What to demand from a Scientific Computing Language, Even if you don't care about computing or languages

Peter Norvig
Google

How I came to Python

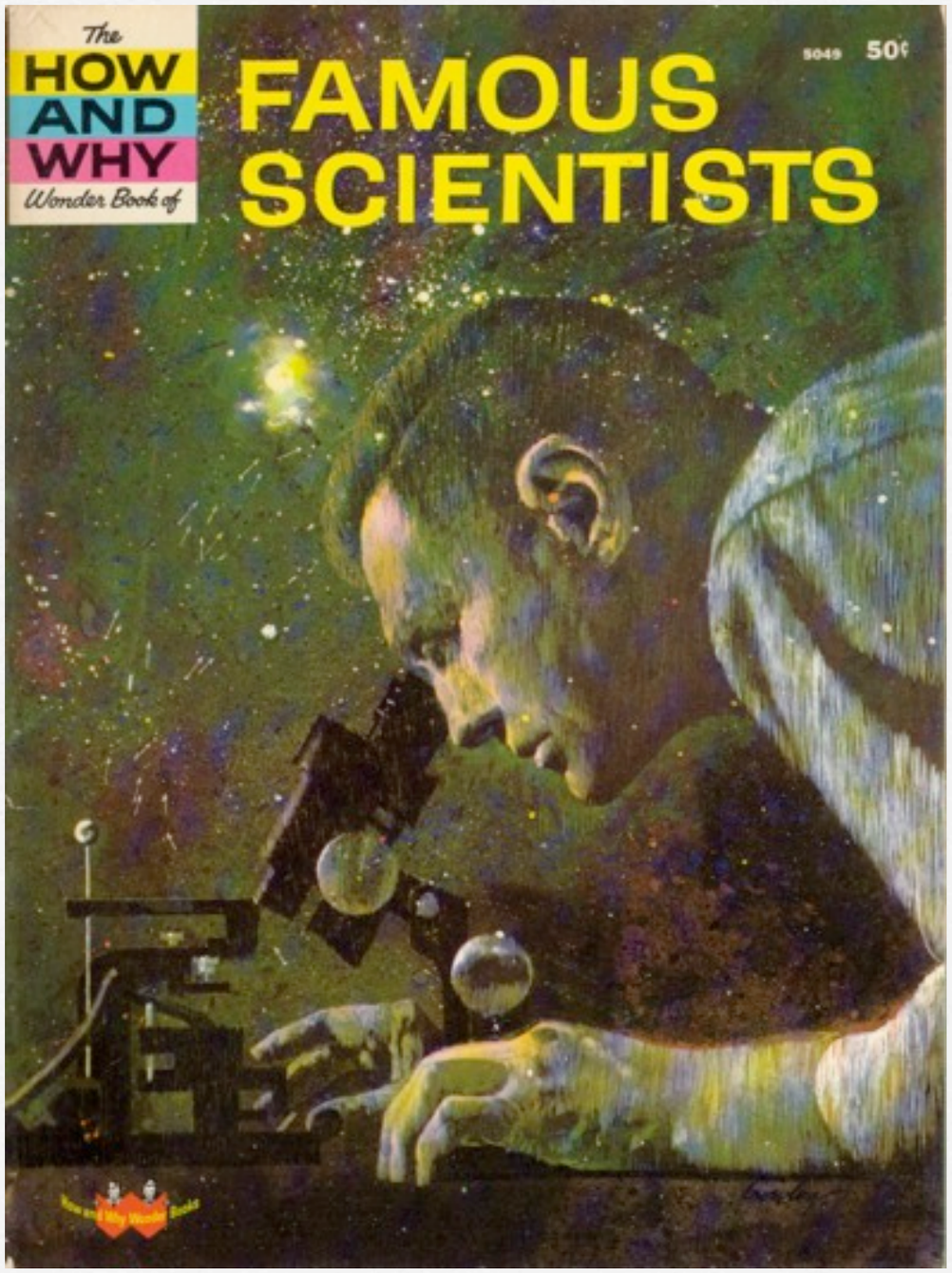
```
(defun backtracking-search (csp)
  (backtrack (make-assignment) csp))

(defun backtrack (assignment csp)
  (if (complete? assignment csp)
      assignment
    (let ((var (select-unassigned-variable csp))
          (inferences nil))
      (loop for value in (order-domain-values var assignment csp) do
            (when (consistent? var value csp)
              (modify-assignment var val assignment)
              (setq inferences (inference csp var value))
              (when (neq inferences 'failure)
                (loop for (x v) in inferences do (modify-assignment x v assignment))
                (let ((result (backtrack assignment csp)))
                  (when (neq result 'failure)
                    (return result)))))
            (undo-assignment var val assignment)
            (loop for (x v) in inferences do (undo-assignment x v assignment)))
      'failure)))
```

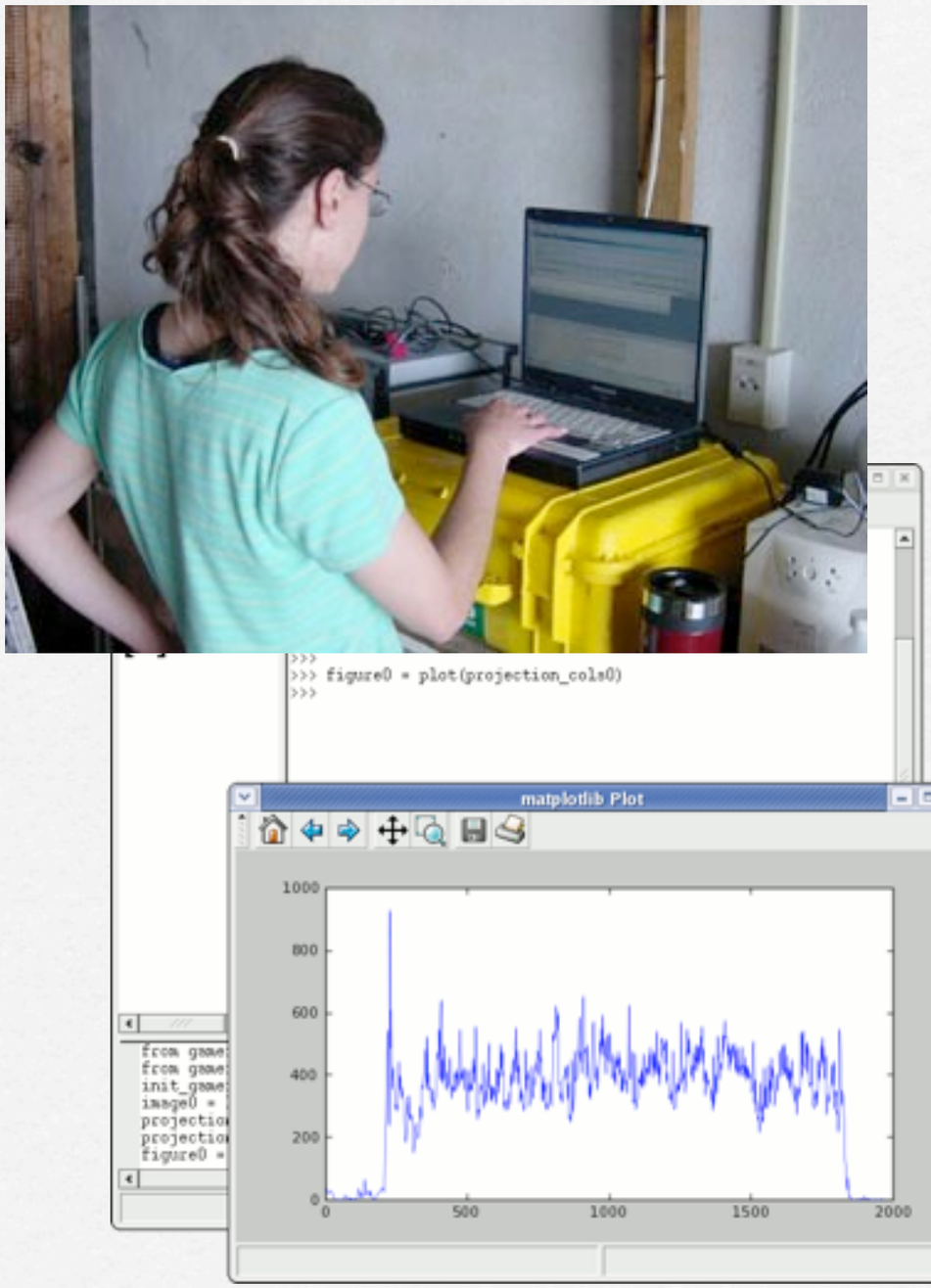
Basic
PL/1
Pascal*
Lisp*
Perl*
Java*
Python

How I came to Python

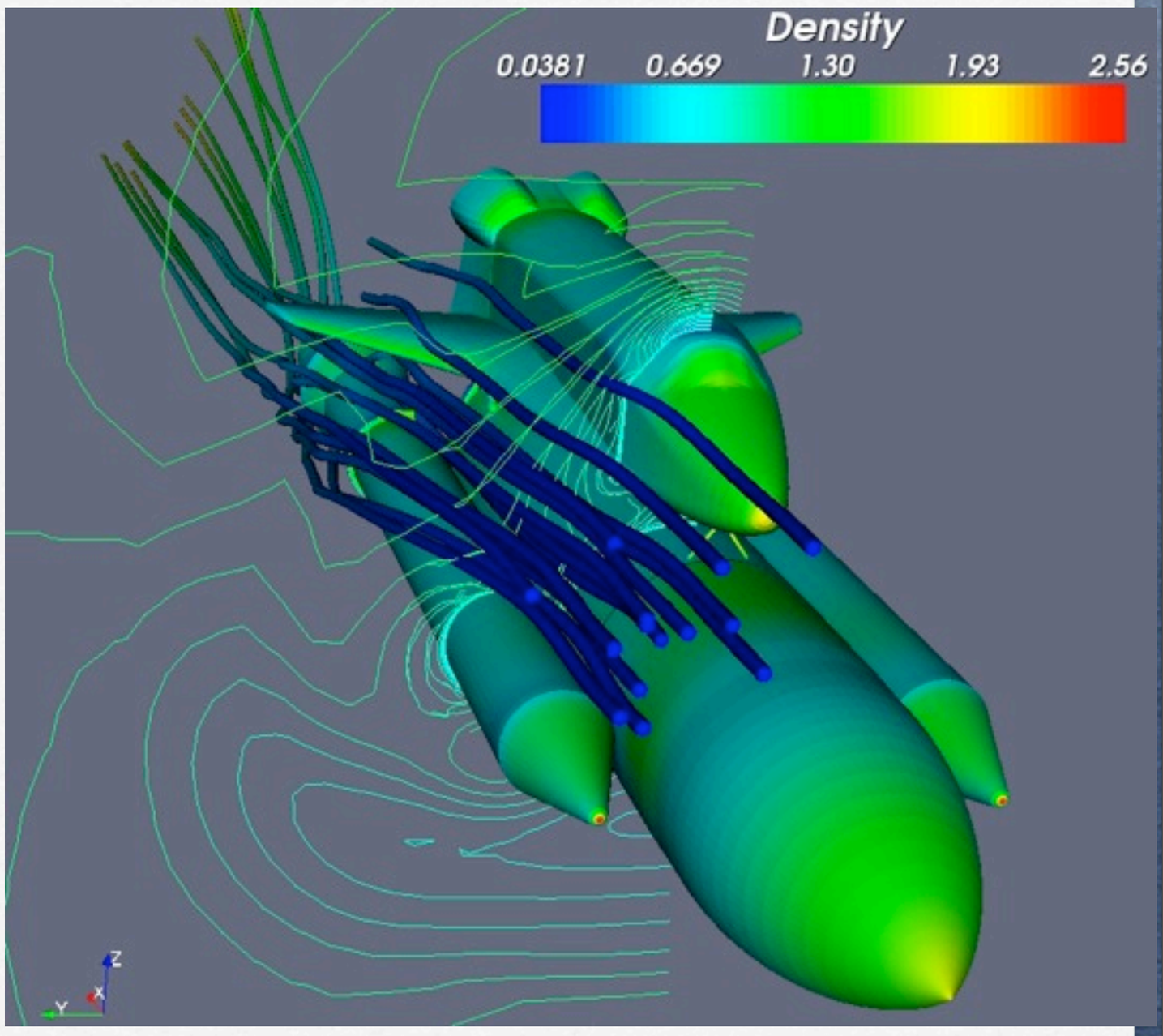
```
function BACKTRACKING-SEARCH(csp) returns a solution, or failure
    return BACKTRACK({ }, csp)

function BACKTRACK(assignment, csp) returns a solution, or failure
    if assignment is complete then return assignment
    var ← SELECT-UNASSIGNED-VARIABLE(csp)
    for each value in ORDER-DOMAIN-VALUES(var, assignment, csp) do
        if value is consistent with assignment then
            add {var = value} to assignment
            inferences ← INFERENCE(csp, var, value)
            if inferences ≠ failure then
                add inferences to assignment
                result ← BACKTRACK(assignment, csp)
                if result ≠ failure then
                    return result
        remove {var = value} and inferences from assignment
    return failure
```

