Applying High-Dimensional Clustering Methods For Phylogenetic Profiling

Ruchira S. Datta

UC Berkeley

Work in progress, joint with Jonathan Eisen & Amber Hartman, UC Davis

Setting the Stage

- *Premise: Found all genes for many species.*

- *Problem: Don't know what newfound genes do.*

- *Definition: Phylogenetic profile of a gene:*
the set of species in which it (or an ortholog)
occurs.

- *Idea: Genes with similar phylogenetic profiles
may do similar things.*

The Data

For each gene in the genome of Haloferax volcanii:
  Searched for orthologous genes in all other complete genomes

$\Rightarrow$ *Data matrix*
  m Rows: 4209 genes of Haloferax volcanii
  n Columns: 301 species with complete genomes
  mn Entries: 1 if gene i has ortholog in species j, 0 otherwise

Row i is the phylogenetic profile of gene i

```
NAME "ARCH-Pyrobaculum aerophilum str IM2" "ARCH-Pyrococcus furiosus DSM 3638"

ORF00001 ORF00001 conserved hypothetical protein 0 0

ORF00002 ORF00002 methanol dehydrogenase regulatory protein 0 1

ORF00003 ORF00003 mutS DNA mismatch repair protein MutS 0 0
```

Picture the Matrix

Clustering: A Problem in Many Contexts

We have items described by feature vectors.
We want to cluster together items whose feature vectors are similar.

Items	Features
Terms	Documents in which they occur
Landmarks	Images in which they occur
Genes	Genomes in which they occur

Data matrix: rows=items, columns=features

The Converse Problem

In these examples, transposing the matrix also gives a clustering problem.

Items	Features
Documents	Terms which occur in them
Images	Landmarks which occur in them
Genomes	Genes which occur in them

A Popular Clustering Method: k-means

Consider the feature vectors as points in a high-dimensional Euclidean space. Fix a number k of clusters to find.

Initialize centroids of k clusters to k random items.

For each item, assign it to the nearest cluster, and update the centroid of that cluster.

Problem: How to choose k?
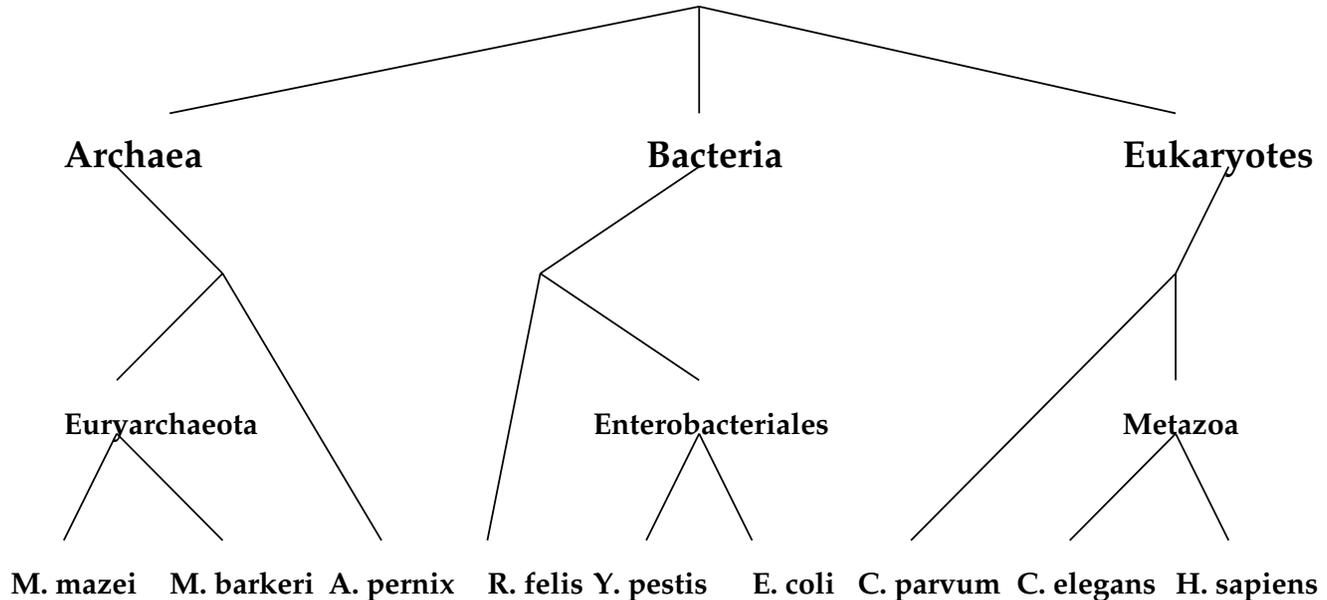
Drawback: Boolean vectors in Euclidean space?

