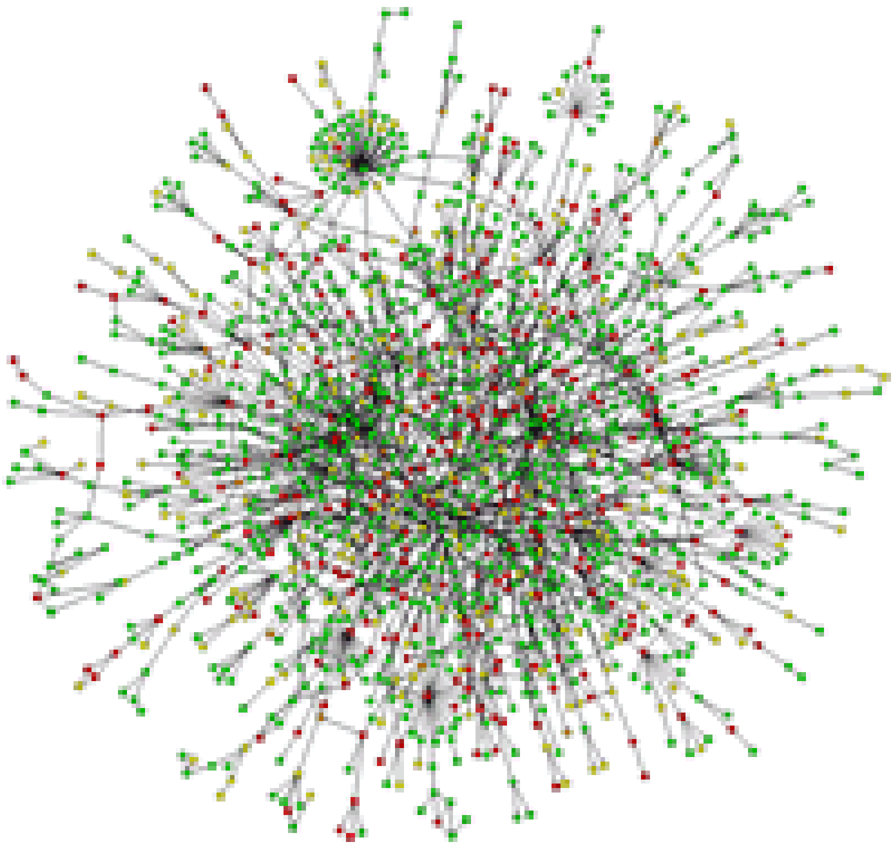
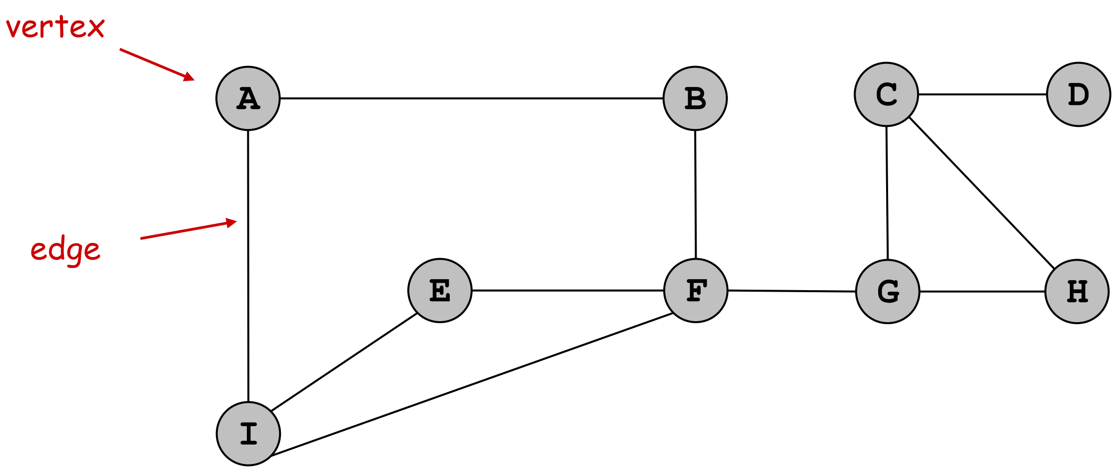
Networks



Graph Data Type

- Graph = data type that represents pairwise connections.
- Vertex = element.
- Edge = connection between two vertices.

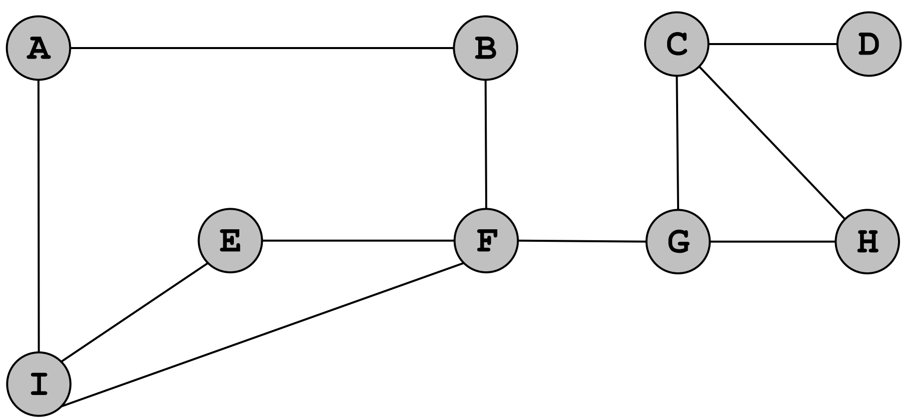


Graph API

Graph data type.

public class Graph	(graph with String vertices)	
	Graph()	*create an empty graph*
	Graph(In in)	*read graph from input stream*
void	addEdge(String v, String w)	*add edge v-w*
Iterable<String>	adjacentTo(String v)	*neighbors of v*

to support use with foreach




```
% more tiny.txt
A/B/I
B/A/F
C/D/G/H
D/C
E/F/I
F/B/E/G
G/C/F/H
H/C/G
I/A/E/F
```

Kevin Bacon Game

Game. Find (shortest) chain of movies connecting a performer to Kevin Bacon.