What do scientists do?





What do Scientists do?



Scientific computing?

- FLOPS: Fortran, C
- Statistics: MATLAB, R, NumPy, ...
- ML: Weka, BNT, Orange, PyML
- Big Data: SPSS, Stata, MapReduce/Hadoop
- Symbolic: Mathematica, Macsyma, Maple
- Everything: Python, Ruby, Scala, Haskell

Faster, Better, Cheaper

An engineering approach

nature physics

Journal content

- Journal home
- Advance online publication
  - About AOP
- Current issue
- Archive
- Supplements
- Focuses
- Press releases

Journal information

- Guide to authors
- Online submission
- For referees
- Pricing
- Contact the journal

Article abstract

Nature Physics
Published online: 26 July 2009 | doi:10.1038/nphys1341

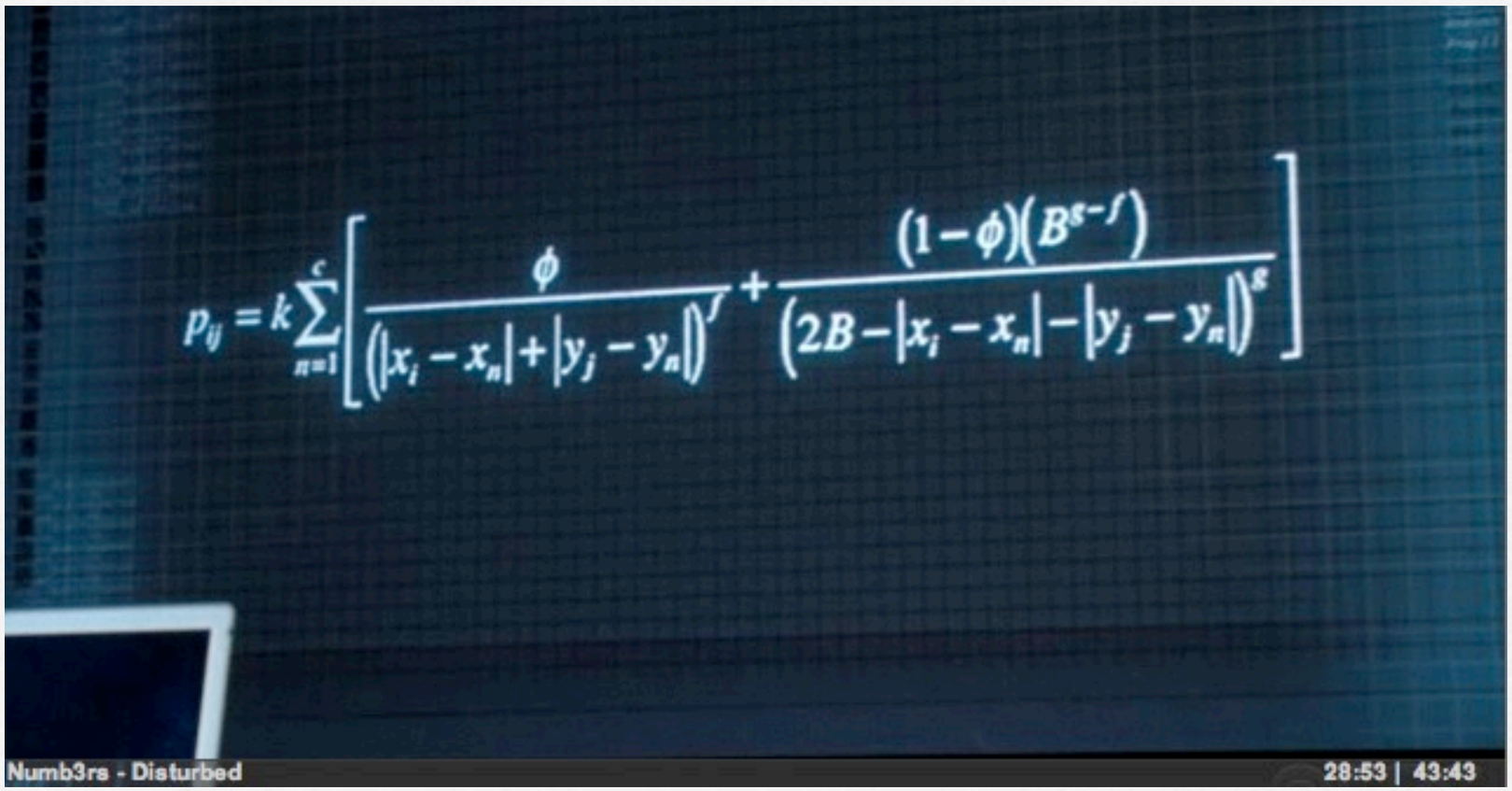
Turning solid aluminium transparent by intense soft X-ray photoionization

Bob Nagler *et al.*[21]

Saturable absorption is a phenomenon readily seen in the optical and infrared wavelengths. It has never been observed in core-electron transitions owing to the short lifetime of the excited states involved and the high intensities of the soft X-rays needed. We report saturable absorption of an L-shell transition in aluminium using record intensities over 10^{16} W cm^{-2} at a photon energy of 92 eV. From a consideration of the relevant timescales, we infer that immediately after the X-rays have passed, the sample is in an exotic state where all of the aluminium atoms have an L-shell hole, and the valence band has approximately a 9 eV temperature, whereas the atoms are still on their crystallographic positions. Subsequently, Auger decay heats the material to the warm dense matter regime, at around 25 eV temperatures. The method is an ideal candidate to study homogeneous warm dense matter, highly relevant to planetary science, astrophysics and inertial confinement fusion.


NUMB3RS

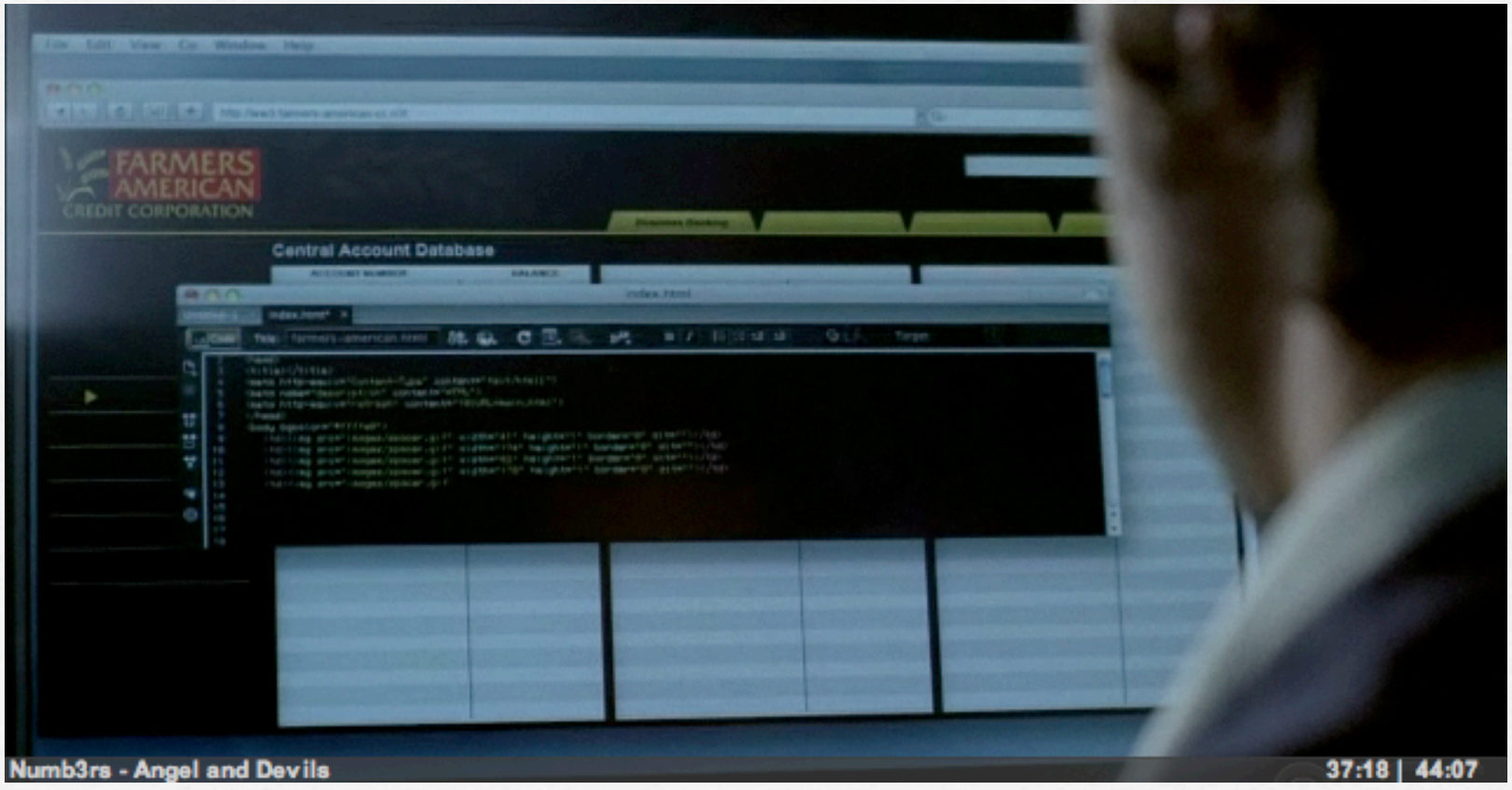
$$p_{ij} = k\sum_{n=1}^{c}\left[\frac{\phi}{\left(|x_i - x_n| + |y_j - y_n|\right)^f} + \frac{(1-\phi)\left(B^{g-f}\right)}{\left(2B - |x_i - x_n| - |y_j - y_n|\right)^g}\right]$$

