

Biology, Computers & Python

Michael Schatz

Sept 3, 2013

QB Bootcamp Lecture I





Outline

Part 1: Overview & Fundamentals

- Overview of Computer Systems
- Python Primer

Part 2: Sequence Analysis Theory

Part 3: Genomics Resources

Part 4: Unix Primer

Part 5: Example Analysis

Modern Biology Challenges



The foundations of biology will continue to be *observation, experimentation, and interpretation*

- Technology will continue to push the frontier
- Measurements will be made *digitally* over large populations, at extremely high resolution, and for diverse applications

Rise in Quantitative and Computational Demands

1. *Experimental design*: selection, collection & metadata
2. *Observation*: measurement, storage, transfer, computation
3. *Integration*: multiple samples, assays, analyses
4. *Discovery*: visualizing, interpreting, modeling

Ultimately limited by the human capacity to execute extremely complex experiments and interpret results

How do we draw conclusions?

- Comparison & Correlations: How does X compare to Y?

X	Y
Exomes of kids with autism	Exomes of kids that do not
Genomes of Europeans	Genomes of non-Europeans, mammals, ...
Gene expression in mutants	Gene expression in wild type
Firing patterns of mutant fly neurons	Firing patterns of wild type

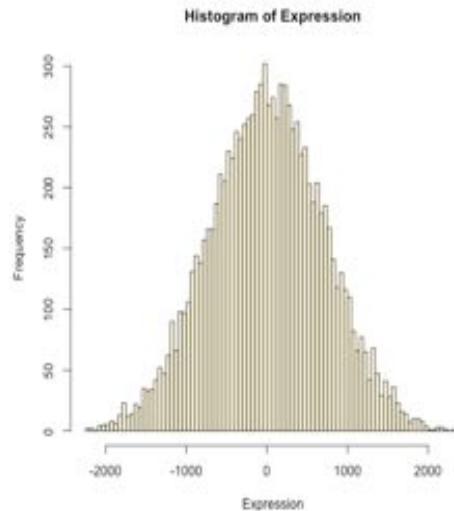
- Modeling & Predictions: How will X respond to Y?

X	Y
Mutant tomatoes	Increased temperatures
Human Microbiome	Probiotic treatments
Gene expression in mice	Knockout of transcription factor
Firing rate in flies	Decreased sodium levels

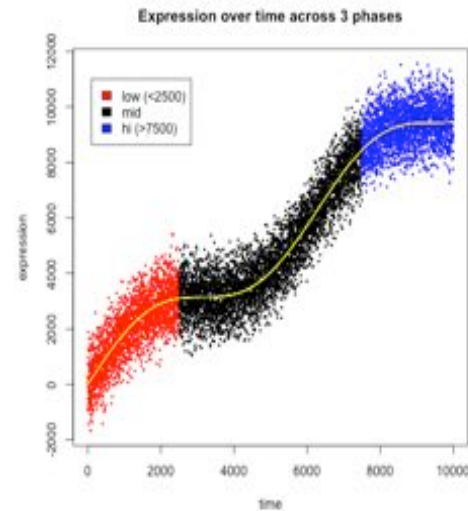
How do we DRAW conclusions?

-902.473
242.817
-872.453
73.9297
236.169
46.7525
975.014
716.563
-533.971
-120.282
725.12
-736.76
176.156
189.224
1847.46
-159.099
-56.4754
-973.626
1181.9
-315.455
-1480.43
215.293
-747.505
682.577
...

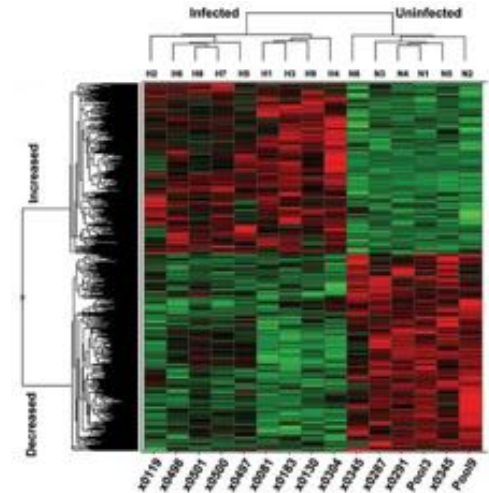
Histogram



Scatterplot



Heatmap



Data and data transformations are ubiquitous in science
Data are too numerous and transformations are too complex to do by hand
==> Mendel: 100k observations, 10 years
==> HiSeq 2000: 600B observations, 10 days
==> Make friends with your computational tools

What is a computer?