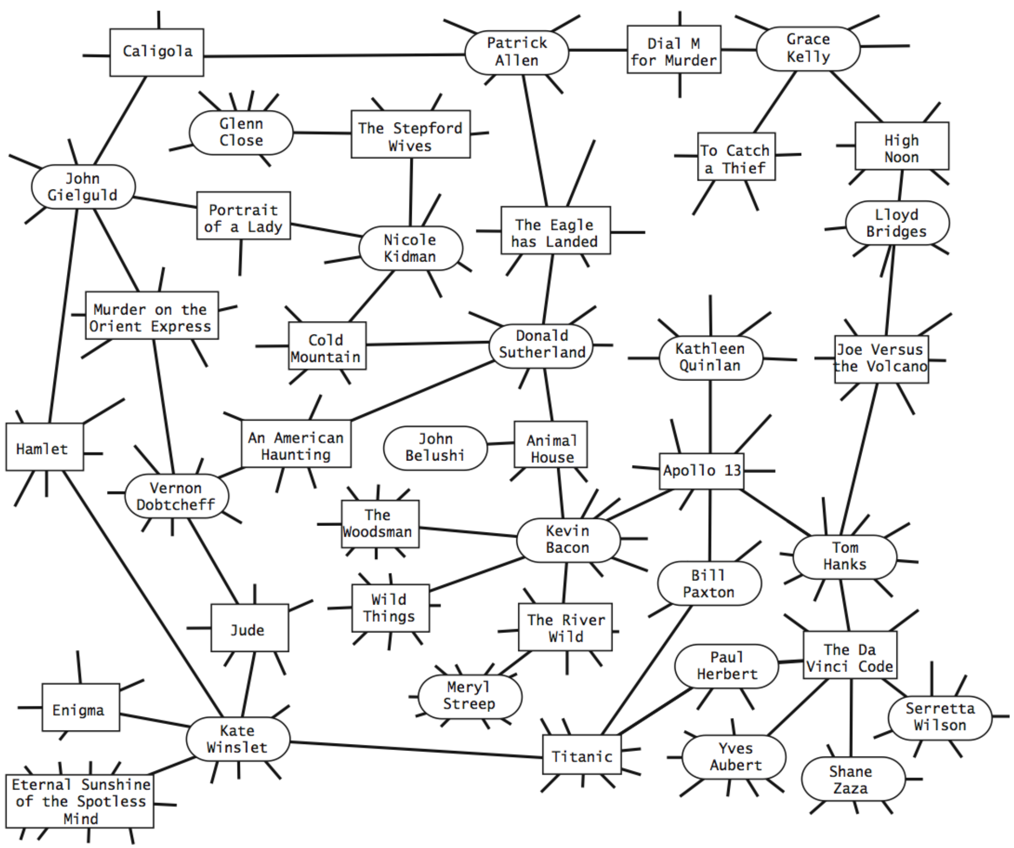
performer	was in	with
Kevin Kline	French Kiss	Meg Ryan
Meg Ryan	Sleepless in Seattle	Tom Hanks
Tom Hanks	Apollo 13	Kevin Bacon
Kevin Bacon		

Internet Movie Database

Q. How to represent the movie-performer relationships?

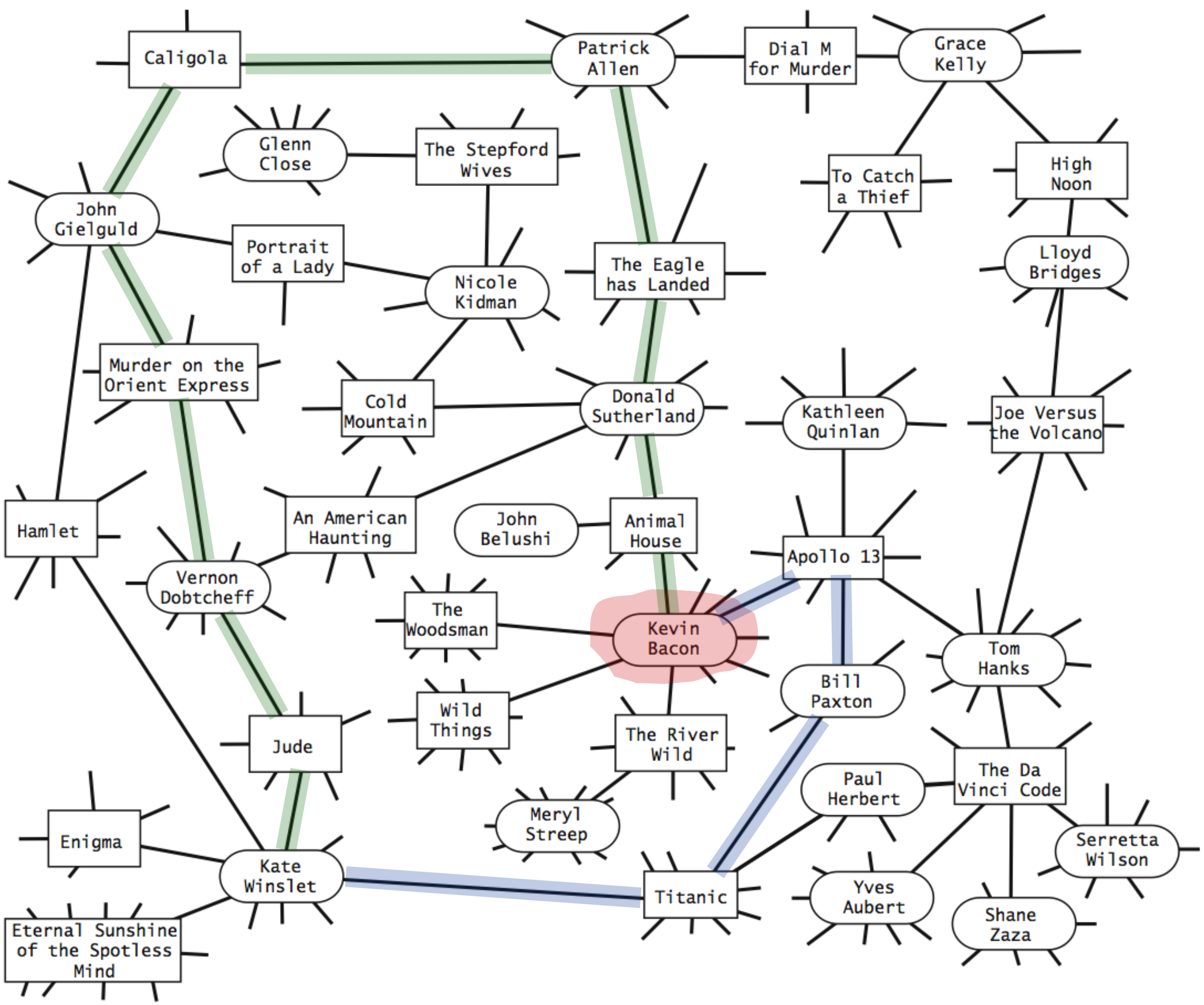
A. Use a graph.

- Vertex: performer or movie.
- Edge: connect performer to movie.