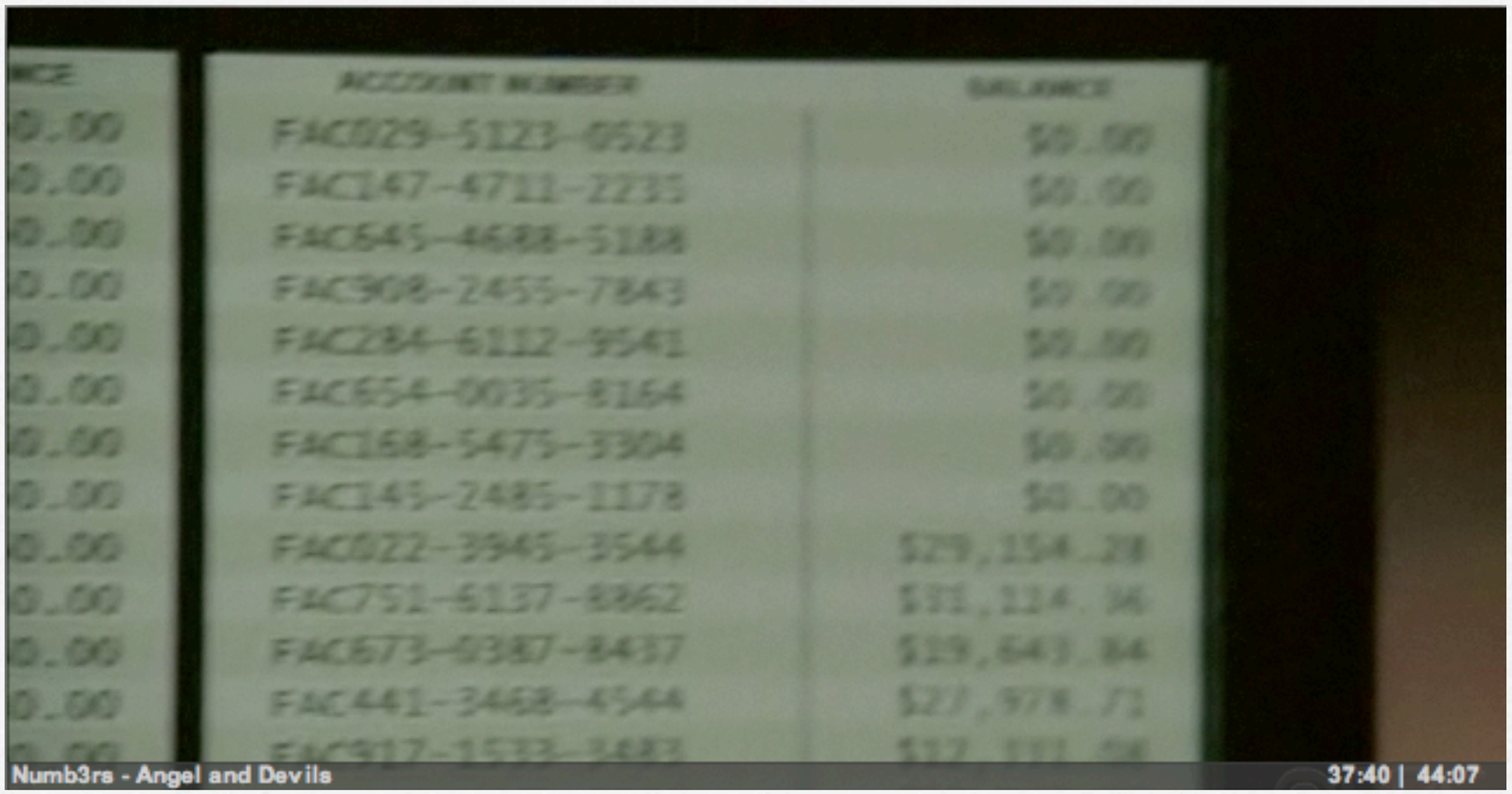



Split
S
Merge
Voronoi Cell Construction
Theorem 1 For any node
P ∈ N,
Property 1 (Voronoi Cell
Monotonicity)
For any sets of nodes

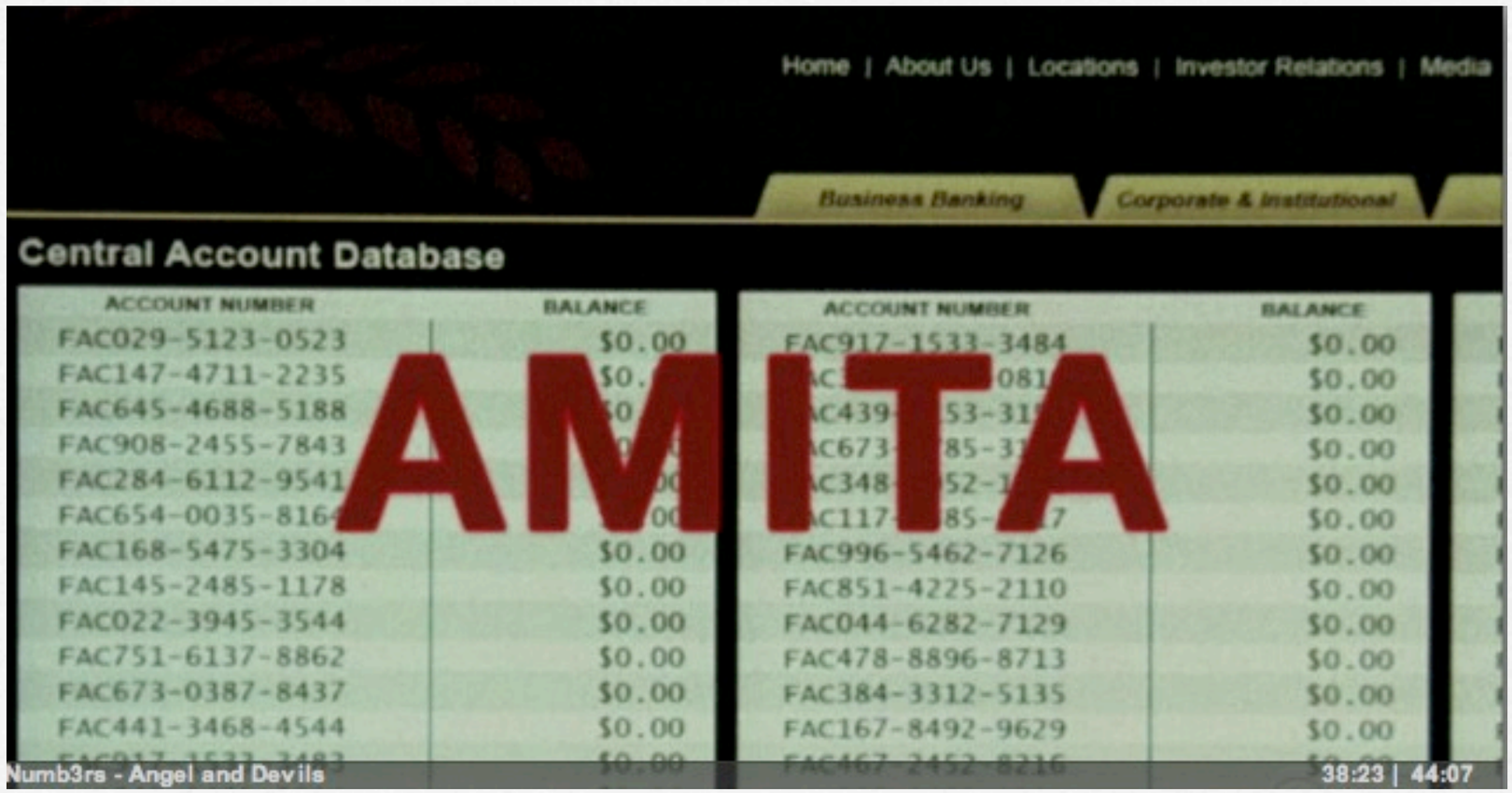
```
>
>    switchView = [[UISwitch alloc] initwithF
>    [switchView setTag:997];
>    [cell addSubview:switchView];
>    [switchView release];
>    }
>
>    // recover switchView from the cell
>    switchView = [cell viewWithTag:997];
>    [switchView addTarget:self action:@selec
>
>    // Set up the cell
>    NSArray *crayon + [[[section obejectAtIn
>    [cell setText:[objecgtAtIndex:0]];
>
>    // extract the component
>    float xx = - [acceleration x];
```

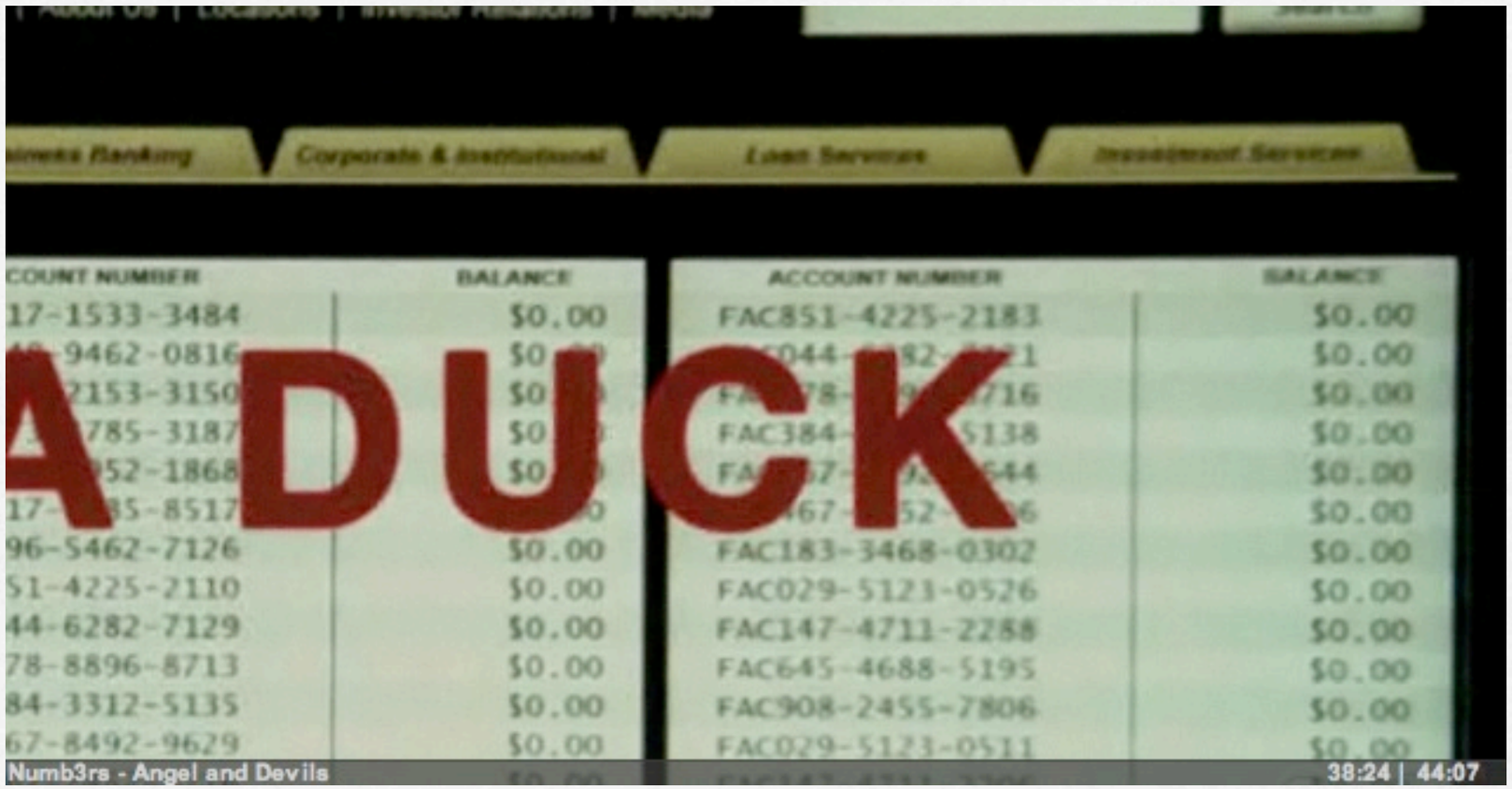
Numb3rs - Angel and Devils 26:50 | 44:07