Phylogenetic Trees

- Complete genomes of the species in our data matrix have been sequenced.
- Multiple sequences have been aligned.
- A phylogenetic tree has species at the leaves.
- The root of the smallest subtree containing two species is their most recent common ancestor in evolutionary terms.
- Use sequence similarity to generate

Phylogenetic Tree of the Species

Main Source: National Microbial Pathogen Data Resource
Performed neighbor joining by hand from several other sources

Archaea   Bacteria   Eukaryotes

Euryarchaeota   Enterobacteriales   Metazoa

M. mazei   M. barkeri   A. pernix   R. felis   Y. pestis   E. coli   C. parvum   C. elegans   H. sapiens

Trees Give Clusterings

At the bottom are lots of singleton clusters: the leaves
(M. mazei) (M. barkeri) (A. pernix) (R. felis) (Y. pestis) (E. coli) (C. parvum)
(C. elegans) (H. sapiens)
Rising up one level in the tree, cluster together leaves which are connected at that
level to get many small clusters
(M. mazei, M. barkeri) (A. pernix) (R. felis) (Y. pestis, E. coli) (C. parvum)
(C. elegans, H. sapiens)
Rising up further in the tree, cluster together leaves connected at this higher level to
get fewer, larger clusters
(M. mazei, M. barkeri, A. pernix) (R. felis, Y. pestis, E. coli)
(C. parvum, C. elegans, H. sapiens)

Trees Give Orderings

An ordering of the leaves which doesn't make edges cross is compatible with the tree. It's also compatible with all the clusterings: leaves which are clustered together are close together in the ordering.
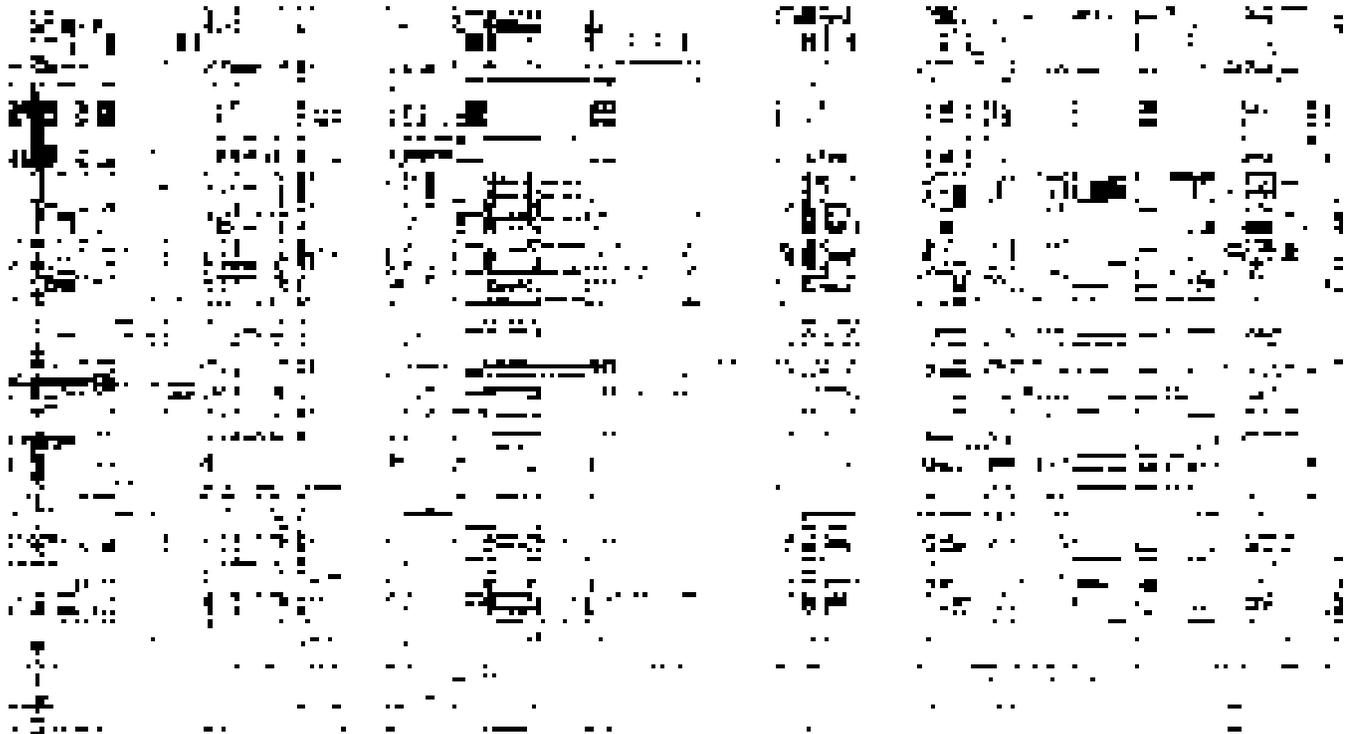
This ordering is compatible:

C. elegans, H. sapiens, C. parvum, A. pernix, M. mazei, M. barkeri, E. coli, Y. pestis, R. felis

This ordering is incompatible:

A. pernix, C. elegans, C. parvum, E. coli, H. sapiens, M. barkeri, M. mazei, R. felis

Picture the Species-Ordered Matrix

Similarity of Sparse Boolean Vectors

We may consider sparse Boolean vectors to be similar if they overlap "a lot".