Computing Bacon Numbers

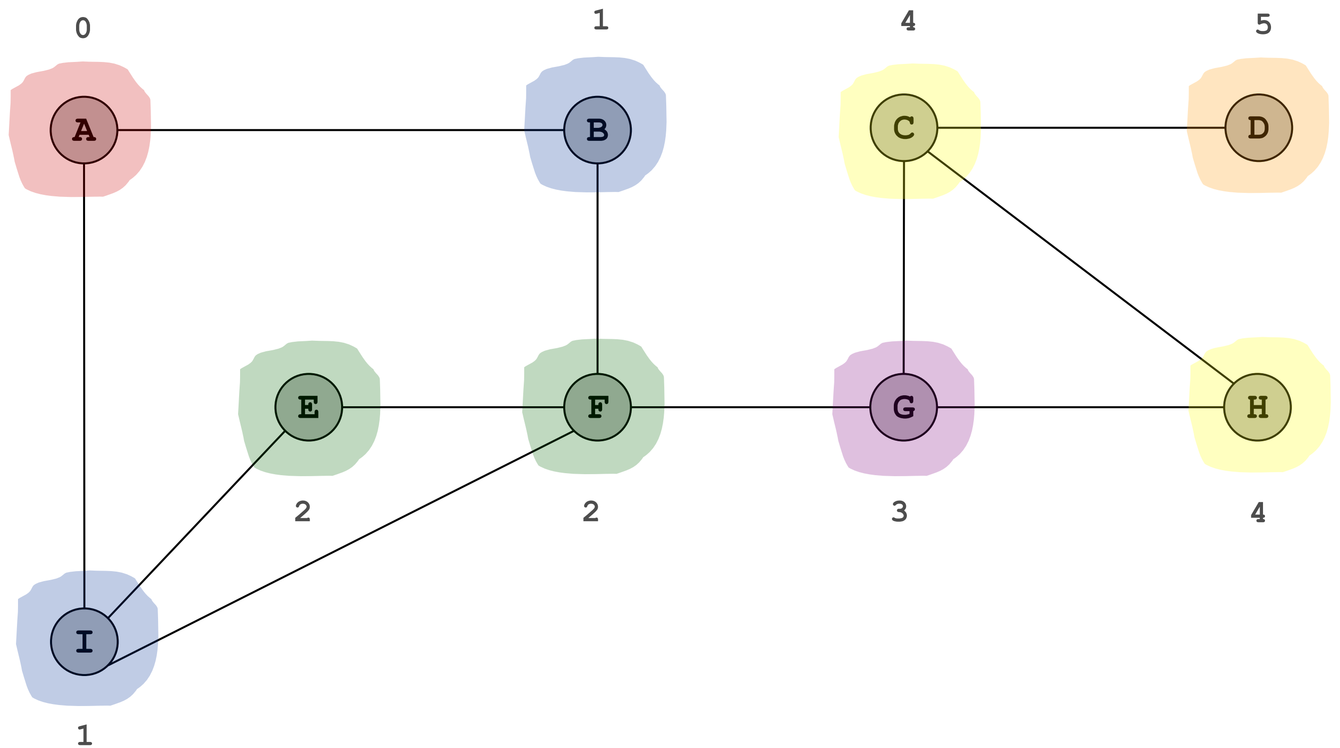
How to compute. Find shortest path in performer-movie graph.



Computing Shortest Paths

To compute shortest paths:

- Source vertex is at distance 0.
- Its neighbors are at distance 1.
- Their remaining neighbors are at distance 2.
- Their remaining neighbors are at distance 3.
- ...



Breadth First Search

Goal. Given a vertex s, find shortest path to every other vertex v.

BFS from source vertex s

Put s onto a FIFO queue.

Repeat until the queue is empty:

- dequeue the least recently added vertex v
- add each of v's unvisited neighbors to the queue, and mark them as visited.

Key observation. Vertices are visited in increasing order of distance from s because we use a FIFO queue.

Path Finder API

Path finder API.

public class PathFinder	*(data type to compute shortest paths)*
`PathFinder(Graph G, String s)`	*process graph G with source s*
`int distanceTo(String v)`	*return shortest distance between s and v*
`void showPath(String v)`	*print shortest path between s and v*

Design principles.

- Decouple graph algorithm from graph data type.
- Avoid feature creep.

Computing Bacon Numbers: Java Implementation

```
public class Bacon {
   public static void main(String[] args) {
      In in = new In(args[0]);
      Graph G = new Graph(in);

      String s = "Bacon, Kevin";
      PathFinder finder = new PathFinder(G, s);

      while (!StdIn.isEmpty()) {
         String performer = StdIn.readLine();
         finder.showPath(performer);
      }
   }
}
```

← read in the graph from a file

← create object to return shortest paths

← process queries

```
% java Bacon top-grossing.txt
Stallone, Sylvester
Rocky III (1982)
Tamburro, Charles A.
Terminator 2: Judgment Day (1991)
Berkeley, Xander
Apollo 13 (1995)
Bacon, Kevin
```

```
% java Bacon top-grossing.txt
Goldberg, Whoopi
Sister Act (1992)
Grodénchik, Max
Apollo 13 (1995)
Bacon, Kevin

Tilghman, Shirley
```

Breadth First Searcher: Preprocessing

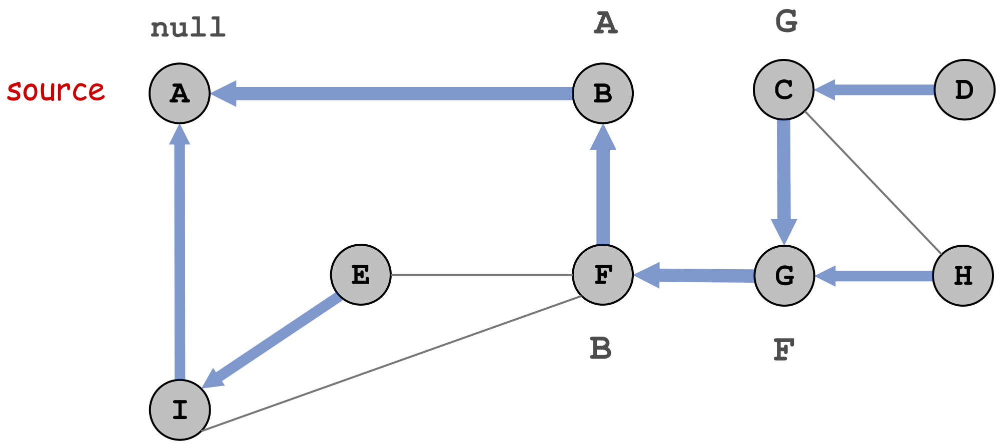
```
public class PathFinder {
   private ST<String, String>  prev = new ST<String, String>();
   private ST<String, Integer> dist = new ST<String, Integer>();

   public PathFinder(Graph G, String s) {
      Queue<String> q = new Queue<String>();
      q.enqueue(s);
      dist.put(s, 0);
      while (!q.isEmpty()) {
         String v = q.dequeue();
         for (String w : G.adjacentTo(v)) {
            if (!dist.contains(w)) {
               q.enqueue(w);
               dist.put(w, 1 + dist.get(v));
               prev.put(w, v);
            }
         }
      }
   }
```

Breadth First Searcher: Printing the Path

To print shortest path: follow `prev[]` from vertex `v` back to source `s`.