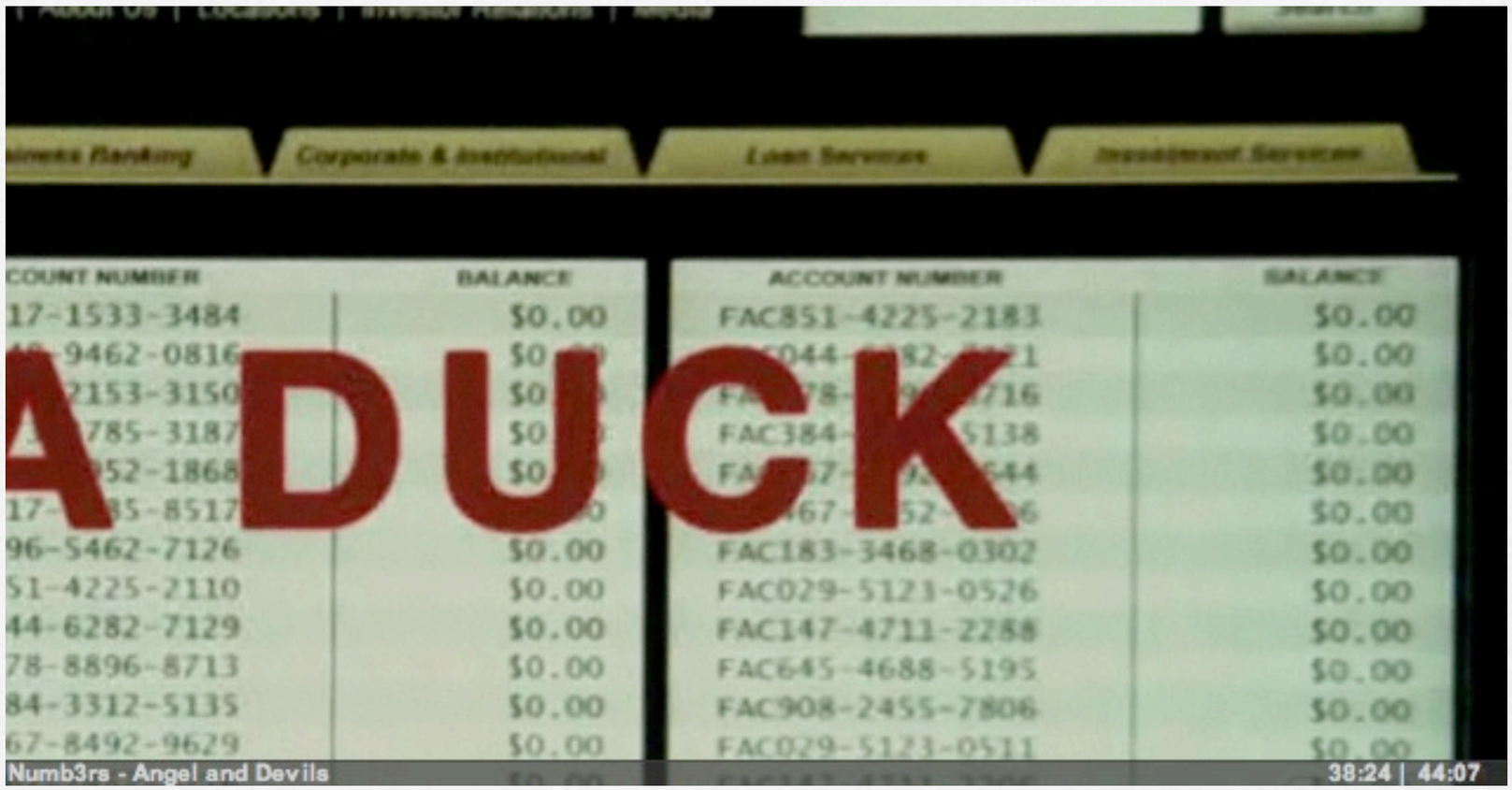

FARMERS AMERICAN
CREDIT CORPORATION
Central Account Database
Numb3rs - Angel and Devils
37:18 | 44:07


ACCOUNT NUMBER
BALANCE
FAC029-5123-0523
$0.00
FAC147-4711-2235
$0.00
FAC545-4688-5188
$0.00
FAC308-2455-7843
$0.00
FAC284-6112-9541
$0.00
FAC654-0035-8164
$0.00
FAC168-5475-3304
$0.00
FAC145-2485-1178
$0.00
FAC022-3945-3544
$29,154.28
FAC751-6137-8862
$31,114.36
FAC673-0387-8437
$19,643.84
FAC441-3458-4544
$27,978.71
Numb3rs - Angel and Devils
37:40 | 44:07


Numb3rs - Angel and Devils
38:22 | 44:07


Home | About Us | Locations | Investor Relations | Media
Business Banking
Corporate & Institutional
Central Account Database
ACCOUNT NUMBER
BALANCE
FAC029-5123-0523
$0.00
FAC147-4711-2235
FAC645-4688-5188
FAC908-2455-7843
FAC284-6112-9541
FAC654-0035-8164
FAC168-5475-3304
$0.00
FAC145-2485-1178
$0.00
FAC022-3945-3544
$0.00
FAC751-6137-8862
$0.00
FAC673-0387-8437
$0.00
FAC441-3468-4544
$0.00
ACCOUNT NUMBER
BALANCE
FAC917-1533-3484
$0.00
$0.00
$0.00
$0.00
$0.00
FAC996-5462-7126
$0.00
FAC851-4225-2110
$0.00
FAC044-6282-7129
$0.00
FAC478-8896-8713
$0.00
FAC384-3312-5135
$0.00
FAC167-8492-9629
$0.00
AMITA
Numb3rs - Angel and Devils
38:23 | 44:07


Business Banking
Corporate & Institutional
Loan Services
Investment Services
ACCOUNT NUMBER
BALANCE
17-1533-3484
$0.00
9462-0816
2153-3150
785-3187
52-1868
35-8517
96-5462-7126
$0.00
51-4225-2110
$0.00
44-6282-7129
$0.00
78-8896-8713
$0.00
84-3312-5135
$0.00
67-8492-9629
$0.00
ACCOUNT NUMBER
BALANCE
FAC851-4225-2183
$0.00
$0.00
$0.00
FAC384-
5138
$0.00
544
$0.00
$0.00
FAC183-3468-0302
$0.00
FAC029-5123-0526
$0.00
FAC147-4711-2288
$0.00
FAC645-4688-5195
$0.00
FAC908-2455-7806
$0.00
FAC029-5123-0511
$0.00
A DUCK
Numb3rs - Angel and Devils
38:24 | 44:07



A DUCK

```
<div style="text-align:center;
font-size=1in; text-color:FF0000">
AMITA DUCK
</div>
```

What do real scientists do?

VIEWPOINT

Machine Learning for Science: State of the Art and Future Prospects

Eric Mjolsness* and Dennis DeCoste

Recent advances in machine learning methods, along with successful applications across a wide variety of fields such as planetary science and bioinformatics, promise powerful new tools for practicing scientists. This viewpoint highlights some useful characteristics of modern machine learning methods and their relevance to scientific applications. We conclude with some speculations on near-term progress and promising directions.

Machine learning (ML) (*1*) is the study of computer algorithms capable of learning to improve their performance of a task on the basis of their own previous experience. The field is closely related to pattern recognition and statistical inference. As an engineering field, ML has become steadily more mathematical and more successful in applications over the past 20 years. Learning approaches such as data clustering, neural network classifiers, and nonlinear regression have found surprisingly wide application in the practice of engineering, business, and science. A generalized version of the standard Hidden Markov Models of ML practice have been used for ab initio prediction of gene structures in genomic DNA (*2*). The predictions correlate surprisingly well with subsequent gene expression analysis (*3*). Postgenomic biology prominently features large-scale gene expression data analyzed by clustering methods (*4*), a standard topic in unsupervised learning. Many other examples can be given of learning and pattern recognition applications in science. Where will this trend lead? We believe it will lead to appropriate, partial automation of every element of scientific method, from hypothesis generation to model construction to decisive experimentation. Thus, ML has the potential to amplify every aspect of a working scientist's progress to understanding. It will also, for better or worse, endow intelligent computer systems with some of the general analytic power of scientific thinking.

Machine Learning Systems Group, Jet Propulsion Laboratory/California Institute of Technology, Pasadena, CA, 91109, USA.

*To whom correspondence should be addressed. E-mail: mjolsness@jpl.nasa.gov

Machine Learning at Every Stage of the Scientific Process

Each scientific field has its own version of the scientific process. But the cycle of observing, creating hypotheses, testing by decisive experiment or observation, and iteratively building up comprehensive testable models or theories is shared across disciplines. For each stage of this abstracted scientific process, there are relevant developments in ML, statistical inference, and pattern recognition that will lead to semiautomatic support tools of unknown but potentially broad applicability.

Increasingly, the early elements of scientific method—observation and hypothesis generation—face high data volumes, high data acquisition rates, or requirements for objective analysis that cannot be handled by human perception alone. This has been the situation in experimental particle physics for decades. There automatic pattern recognition for significant events is well developed, including Hough transforms, which are foundational in pattern recognition. A recent example is event analysis for Cherenkov detectors (*8*) used in neutrino oscillation experiments. Microscope imagery in cell biology, pathology, petrology, and other fields has led to image-processing specialties. So has remote sensing from Earth-observing satellites, such as the newly operational Terra spacecraft with its ASTER (a multispectral thermal radiometer), MISR (multiangle imaging spectral radiometer), MODIS (imaging

What do scientists do?

1. Observe and explore interesting phenomena
2. Generate hypotheses
3. Formulate models to explain phenomena
4. Test predictions made by the theory
5. Modify theory and repeat

What do scientists do?

