are your users?

- Descriptive data
- Slice into your data with better segmentation
- Lookup tables, spreadsheet-style calculation
- Runs fast, easy to do

Exercise 1: Gender Assignment

- User.select(‘first_name’).all
- gem ‘sexmachine’

Exercise 1: Code

```
require ‘sexmachine’

alg = SexMachine::Detector.new(case_sensitive: false)
```

Exercise 1: Code

```
d = SexMachine::Detector.new(case_sensitive: false)

puts "Bob is #{d.get_gender('Bob')}"
```

Exercise 1: Code

```
# Create detector
d = SexMachine::Detector.new(case_sensitive: false)

# Use the algorithm to assign a gender to 'Bob'
puts "Bob is #{d.get_gender('Bob')}"
```

Exercise 1: Your Turn

ex1_gender

```
check your gender.rb
assign_gender_to_users.rb
```

Exercise 2: Location Awareness

- User.select('ip_address').all
- https://github.com/fiorix/freegeoip (freegeoip.net)
  - needs Python
  - server is Go-based
  - uses Maxmind free data

Exercise 2: Code

```
GEOCODER = ‘http://127.0.0.1:8080' # local freegeoip

conn = Faraday.new(url: GEOCODER) do |faraday|
  faraday.request  :url_encoded  # form-encode POST params
  faraday.adapter  Faraday.default_adapter  # use Net::HTTP
end
```

Exercise 2: Code

```
users.each do |user|

 if user[:current_login_ip].match(Resolv::IPv4::Regex)
```

Exercise 2: Code

```
users.each do |user|

  json = conn.get("/json/#{user[:current_login_ip]}").body
  geodata = JSON.parse(json)
```

Exercise 2: Code

```
users.each do |user|

    demo.insert(user_id: user[:id],
      lat: geodata["latitude"],
      lng: geodata["longitude"], …
      location_json: json)

end
```

Exercise 2: Your Turn

ex2_geolocation
  assign_location_to_users.rb


MISSION ACCOMPLISHED
USERS DESCRIBED
memegenerator.net

Act II: Stories of Myth and Legend


Here
Be
Dragons

Exercise 3: Tribal Culture

- User.includes(:important_properties).all
- gem ‘ai4r’

```
# Define our clusters and initialize them with two users
clusters = []
k.times {clusters << Cluster.new}

users.each do |user|
  n = user[:id] % k # assign randomly to groups
  clusters[n].add(badges: user[:person_badges_count])
end
```

```
while changed do
  changed = false

 clusters.each_with_index do |cluster, i|
    cluster.calculate_centroid

    cluster.get_people.each do |person|
      clusters.each_with_index do |other_cluster, j|
        if other_cluster.calculate_gd(person) < cluster.calculate_gd(person)
          cluster.remove(person)
          other_cluster.add(person)
          changed = true
        end
      end
    end

  end
end
```

```
while changed do
  changed = false

 clusters.each_with_index do |cluster, i|
    cluster.calculate_centroid

    cluster.get_people.each do |person|
      clusters.each_with_index do |other_cluster, j|
        if other_cluster.calculate_gd(person) < cluster.calculate_gd(person)
          cluster.remove(person)
          other_cluster.add(person)
          changed = true
        end
      end
    end

  end
end
```