- Print `v`, `prev[v]`, `prev[prev[v]]`, …, `s`.
- Ex: shortest path from `C` to `A`: `C – G - F - B - A`



key	prev	dist
A	-	0
B	A	1
C	G	4
D	C	5
E	I	2
F	B	2
G	F	3
H	G	4
I	A	1

symbol tables

```
public void showPath(String v) {
    while (prev.contains(v)) {
        StdOut.println(v);
        v = prev.get(v);
    }
}
```

Running Time Analysis

Analysis. BFS scales to solve huge problems.

data File	movies	performers	edges	read input	build graph	BFS	show
G.txt	1,288	21,177	28K	0.26 sec	0.52 sec	0.32 sec	0 sec
PG13.txt	2,538	70,325	100K	0.31 sec	0.99 sec	0.72 sec	0 sec
action.txt	14,938	139,861	270K	0.72 sec	2.8 sec	2.0 sec	0 sec
mpaa.txt	21,861	280,624	610K	2.1 sec	7.5 sec	5.5 sec	0 sec
all.txt	285,462	933,864	3.3M	15 sec	56 sec	39 sec	0 sec

60MB (all.txt)

data as of April 9, 2007

Data Analysis

Exercise. Compute histogram of Kevin Bacon numbers.
Input. 285,462 movies, 933,864 actors.

Bacon #	Frequency
0	1
1	2,249
2	218,088
3	561,161
4	111,149
5	7,905
6	903
7	100
8	14
∞	32,294

Buzz Mauro, Jessica Drizd, Pablo Capussi
Argentine short film *Sweet Dreams* (2005) → 8

Fred Ott, solo actor in
Fred Ott Holding a Bird (1894) → ∞

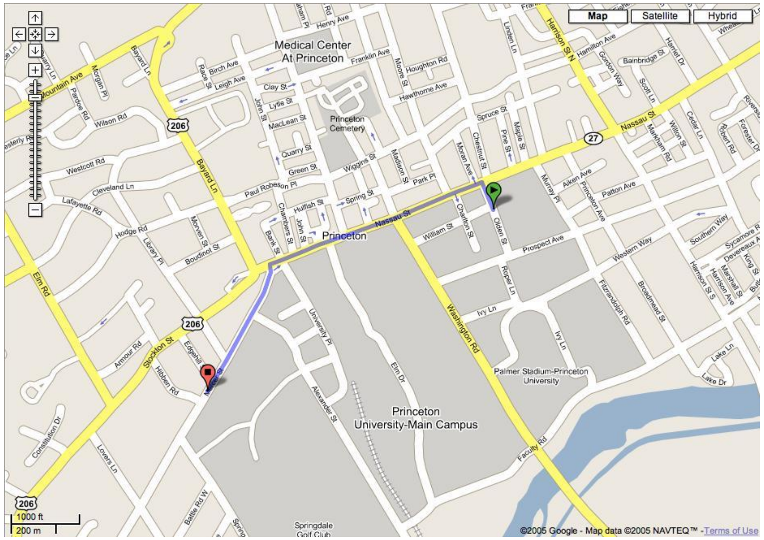
data as of April 9, 2007

Applications of Breadth First Search

More BFS applications.

- Particle tracking.
- Image processing.
- Crawling the Web.
- Routing Internet packets.
- ...

Extensions. Google maps.



Linked list. Ordering of elements.

Binary tree. Hierarchical structure of elements.

Graph. Pairwise connections between elements.

Data structures.

- Queue: linked list.
- Set: binary tree.
- Symbol table: binary tree.
- Graph: symbol table of sets.
- Breadth first searcher: graph + queue + symbol table.

Importance of data structures.

- Enables us to build and debug large programs.
- Enables us to solve large problems efficiently.

Erdös Number

Erdös Numbers

Paul Erdös. Legendary, brilliant, prolific mathematician who wrote over 1500 papers!

What's your Erdös number?

- Co-authors of a paper with Erdös: 1.
- Co-authors of those co-authors: 2.
- And so on …

Paul Erdös (1913-1996)

Erdös #	Frequency
0	1
1	502
2	5,713
3	26,422
4	62,136
5	66,157
6	32,280
7	10,431
8	3,214
9	953
10	262
11	94
12	23
13	4
14	7
15	1
∞	4 billion +

Erdös has a Bacon number!

Erdös has a Kevin Bacon number of 4.

```
% java Bacon cast.txt
Erdös, Paul
N Is a Number (1993)
Patterson, Gene
Box of Moon Light (1996)
Turturro, John
Cradle Will Rock (1999)
Tim Robbins
Mystic River (2003)
Bacon, Kevin
```

… but so far, Kevin Bacon doesn't have an Erdös number.

Erdös-Bacon Numbers

Sum of your Erdös and Bacon numbers.

- For most people: *infinity!*
- But for some …

Prof. of Computer Science Bernard Chazelle

Erdös number 2:
Bernard– Janos Pach-- Erdös

Bacon number 3!
Bernard in *Guy and Madeline on a Park Bench* w/ Jerry Quinn
Quinn in *Mr North* w/ Mary Stuart Masterson
Masterson in *Digging to China* w/Kevin Bacon

Erdös-Bacon number 5

Erdös-Bacon Numbers

Abigail A. Baird, Jerome Kagan, Thomas Gaudette, Kathryn A. Walz, Natalie Hershlag and David A. Boas
"Frontal Lobe Activation during Object Permanence: Data from
Near-Infrared Spectroscopy." *NeuroImage* Vol. 16, Issue 4,
Aug. 2002, pp. 1120-1126.