0. Write grant applications

1. Observe and explore interesting phenomena

2. Generate hypotheses

3. Formulate models to explain phenomena

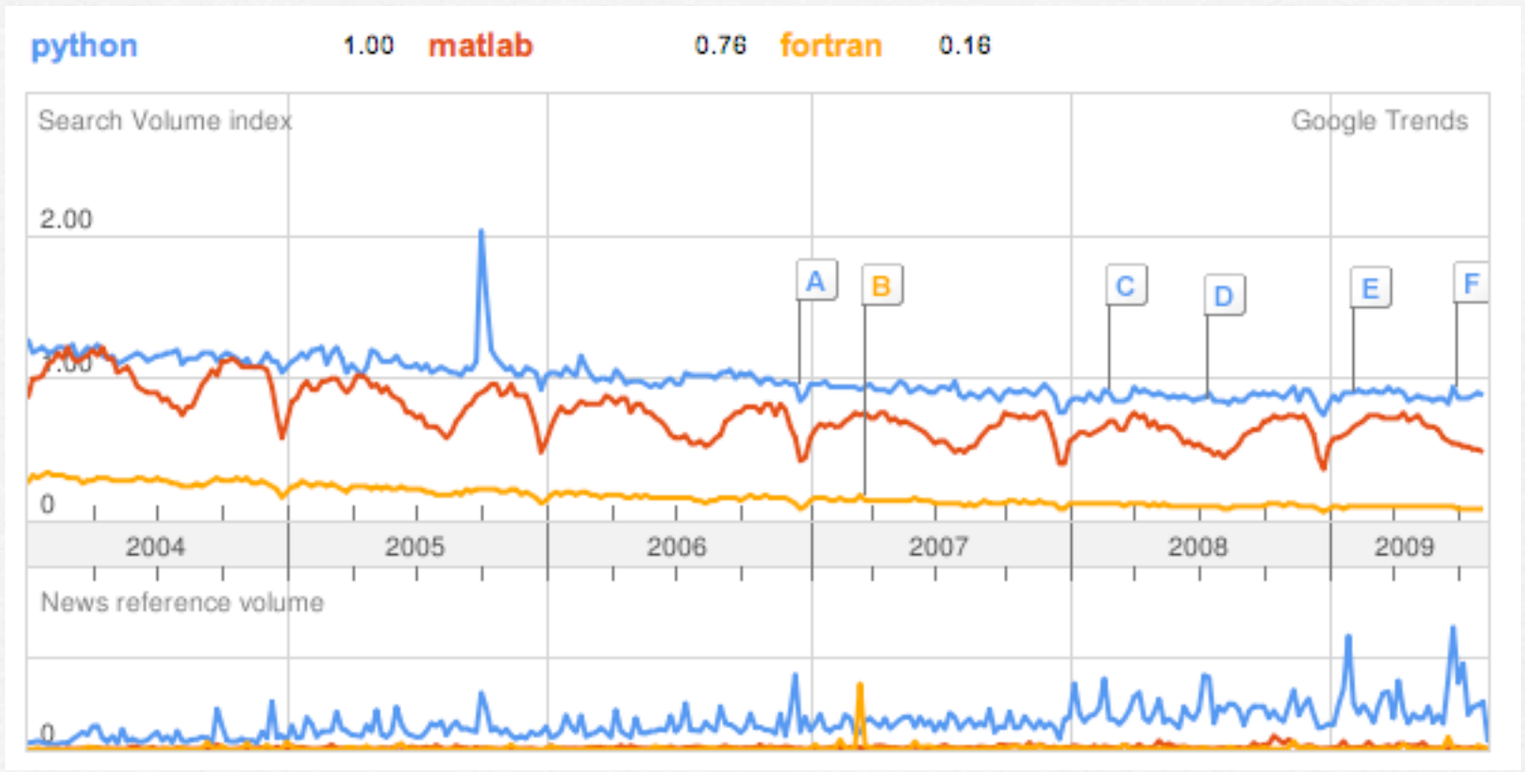
4. Test predictions made by the theory

5. Modify theory and repeat

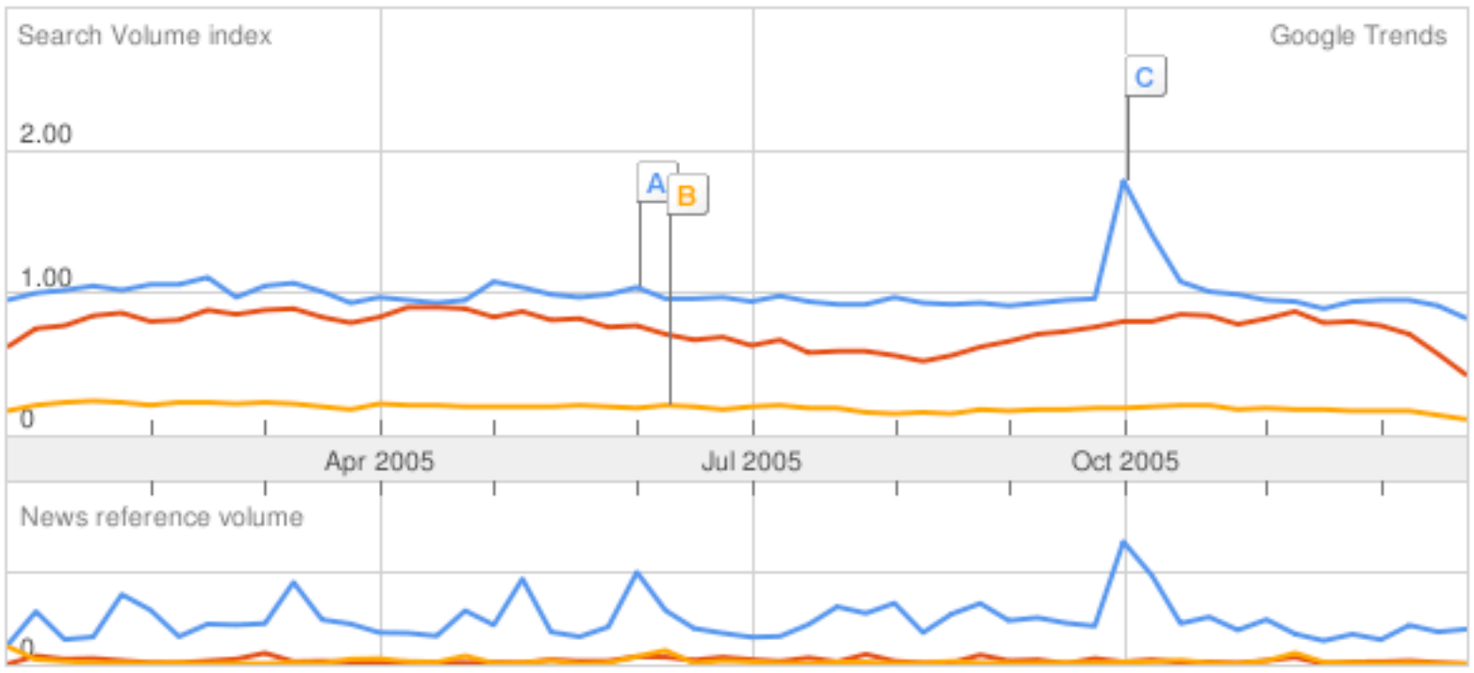
6. Publish in Science

1. Observe and Explore

- Let me/anyone manipulate data easily
- Show it, and basic statistics
- Let me play with visualizations
- When I get an idea, make it easy to see
- Relate to other data stored elsewhere
- Marketplace / community for data


python 1.00
matlab 0.76
fortran 0.16
Search Volume index
Google Trends
2.00
1.00
0
A
B
C
D
E
F
2004
2005
2006
2007
2008
2009
News reference volume
0