Definition: The Jaccard measure of two sets is

$$\frac{\text{\# of elements in their intersection}}{\text{\# of elements in their union}}$$

0 if no overlap, 1 if overlap exactly.

The Jaccard dissimilarity between two Boolean vectors is 1 - the Jaccard measure.

An Algebraic Trick

Let M be our data matrix, filled with ones and zeros.
Let N be the intersection matrix:

$$N_{ij} = \text{\# of elements in intersection of row } i \text{ and row } j$$

Let U be the union matrix:

$$U_{ij} = \text{\# of elements in union of row } i \text{ and row } j$$

Then $N = MM^T$.
Let 1_{mn} denote the $m \times n$ matrix all of whose entries are 1.
Since $A \cup B = ((A \cup B)^c)^c = (A^c \cap B^c)^c$,
we have $U = 1_{mm} - (1_{mn} - M)(1_{mn} - M)^T$.

Hierarchical Agglomerative Clustering

- Order all pairs of genes (phylogenetic profiles) by increasing dissimilarity
- Start with a forest of trees, each a single leaf
- Gradually raise the dissimilarity threshold from 0, going through the pairs in order
- If a pair of genes has dissimilarity less than the threshold, connect the two trees containing the genes at the current level

Hierarchical Agglomerative Clustering Example

$j(A, B) = 0.2, \qquad j(B, C) = 0.6, \qquad j(A, C) = 0.7$
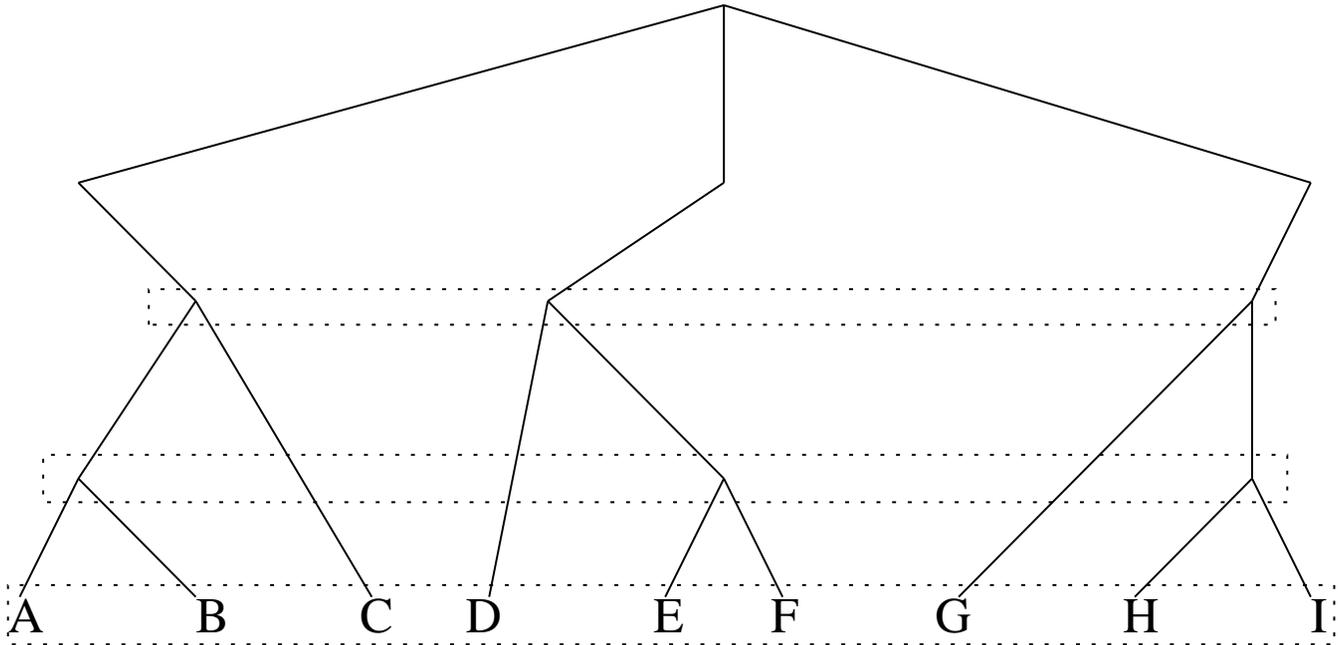
Single Linkage

Here we merge two clusters if the dissimilarity between any two elements in the clusters is below the current threshold (single linkage).