Erdös number 4

Stage name: Natalie Portman

Bacon number 2

Erdös-Bacon number 7

Erdös-Bacon Numbers

Danica McKellar

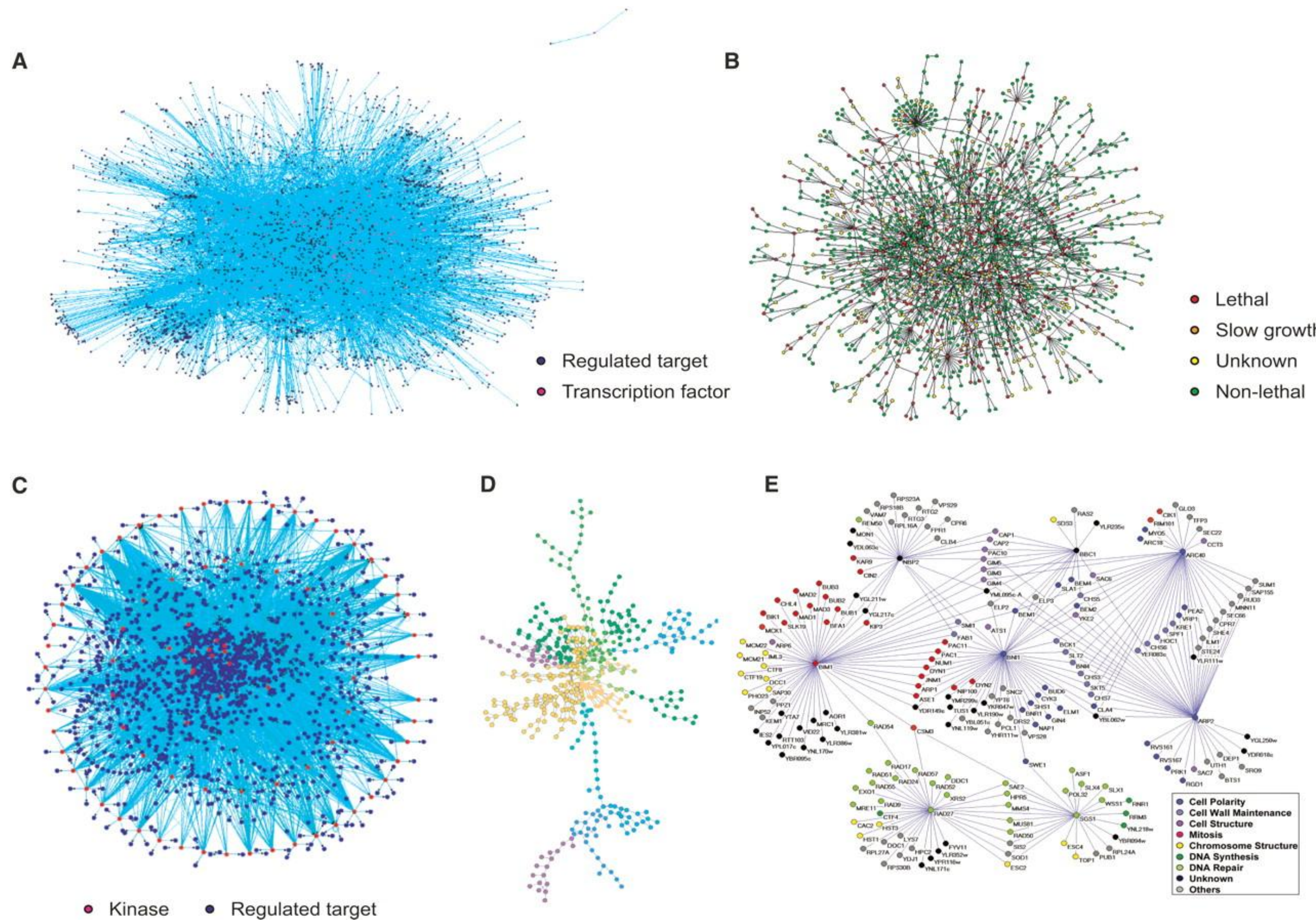
Bacon number 2

Erdös-Bacon number 6

Erdös number 3

Chayes, L., McKellar, D. & Winn, B. (1998)
Percolation and Gibbs states multiplicity for ferromagnetic Ashkin–Teller models on Z^2.
Journal of Physics A: Mathematics and General, 31, 9055–9063.

Figure 1. Examples of the five major biological networks



Xiaowei Zhu et al. Genes Dev. 2007; 21: 1010-1024

Cold Spring Harbor Laboratory Press

Table 1. *Current status of biological networks*

Type of network	Species	Number of nodes	Number of interactions
Transcription factor-binding network	*S. cerevisiae*	3528	7419
		3207	11231
Protein–protein interaction	*C. elegans*	2788	4441
	D. melanogaster	7546	25403
	Homo sapiens	7509	20979
	Mus musculus	209	393
	S. cerevisiae	5325	51773
Phosphorylation network	*S. cerevisiae*	1325	4200
Metabolic network	*E. coli*	473	574
	S. cerevisiae	646	1149
Genetic network	*S. cerevisiae*	3258	13963

[a]Transcriptional factor-binding data collected at rich-media condition.
[b]Transcriptional factor-binding data collected at a variety of growth conditions.
[c]Synthetic lethal interactions among nonessential genes.

Xiaowei Zhu et al. Genes Dev. 2007; 21: 1010-1024

- How do the networks specify organisms?
- What “functions” are computed by the networks?
- What are the principles that govern the structure and function of networks (or subnetworks)?
- How are networks organized?
- How are the specific functions of the cell accomplished?
- What behaviors emerge from these networks?
- How did the networks evolve?