python 1.00 matlab 0.75 fortran 0.19

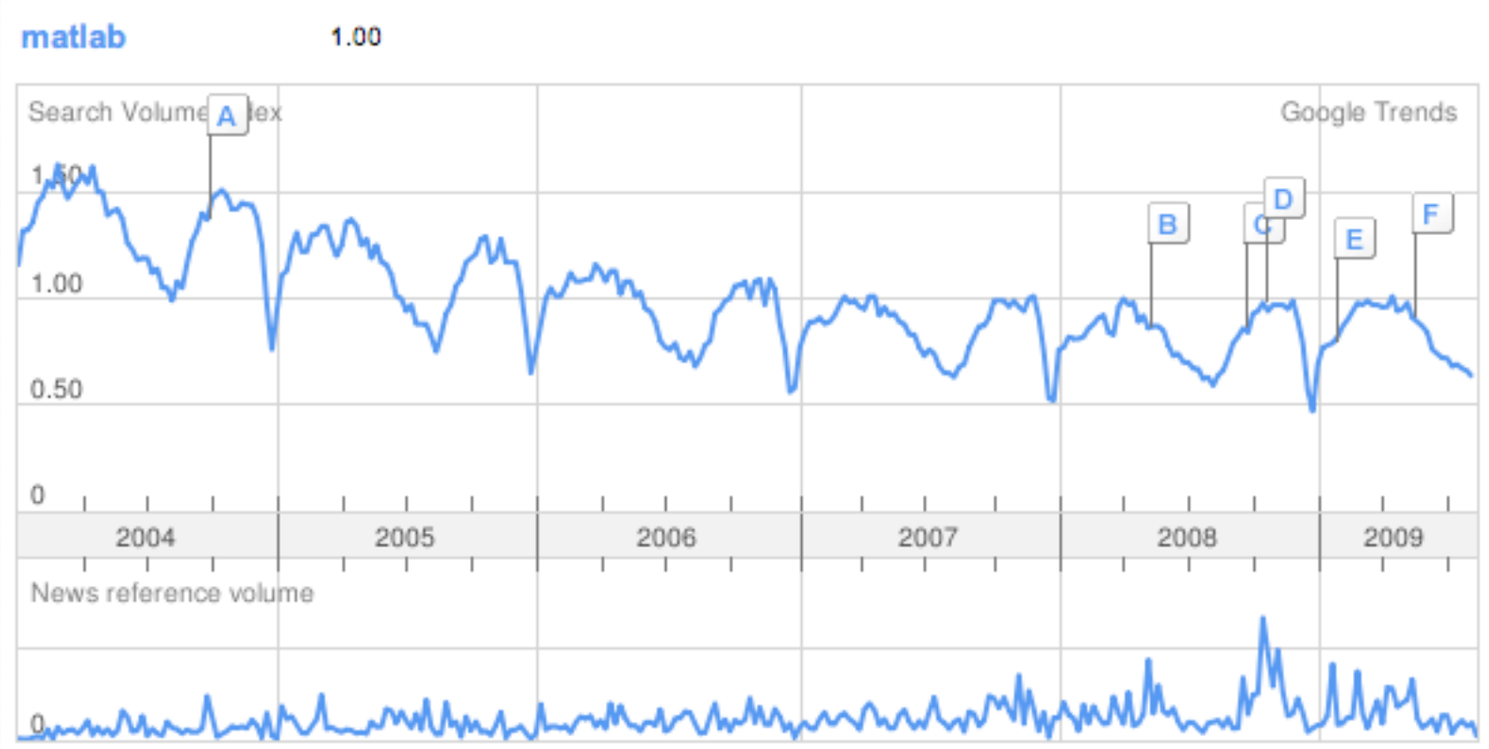


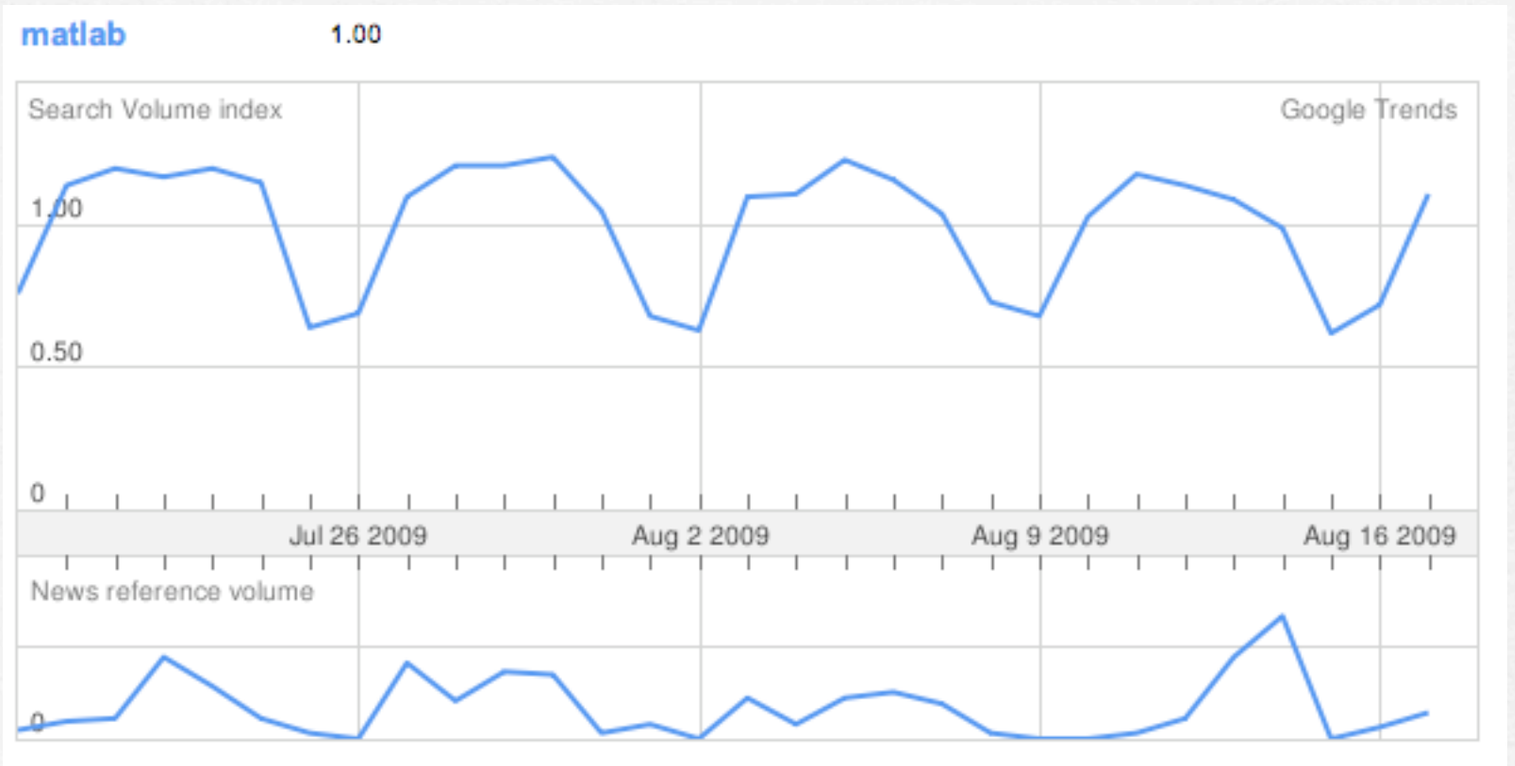
A Python musical heading for West End
Chortle - Jun 6 2005

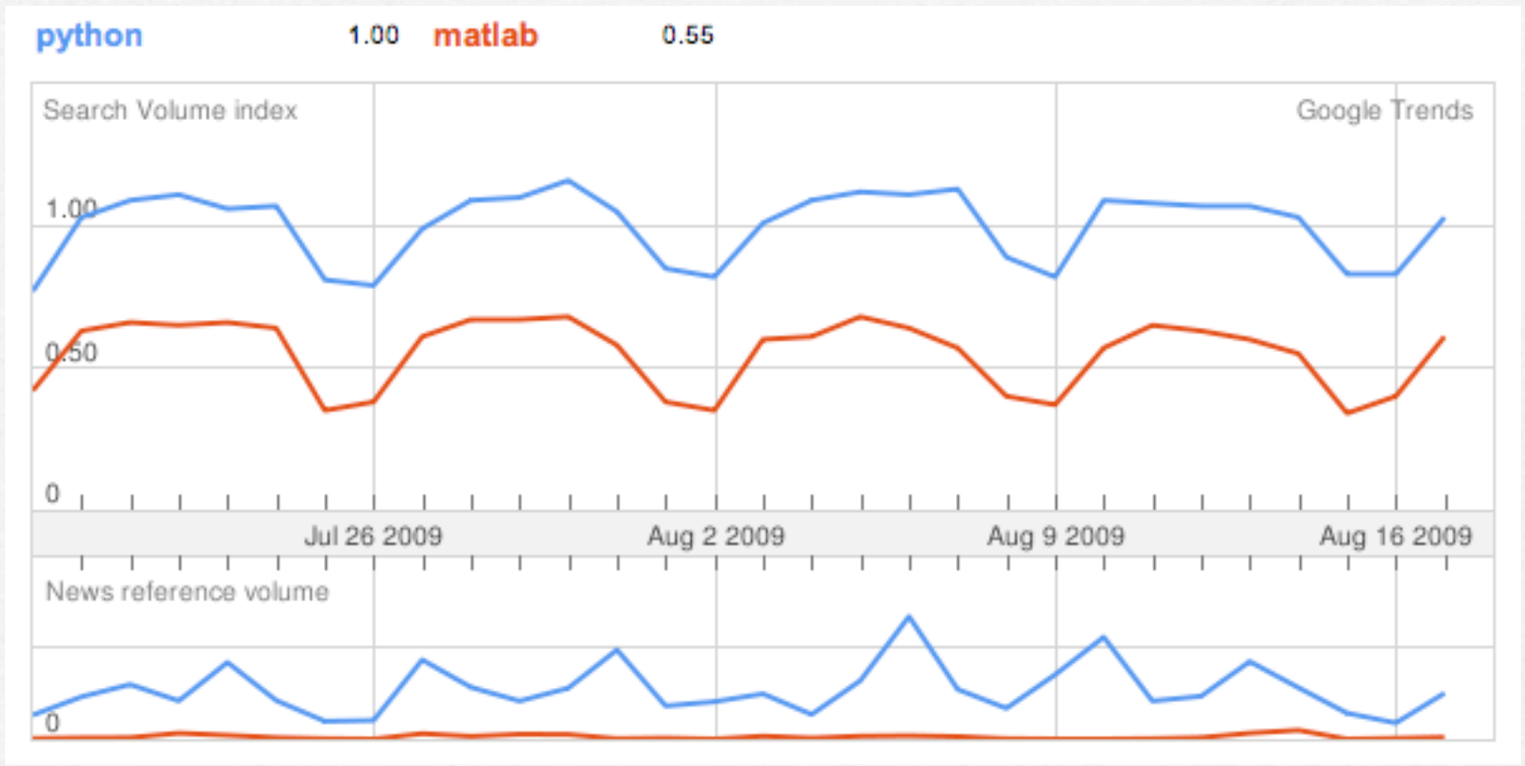
B Intel launches new version 9.0 C++ and FORTRAN compile
PC Pro - Jun 15 2005

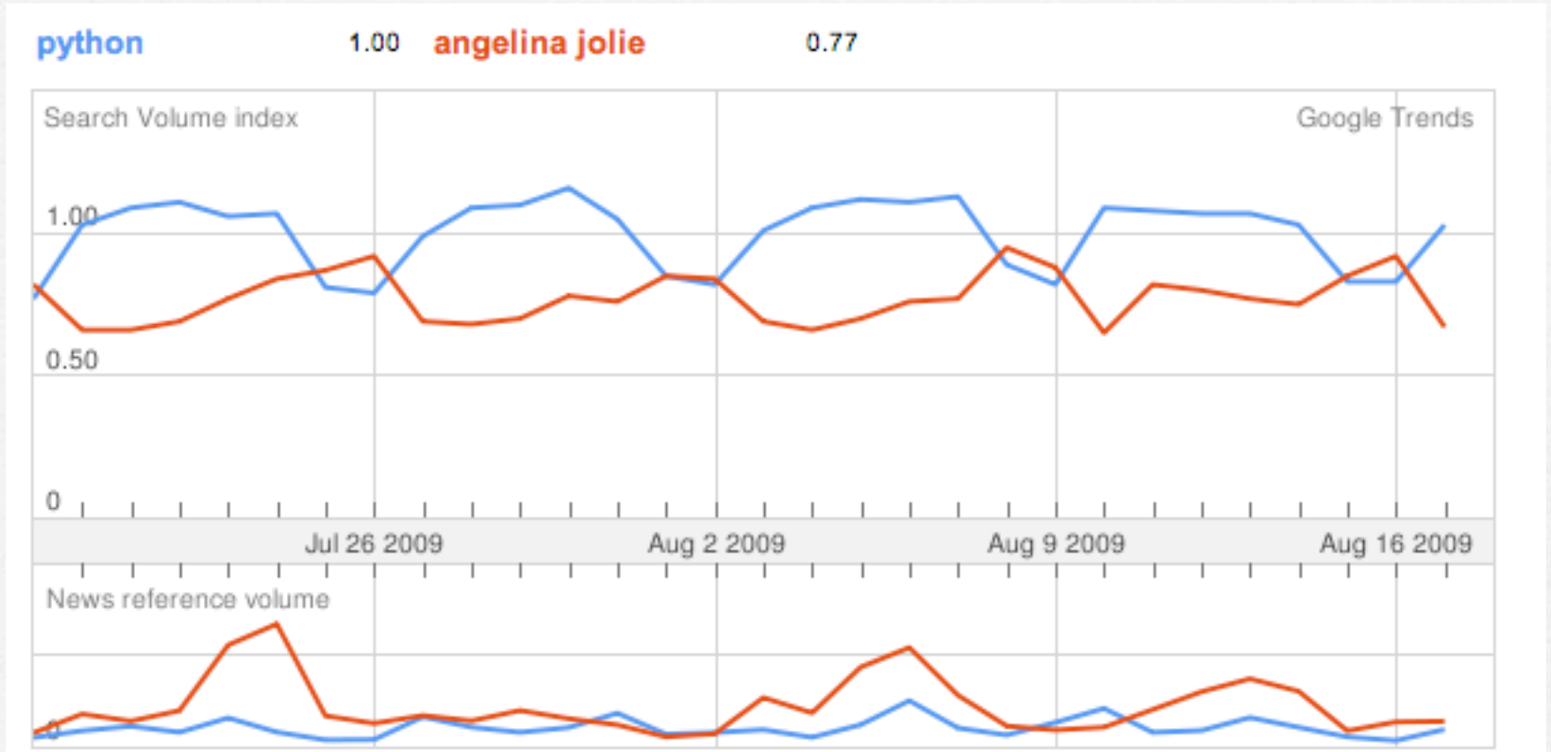
C Python bursts after trying to eat gator
Columbus Ledger-Enquirer - Oct 6 2005

More news results »