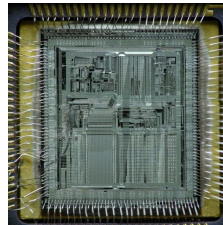
[hardware]



Hard Drive
Permanent Storage – 1TB
(big, slow, cheap)



RAM
Working Storage – 8 GB
(small, fast, expensive)



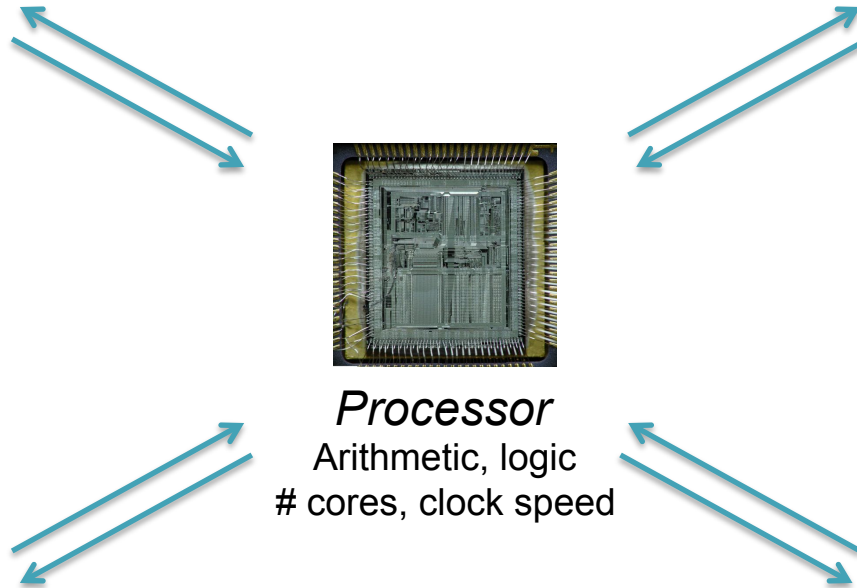
Processor
Arithmetic, logic
cores, clock speed



Display
Human Interface



Network
Computer Interface
Home: 10Mb/s, CSHL: 1Gb/s

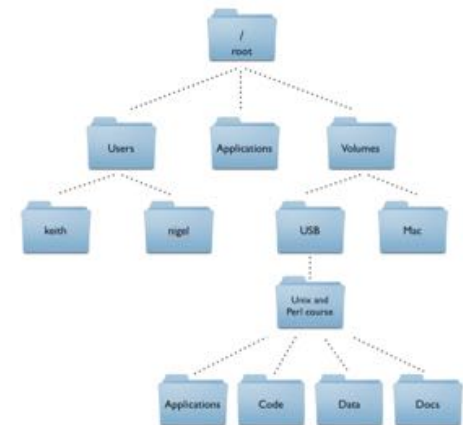


What is a computer?

[software]

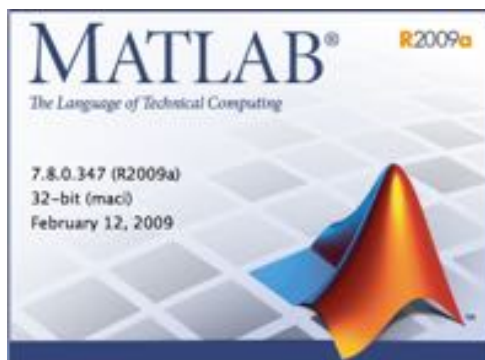


Office Applications
Presentations, Documents
Simple statistics and plots



Files / Data
Papers, sequences,
measurements

Operating System
Mission Control
Windows, Mac, Unix, iOS



Scientific Applications
Specialized Analysis
Commercial



Code / Scripts
Research Applications
Academic

Programming I01

Mozart
Sinfonia Concertante in Eb
for Violin and Viola
K. 364

Allegro maestoso.

Obol.

Corni in Es.

Violino principale.

Viola principale.
(scuola del maestro non più soli)

Violino I.

Violino II.

Viola I.

Viola II.

Violoncello.