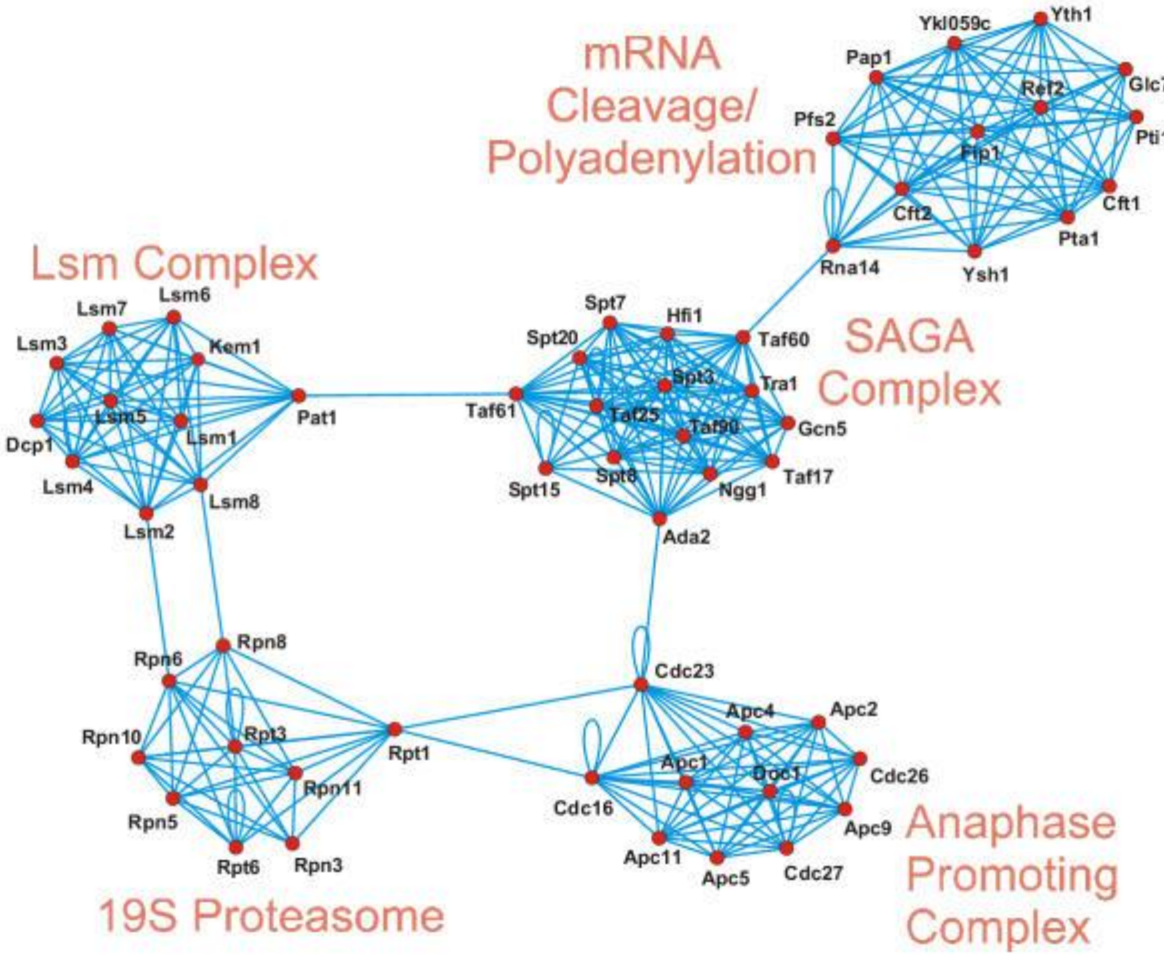
Modules

Biological organization of networks based on functional modules

- Composed of many different types of molecules
- Functions of modules are (more or less) discrete entities and arise as a result of interactions among its components

Hartwell et al 1999

Uncovering modules: clustering



Network clustering

Distance-based hierarchical clustering

- Similarity between two proteins estimated as
  - Function of shared neighbors
  - Shortest-path profiles etc.