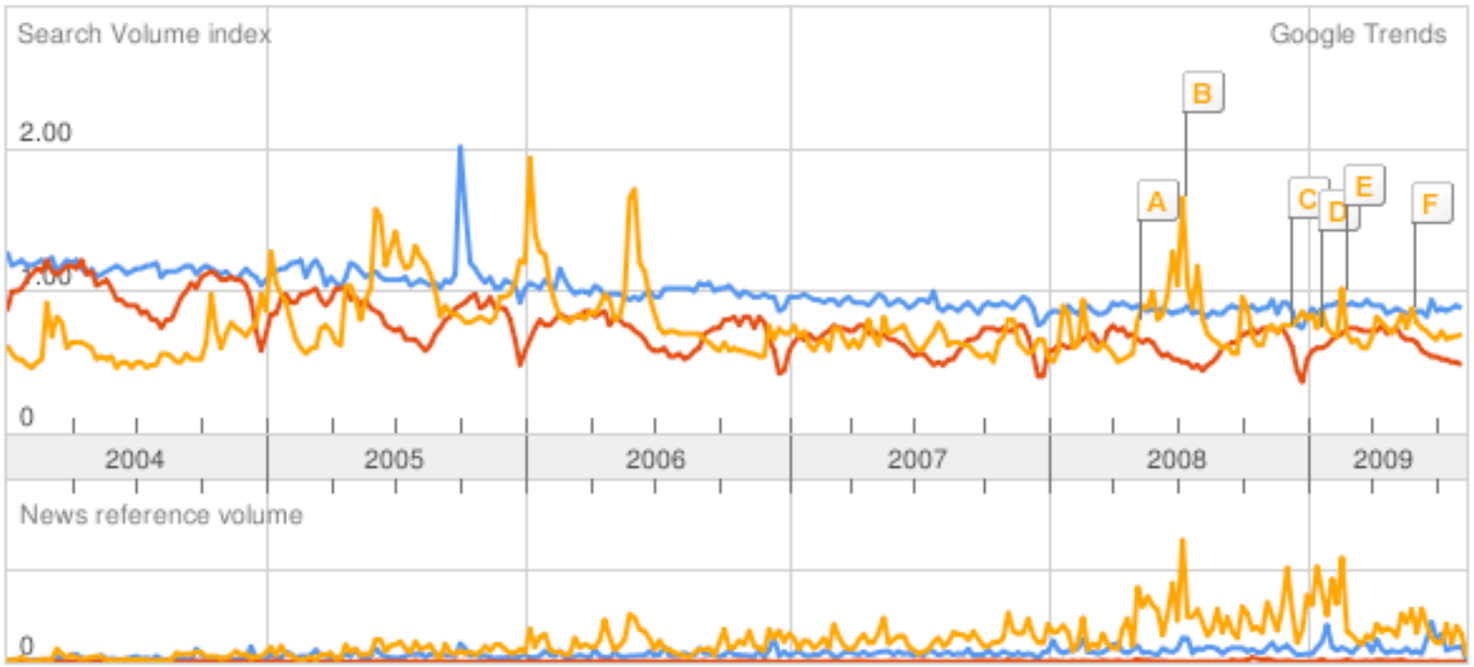

matlab
1.00
Search Volume Index
Google Trends
A
B
C
D
E
F
1.50
1.00
0.50
0
2004
2005
2006
2007
2008
2009
News reference volume
0


matlab
1.00
Search Volume index
Google Trends
1.00
0.50
0
Jul 26 2009
Aug 2 2009
Aug 9 2009
Aug 16 2009
News reference volume
0


python 1.00 matlab 0.55
Search Volume index
Google Trends
1.00
0.50
0
Jul 26 2009
Aug 2 2009
Aug 9 2009
Aug 16 2009
News reference volume
0


python 1.00
angelina jolie 0.77
Search Volume index
Google Trends
1.00
0.50
0
Jul 26 2009
Aug 2 2009
Aug 9 2009
Aug 16 2009
News reference volume
0

python 1.00 matlab 0.76 angelina jolie 0.76



A Angelina Jolie confirms twins
soFeminine.co.uk - May 15 2008

B Angelina Jolie gives birth to twins
Newswatch 50 - Jul 13 2008

C Brad Pitt defends Angelina Jolie
MSNBC - Dec 11 2008

D Angelina Jolie
New York Times Blogs - Jan 22 2009

E Angelina Jolie : Angelina Jolie is not pregnant, says her designe
Entertainment and Showbiz! - Feb 23 2009

F Angelina Jolie Tops Forbes' Celebrity 100
WISC - Jun 4 2009

More news results »

Rank by python

Regions

1. Czech Republic
2. Norway
3. Russian Federation
4. Finland
5. Australia
6. New Zealand
7. United States
8. Sweden
9. India
10. Canada

Cities

1. San Francisco, CA, USA
2. Seattle, WA, USA
3. San Diego, CA, USA
4. Sydney, Australia
5. Beijing, China
6. Chicago, IL, USA
7. Melbourne, Australia
8. Atlanta, GA, USA
9. Los Angeles, CA, USA
10. New York, NY, USA

Languages

1. Czech
2. English
3. Russian
4. Swedish
5. Finnish
6. German
7. Danish
8. Polish
9. French
10. Dutch

Australia Low

New Zealand Moderate

Northern hemisphere

The flu season in the northern hemisphere typically spans from November to March, the northern autumn and winter months.

United States Minimal

2. Generate Hypotheses

- So far, mostly on our own, but...
- Unsupervised learning (clustering)
- Dimensionality reduction
- Summarization of learned models

2. Generate Hypotheses

- 
- g)
- Summarization of learned models

3. Formulate models

- Supervised learning
  (classification and regression)
- Some toolkits for apply-all-methods
- A thriving community helps
- Still requires real programming

HMM model

```
states = ('Rainy', 'Sunny')

observations = ('walk', 'shop', 'clean')

start_p = {'Rainy': 0.6, 'Sunny': 0.4}

transition_p = {
   'Rainy' : {'Rainy': 0.7, 'Sunny': 0.3},
   'Sunny' : {'Rainy': 0.4, 'Sunny': 0.6},
   }

emission_p = {
   'Rainy' : {'walk': 0.1, 'shop': 0.4, 'clean': 0.5},
   'Sunny' : {'walk': 0.6, 'shop': 0.3, 'clean': 0.1},
   }
```

```
maze = """
####################
#....#.....#...#...#
##...#.##.###..#.#.#
#....#.#...#...#..##
###..#.###....##...#
#..#...#...#......##
####################""".strip().splitlines()

X,Y = len(maze[0]), len(maze)

contents = ''.join(maze)

states = [s for (s, c) in enumerate(contents)) if c != '#']

def neighbors(s):
    return [s1 for s1 in (s-1, s+1, s-X, s+X) if s1 in states] or [s]

def obs(s):
    return ''.join([contents[s1] for s1 in  (s-1, s+1, s-X, s+X)])

start_p = uniform(states)
transition_p = dict((s, uniform(neighbors(s))) for s in states)
emission_p = dict((s, zdict([(obs(s), 1)])) for s in states)
```

Game Theory model

```
def strats(acts): return [{'A': a, 'K': b} for a in acts for b in acts]
s1s = strats('rk')
s2s = strats('cf')
deals = ['AA', 'AK', 'AK', 'KK', 'KA', 'KA']

def EU(s1, s2):
    return mean([U(d, s1, s2) for d in deals])

def U((d1, d2), s1, s2):
    if s1[d1] == 'k': return cmp(d2, d1)
    if s2[d2] == 'f': return +1
    return 2 * cmp(d2, d1)

def nash(s1, s2):
    return (s1 == argmax((EU(s, s2),s) for s in s1s) and
            s2 == argmin((EU(s1, s),s) for s in s2s))

for s1 in s1s:
    for s2 in s2s:
        print ' %5.2f%s &' % (EU(s1, s2), ('*' if nash(s1, s2) else ' ')),
    print
```

```
-:** poker.py     15% (15,0)    (Python)
bash-3.2$ python -i poker.py
  0.00  &  -0.17  &   1.17  &   1.00  &
  0.33  &   0.00* &   0.50  &   0.17  &
 -0.33  &  -0.17  &   0.67  &   0.83  &
  0.00  &   0.00  &   0.00  &   0.00  &
>>>
```

Naive Bayes word model

```
@memo
def segment(text):
    "Return a list of words that is the best segmentation of text."
    if not text: return []
    candidates = ([first]+segment(rem) for first,rem in splits(text))
    return max(candidates, key=Pwords)

def splits(text, L=20):
    "Return a list of all possible (first, rem) pairs, len(first)<=L."
    return [(text[:i+1], text[i+1:])
            for i in range(min(len(text), L))]

def Pwords(words):
    "The Naive Bayes probability of a sequence of words."
    return product(Pw(w) for w in words)
```

$$P(W_{1:n}) = \prod_{k=1:n} P(W_k)$$

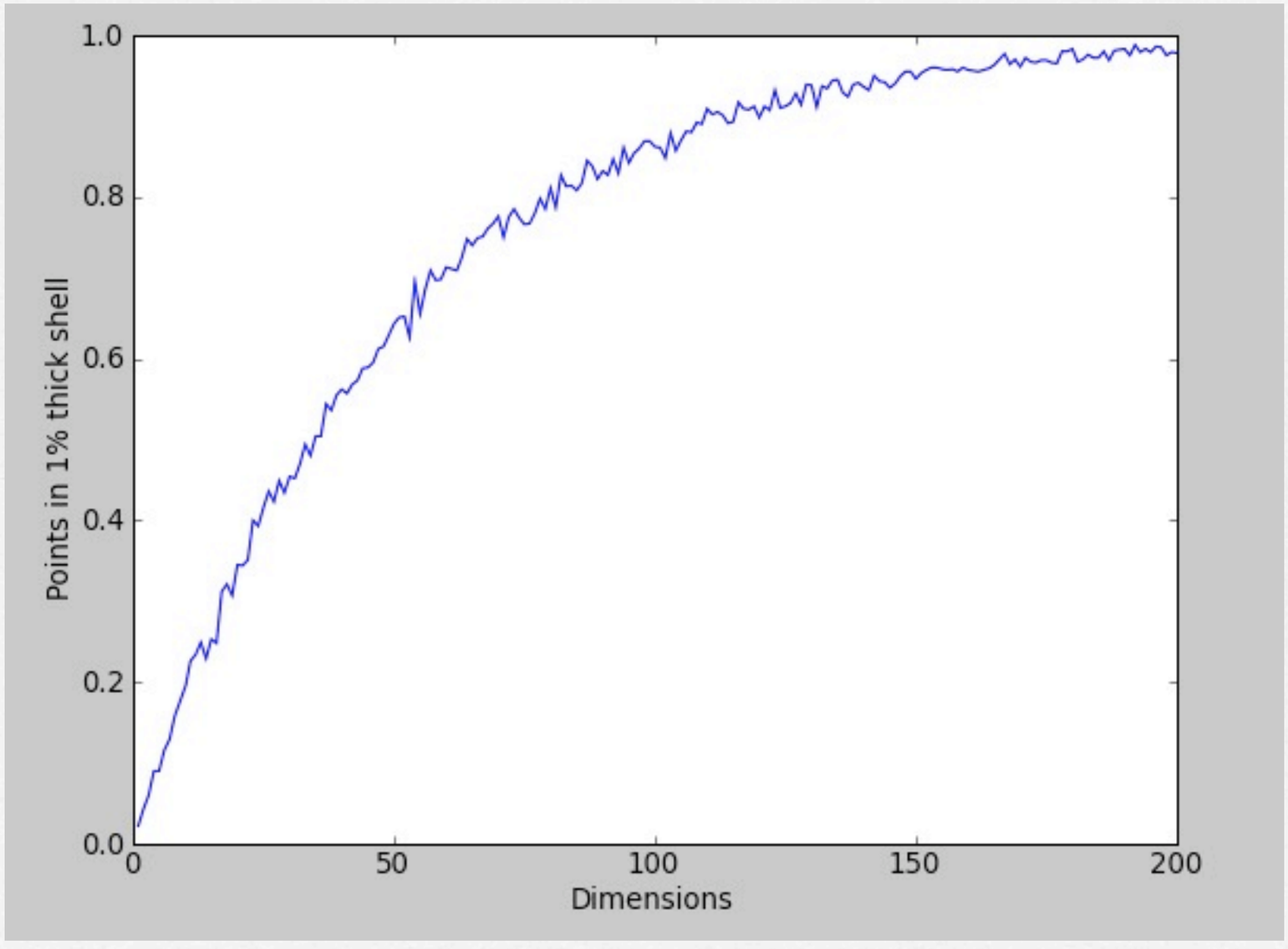
```
-:--  ngrams.py     6% (32,0)    (Python)
```