Exercise 3: Your Turn

ex3_clustering

- assign_users_to_segment.rb
- cluster_with_ai4r.rb

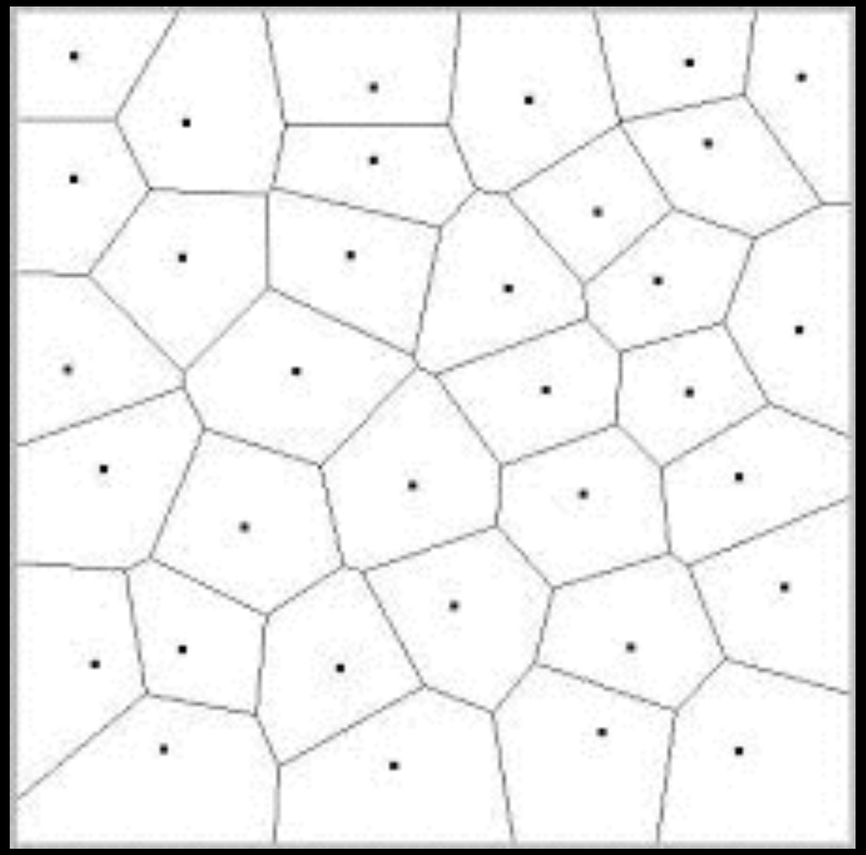
Interlude: On Mathematics

- Linear Algebra is crucial (matrices, vectors)
- Know your datastore query tools (sets)
- There are better numerical tools than Ruby

K-Means

is a method of **vector quantization**, originally from signal processing, that is popular for cluster analysis in data mining. k-means clustering aims to **partition *n* observations into *k* clusters** in which each observation **belongs to the cluster with the nearest mean**, serving as a prototype of the cluster. This results in a partitioning of the data space into **Voronoi cells**

?



Alternatives to K-Means

- **Hierarchical clusterers**
  - create one cluster per element, and then progressively merge clusters, until the required number of clusters is reached.
  - linkage is how the distance is measured
- **Divisive Hierarchical Clusterer**
  - begins with only one cluster with all data items, and divides the clusters until the desired clusters number is reached
  - DIANA (Divisive ANAlysis) is one method