Contrabbasso.

www.viola-in-music.com

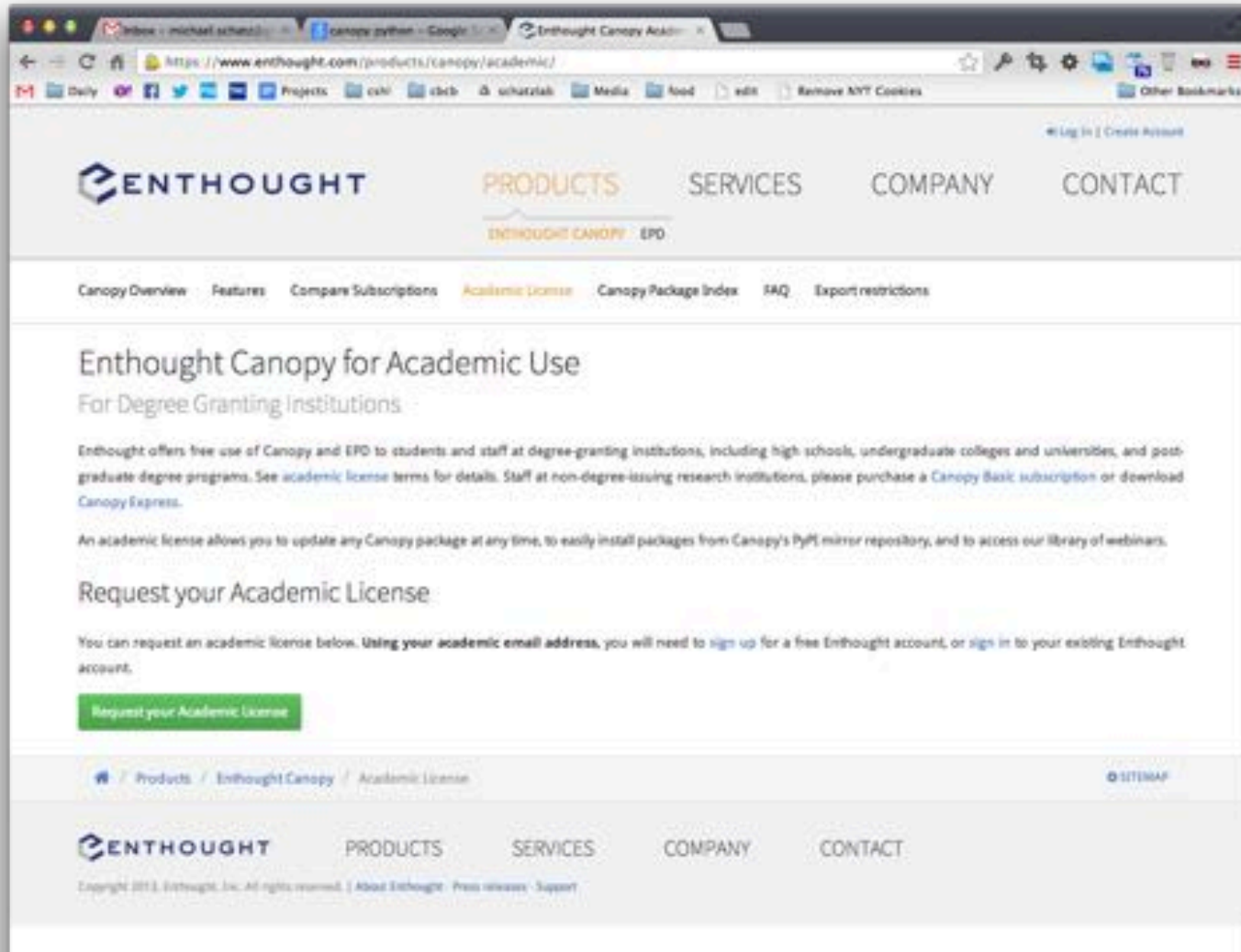
```

1 // Measure time from here
2 for (int i = 0; i < 100; i++) { // Do the next 100 times
3     //Generate numerator = [array objectAtIndex:]
4     NSString wstr;
5     while (i = [enumerator nextObject]) {
6     }
7 }
8 // Measure time from here
9 for (int i = 0; i < 100; i++) { // Do the next 100 times
10    for (NSString wstr in array) {
11    }
12 }
13 // Measure time from here
14 for (int i = 0; i < 100; i++) { // Do the next 100 times
15    for (int j = 0; j < [array count]; j++)
16        NSString wstr = [array objectAtIndex:];
17 }
18 // Measure time from here
19 TIME_T time0 = clock(); // Measure time from here
20
21 double t1 = ((double)(time0-time1)/CLOCKS_PER_SEC)*1000;
22 double t2 = ((double)(time0-time1)/CLOCKS_PER_SEC)*1000;
23 double t3 = ((double)(time0-time1)/CLOCKS_PER_SEC)*1000;
24 printf("enumerator next wstr", t1);
25 printf("for enumeration next wstr", t2);
26 printf("for loop next wstr", t3);
27
28 return 0;

```

A software program is like sheet music for the orchestra inside your computer
Static, written representations of an active process

Programming with Python



<https://www.enthought.com/products/canopy/academic/>
<http://www.codecademy.com/tracks/python>

Questions?

<http://schatzlab.cshl.edu>