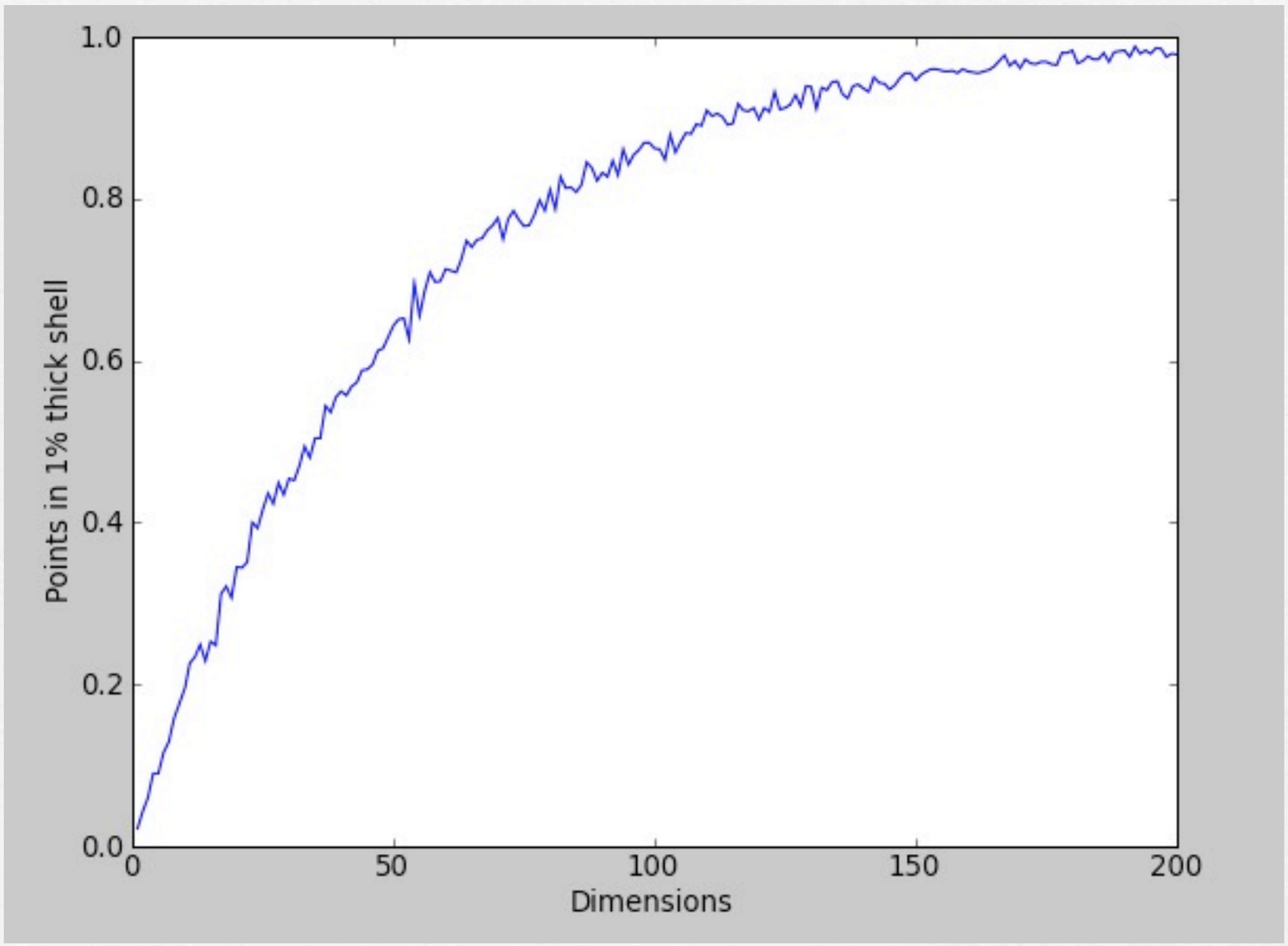
```
>>> segment('scipyconference')
['scipy', 'conference']
>>> segment('scipyconferencekeynotetalk')
['scipy', 'conference', 'keynote', 'talk']
>>> segment('virtualrealityatoolforthehighlyquantitativestudyofanimalbehavior')
['virtual', 'reality', 'a', 'tool', 'for', 'the', 'highly', 'quantitative', 'study', 'of', 'animal', 'behavior']
>>>
```

4. Test predictions

- [ ] Decide what to test next (active learning)
- [ ] Automate tests - significance, sensitivity
- [ ] Do what I did last time
- [ ] Find and convert legacy data

Curse of Dimensionality

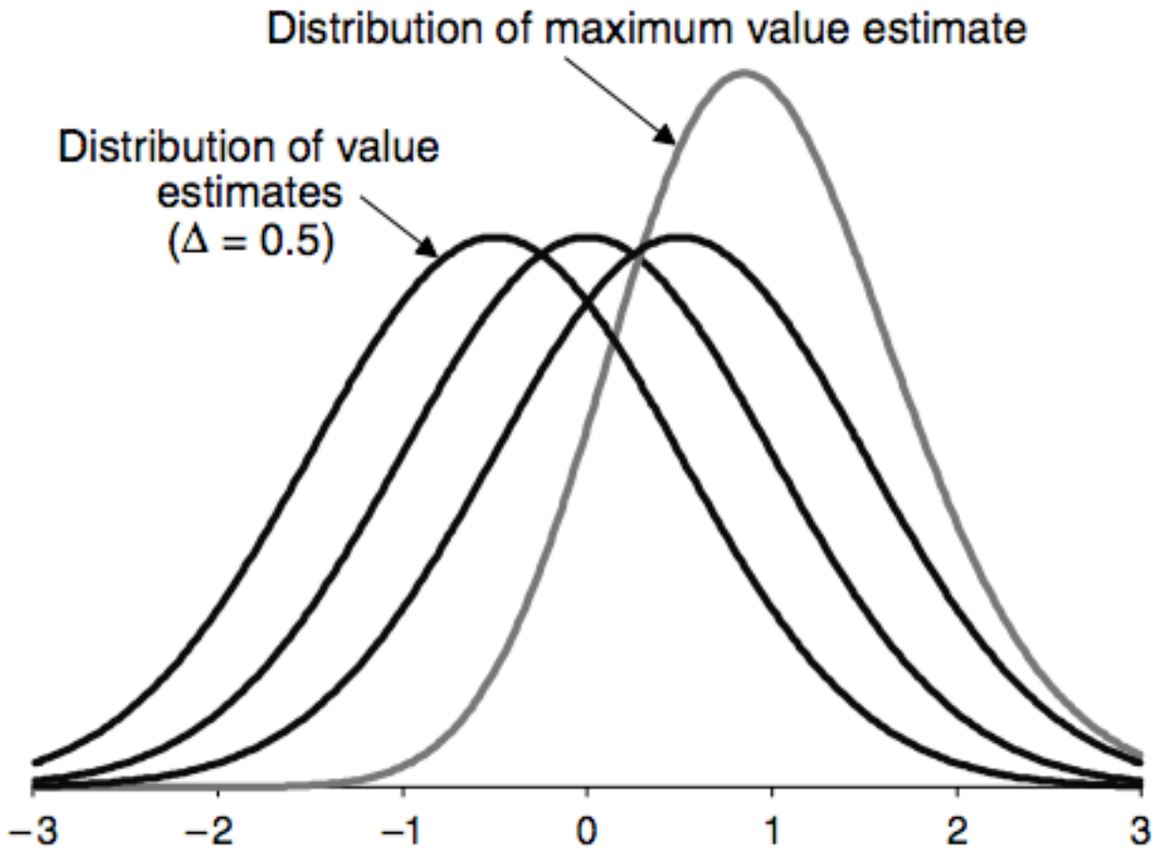





```
def sample(d=2, N=1000): return [[uniform(0.,1.) for i in range(d)] for _ in range(N)]

def corner_count(points):
    return mean([any([(d < .01 or d > .99) for d in p]) for p in points])

def go(Ds=range(1,201)):
    plot(Ds, [corner_count(sample(d)) for d in Ds])
```

Optimizer's curse

Figure 3 **The Distribution of Maximum Value Estimates with Separation Between Alternatives**



Δ	Expected disappointment	Probability of correct choice
0.0	0.85	0.33
0.2	0.66	0.42
0.4	0.51	0.50
0.6	0.39	0.59
0.8	0.30	0.66
1.0	0.22	0.73
1.2	0.17	0.78
1.4	0.12	0.83
1.6	0.10	0.87
1.8	0.07	0.90
2.0	0.05	0.92
2.2	0.03	0.94
2.4	0.02	0.95
2.6	0.01	0.97
2.8	0.01	0.98
3.0	0.00	0.98

5. Modify and repeat

- Notebook (Sage)
- Refactoring of interactive session
- DWIM, DWID

6. Publish

- Pretty graphics
- Accurate statistics
- Verifiable (1-to-1 math to code)
- Reproducible (clear code, intent)
- Repeatable (open source stack, data)

Good style / taste

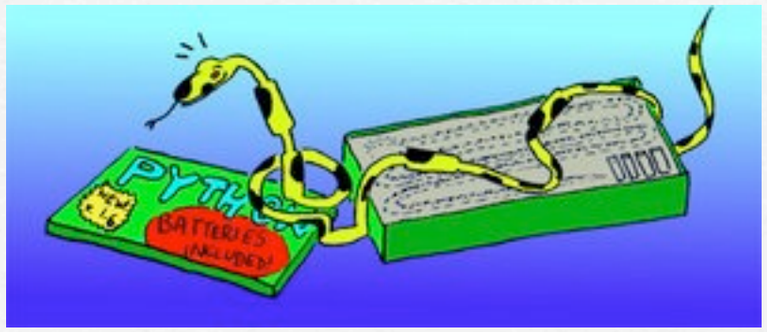
- Be explicit
- Be concise
- Be consistent
- Be helpful
- Don't be obscure
- Use the right abstractions - stratify
- "Elegance is not optional" - R. O'Keefe

My Demands

(OK, polite requests)

To Guido:

- Batteries included:



  NumPy in the standard release!!

- Expressions, not statements!

  Good: `{1:a, 2:b}, reversed, sorted`

  Bad: `no defaultdict(int,...), removed, extended`

To Travis Oliphant:

- SciPy in the standard release (??)
- ... or at least NumPy

To David Beazley(?):

- Fix the Global Interpreter Lock!
- Let me run with multiple cores
- Some frameworks to help...

To everyone:

- Documentation
- Examples
- Unit tests
- Tutorials
- Mentoring

Good documentation:

```
next-line is an interactive compiled Lisp function in `simple.el'.
(next-line &optional arg try-vscroll)

The command C-x C-n can be used to create
a semipermanent goal column for this command.
Then instead of trying to move exactly vertically (or as close as possible),
this command moves to the specified goal column (or as close as possible).
The goal column is stored in the variable `goal-column', which is nil
when there is no goal column.

If you are thinking of using this in a Lisp program, consider
using `forward-line' instead.  It is usually easier to use
and more reliable (no dependence on goal column, etc.).
```

Questions?