Like

Exercise 4: Collaboration

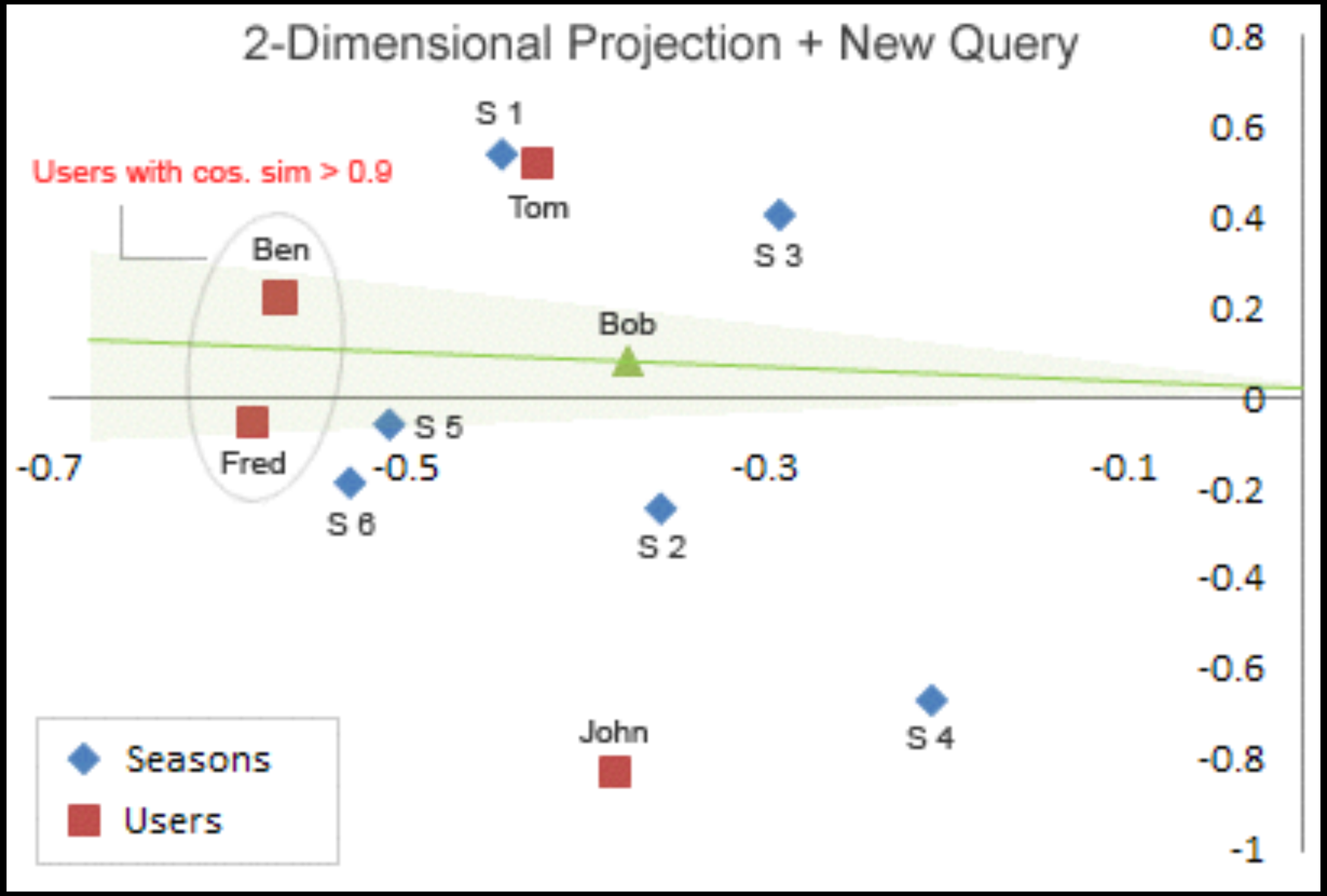
- User.includes(:important_properties).all
- gem ‘linalg’
- SVD

```
m = Linalg::DMatrix.columns(answers)
# Compute the SVD Decomposition
u, s, vt = m.singular_value_decomposition
vt = vt.transpose

u2 = Linalg::DMatrix.join_columns [u.column(0), u.column(1)]
v2 = Linalg::DMatrix.join_columns [vt.column(0), vt.column(1)]
eig2 = Linalg::DMatrix.columns [s.column(0).to_a.flatten[0,2],
s.column(1).to_a.flatten[0,2]]
```

```
m = Linalg::DMatrix.columns(answers)
# Compute the SVD Decomposition
u, s, vt = m.singular_value_decomposition
vt = vt.transpose

u2 = Linalg::DMatrix.join_columns [u.column(0), u.column(1)]
v2 = Linalg::DMatrix.join_columns [vt.column(0), vt.column(1)]
eig2 = Linalg::DMatrix.columns [s.column(0).to_a.flatten[0,2],
s.column(1).to_a.flatten[0,2]]
```

```
# add bob and embed in reduced space
bob = Linalg::DMatrix[a]
bobEmbed = bob * u2 * eig2.inverse

# Compute the cosine similarity between Bob and every user
user_sim, count = {}, 1
v2.rows.each { |x|
  user_sim[count] = (bobEmbed.transpose.dot(x.transpose)) /
(x.norm * bobEmbed.norm)
  count += 1
}
```

```
# add bob and embed in reduced space
bob = Linalg::DMatrix[a]
bobEmbed = bob * u2 * eig2.inverse

# Compute the cosine similarity between Bob and every user
user_sim, count = {}, 1
v2.rows.each { |x|
  user_sim[count] = (bobEmbed.transpose.dot(x.transpose)) /
(x.norm * bobEmbed.norm)
  count += 1
}
```


2-Dimensional Projection + New Query
Users with cos. sim > 0.9
S 1
Tom
S 3
Ben
Bob
S 5
Fred
S 6
S 2
John
S 4
-0.7
-0.5
-0.3
-0.1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
Seasons
Users