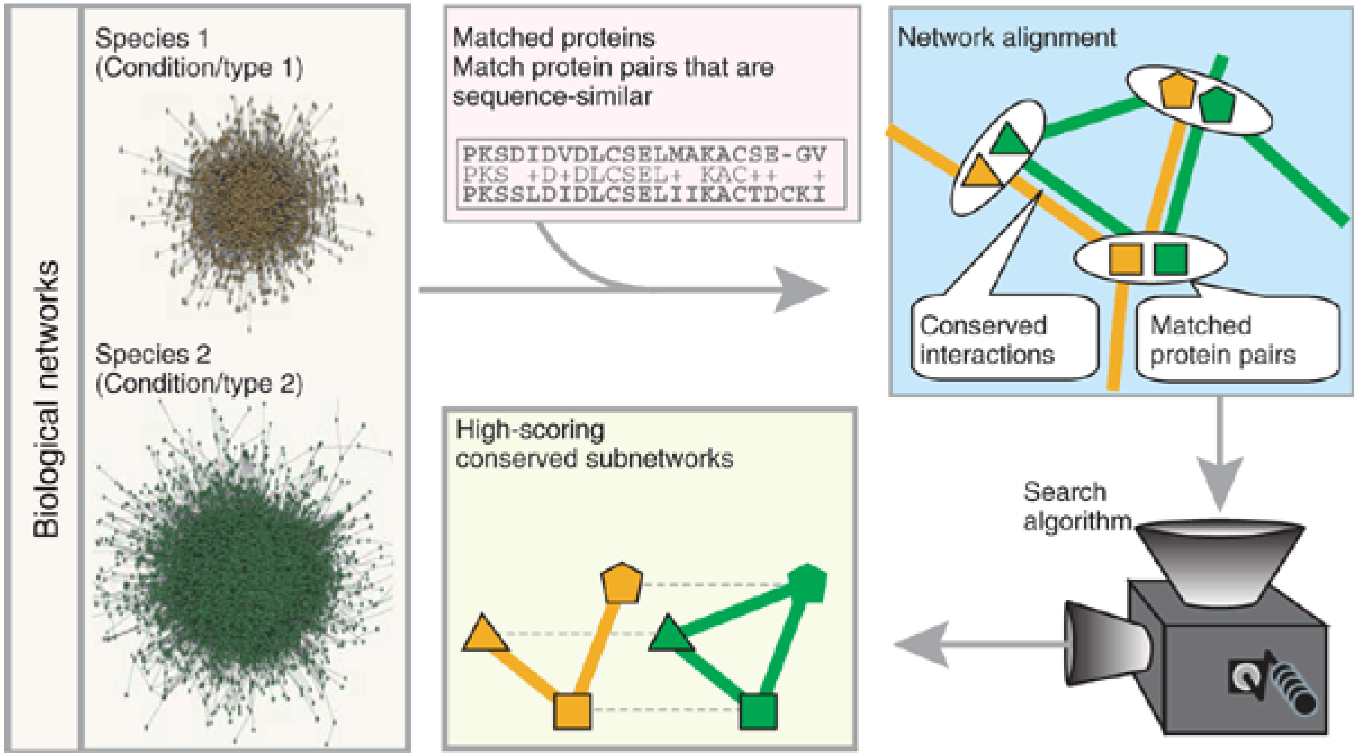
Dense subgraph finding

- Monte-carlo methods
- Greedy expansion
- Building from seed set

Divisive hierarchical approaches

Stochastic-flow…

Modules via Network Alignment

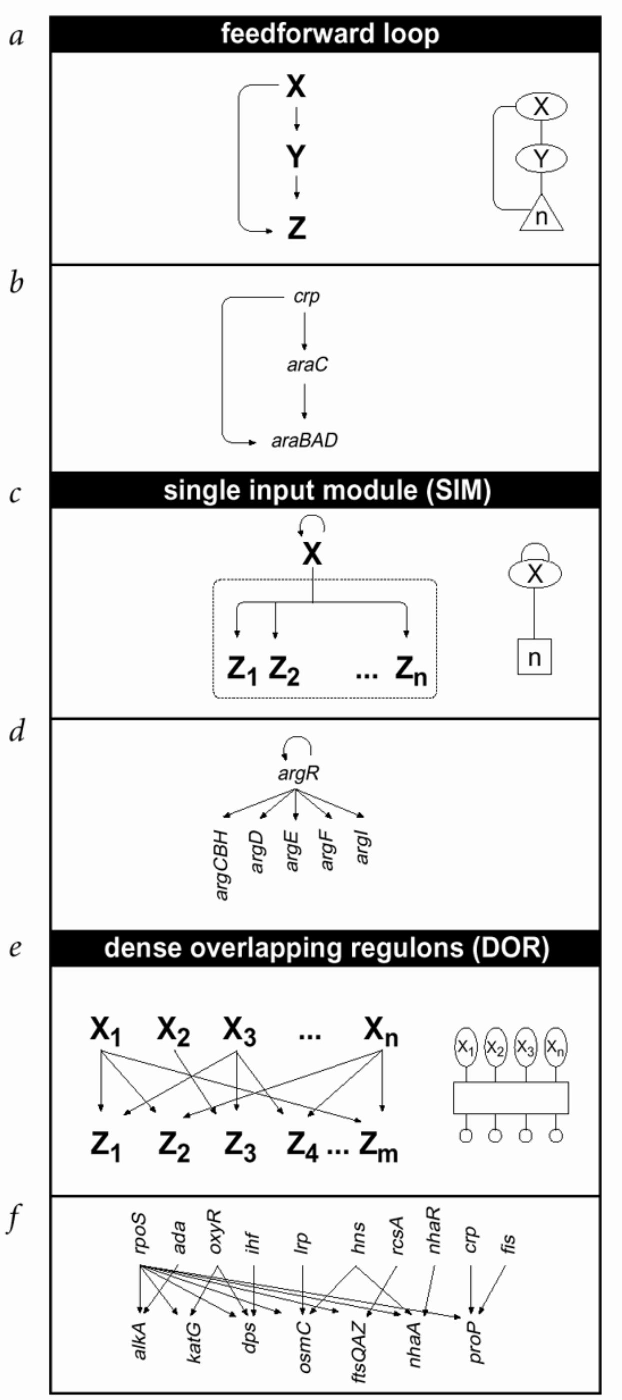


Ideker et al.

Network Motifs

Patterns of interconnections that recur in many different parts of a network at frequencies much higher than those found in randomized networks [Shen-Orr et al 2002]

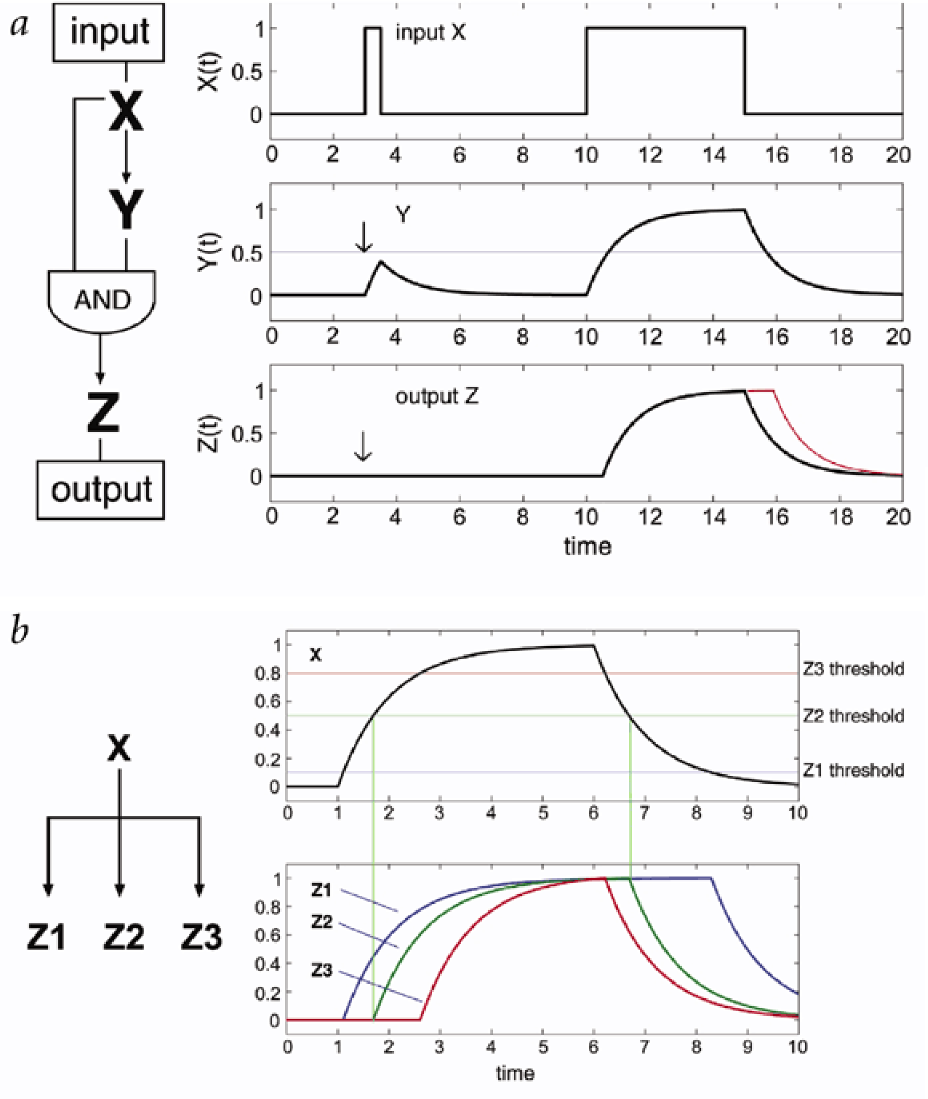
First applied to E. coli transcriptional network



Network Motifs

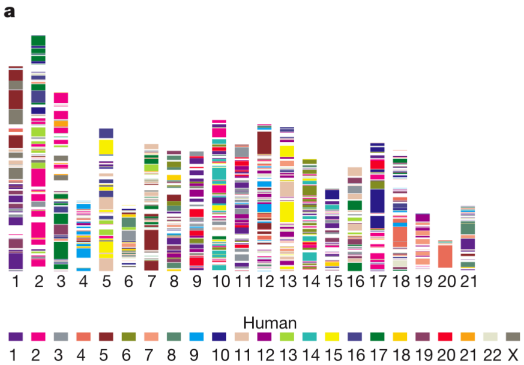
Shen-Orr et al.