Exercise 4: Your Turn

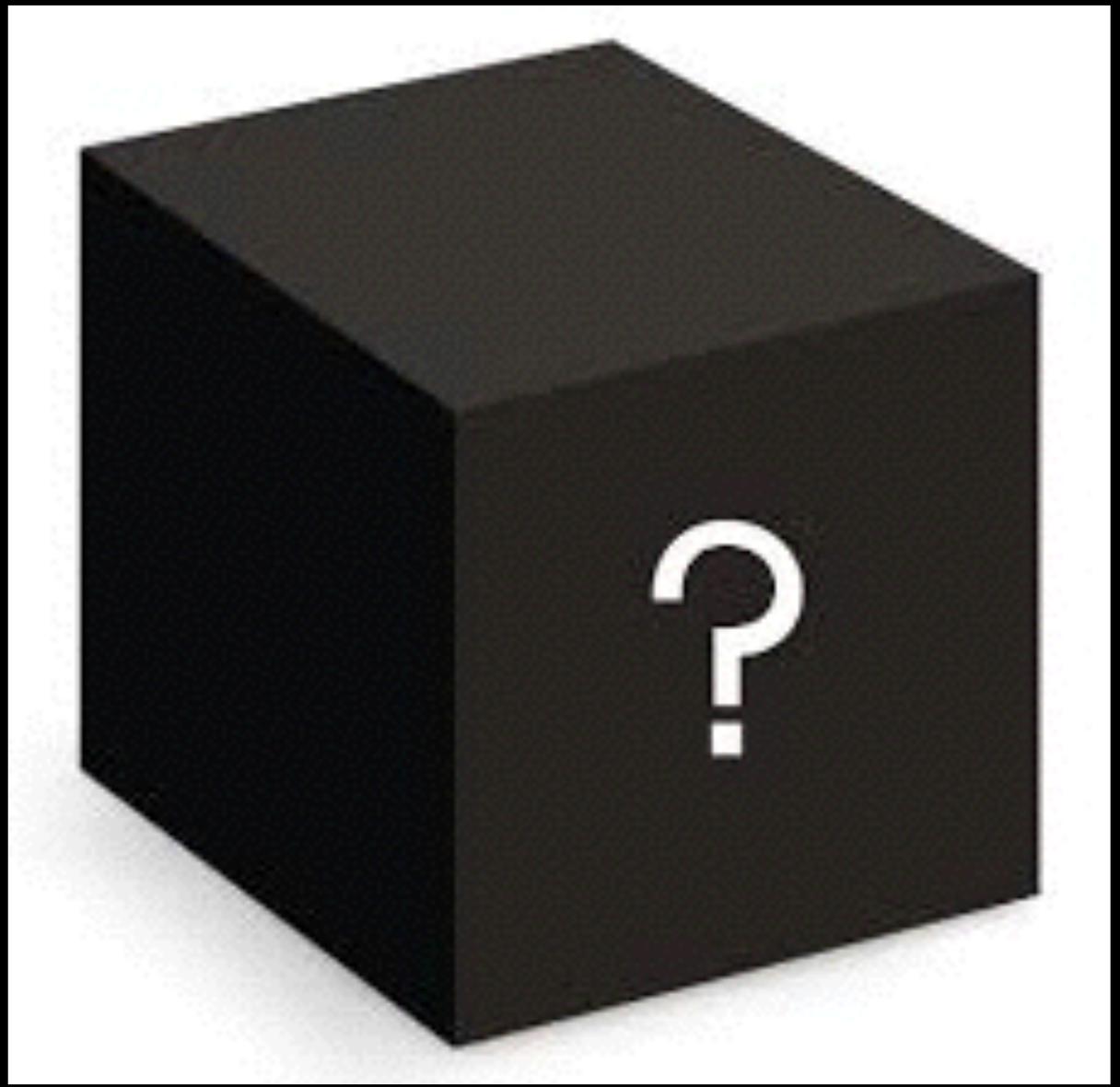
ex4_similarity
  similar_users.rb

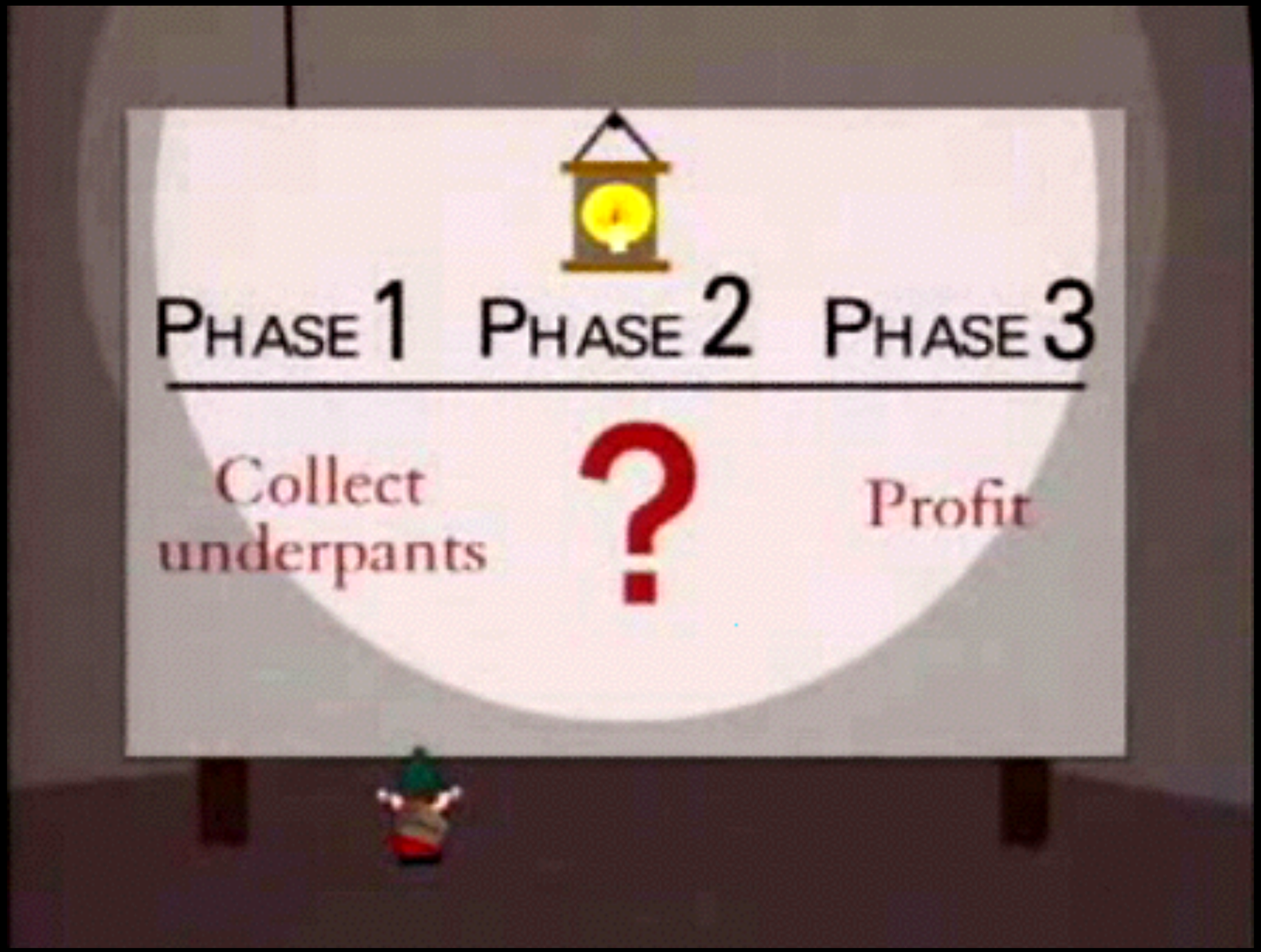
Epilogue

Our Goal, Redux

Use Ruby to answer questions about your users and your business

Leave with the tools to answer some questions today




PHASE 1
PHASE 2
PHASE 3
Collect underpants
?
Profit

Credits

railsconf

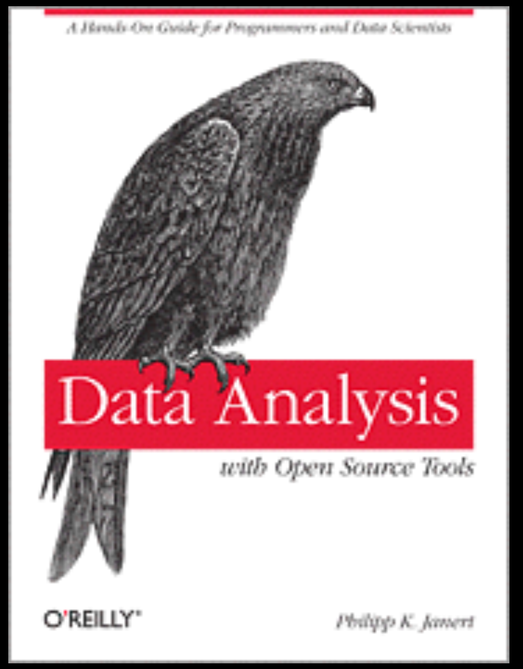
CHICAGO
HOTEL MONACO
A KIMPTON® HOTEL

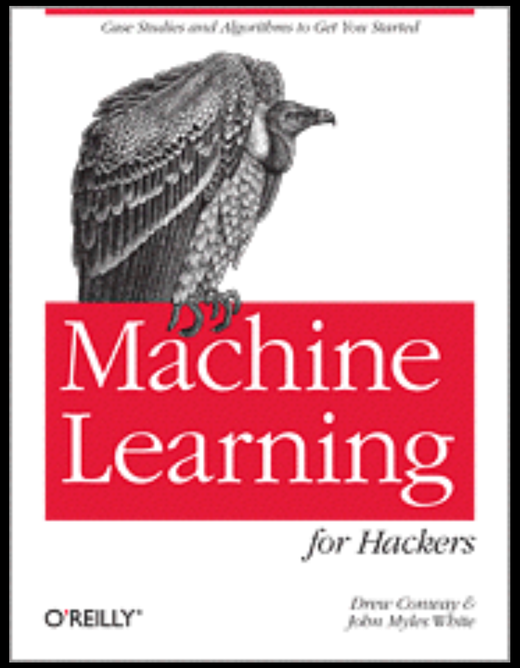
AMTRAK

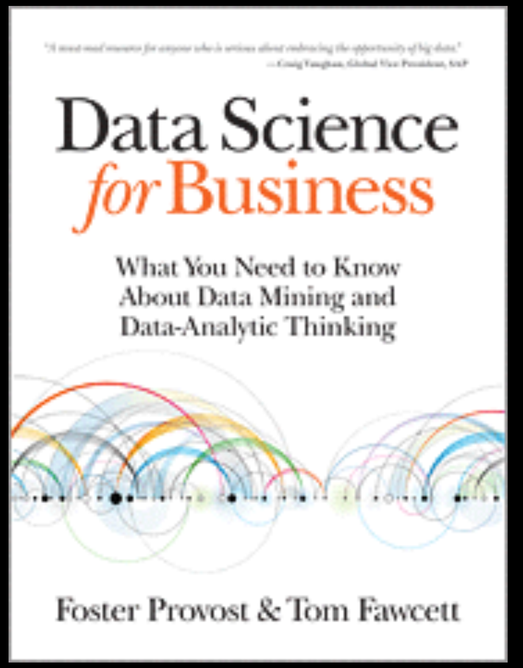
Contact

- John Paul Ashenfelter
- johnpaul@transitionpoint.com
- @johnashenfelter


O'REILLY
Doing Data Science
STRAIGHT TALK FROM THE FRONTLINE
Cathy O'Neil & Rachel Schutt


A Hands-On Guide for Programmers and Data Scientists
Data Analysis
with Open Source Tools
O'REILLY
Philipp K. Janert

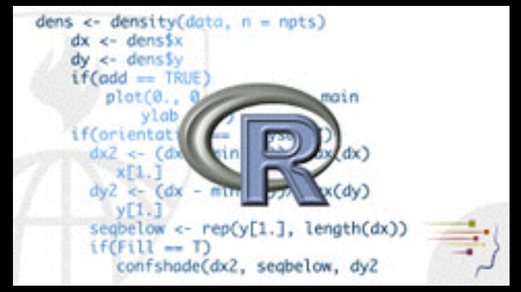

Case Studies and Algorithms to Get You Started
Machine Learning
for Hackers
O'REILLY
Drew Conway & John Myles White

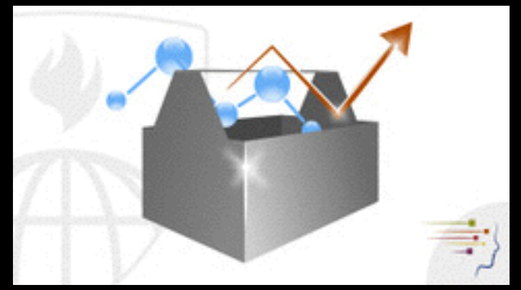

Data Science for Business
What You Need to Know About Data Mining and Data-Analytic Thinking
Foster Provost & Tom Fawcett

Stanford | ONLINE

coursera

JOHNS HOPKINS
UNIVERSITY





John Paul Ashenfelter

- johnpaul@transitionpoint.com
- @johnashenfelter

Tools of the Trade

- Python
- R
- Octave (Matlab)
- Mathematica
- Julia

Kinds of Machine Learning

- Supervised (right answers given)
  - regression — predicts continuous values
  - classification — predicts discrete (0/1) values
- Unsupervised
  - clustering
  - signal separation