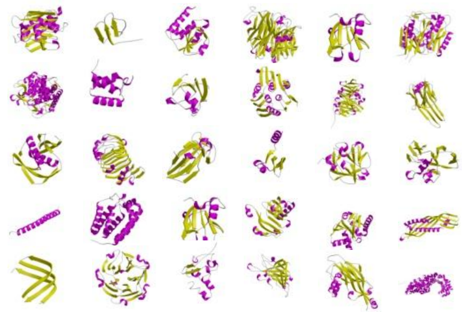
“Functional” properties of motifs

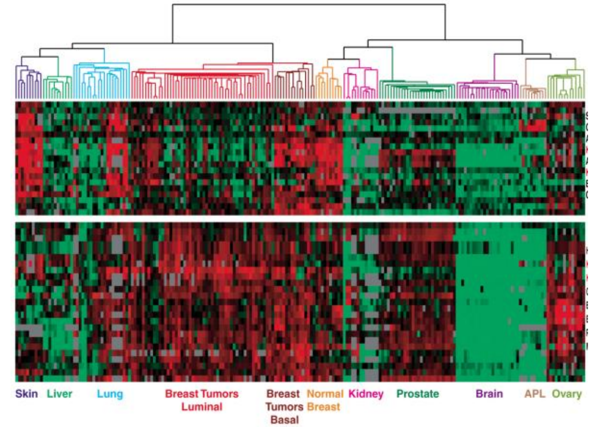


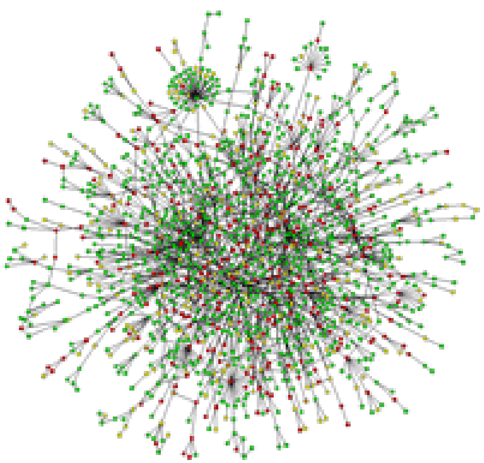
Theory ——— Experiment

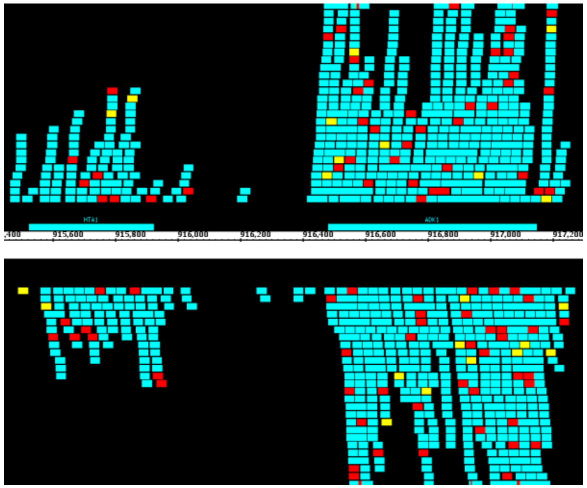
Theory Experiment

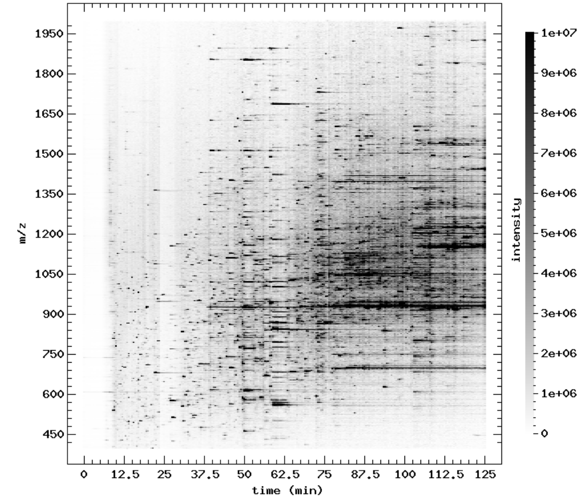

a
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
Human
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 X

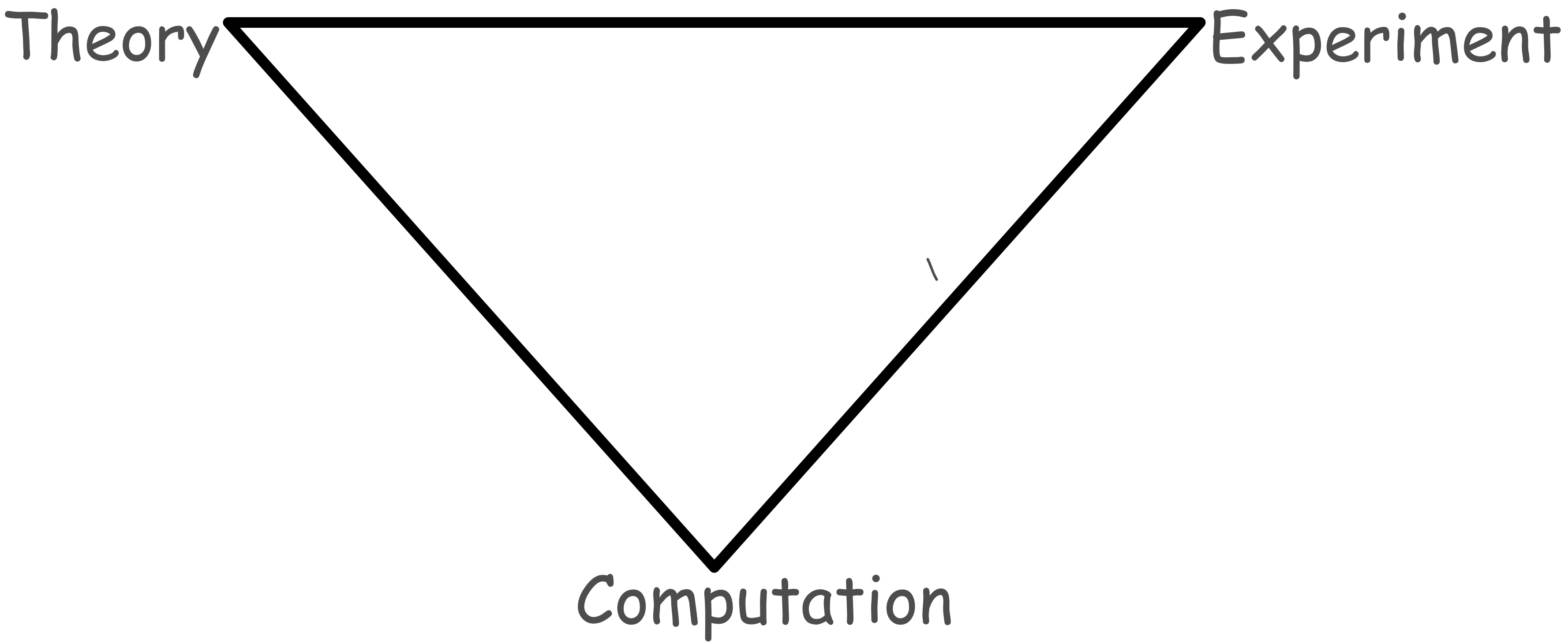



Skin
Liver
Lung
Breast Tumors Luminal
Breast Tumors Basal
Normal Breast
Kidney
Prostate
Brain
APL
Ovary






m/z
1950
1800
1650
1500
1350
1200
1050
900
750
600
450
0 12.5 25 37.5 50 62.5 75 87.5 100 112.5 125
time (min)
intensity
1e+07
9e+06
8e+06
7e+06
6e+06
5e+06
4e+06
3e+06
2e+06
1e+06
0


Theory
Experiment
Computation