We could require the dissimilarity between the two clusters to be below the current threshold. Would need to extend the definition of Jaccard dissimilarity intelligently: as the number of elements increases, the intersection decreases geometrically whereas the union increases arithmetically.

Picture the Gene-and-Species-Ordered Matrix
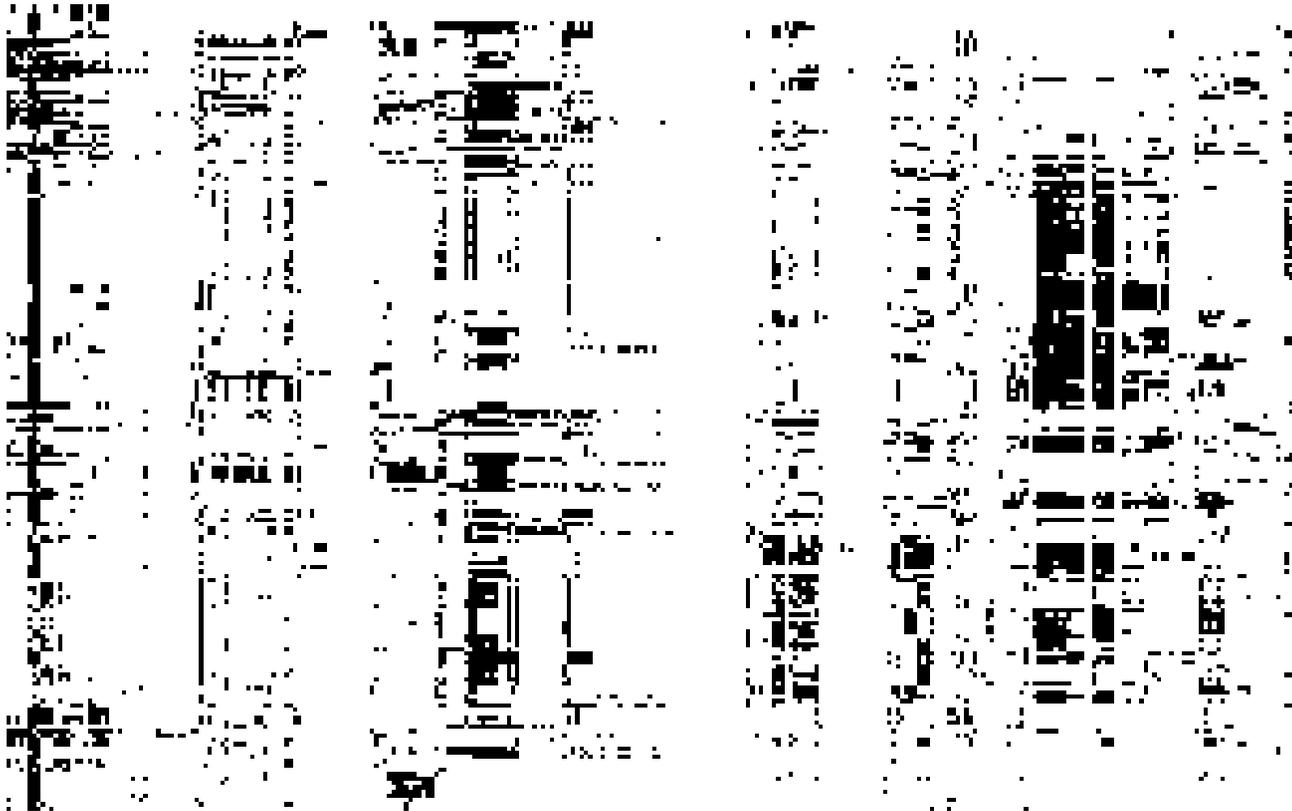
Infer Biological Hypotheses

Consider the dark box in the picture.

Columns: Enterobacteriales (E. coli, Shigella, Salmonella, etc.)

```
2109 3825 ORFB00099 ORFB00099 arcR-6 transcription regulator

2110 2898 ORF02898 ORF02898 arcR transcription regulator

2111 3211 ORFA00138 ORFA00138 transcriptional regulator putative

2112 3768 ORFB00042 ORFB00042 unknown

2113 400 ORF00400 ORF00400 gpdA Anaerobic glycerol-3-phosphate dehydrogenase sub-

unit A(G-3-P dehydrogenase).

2114 3208 ORFA00135 ORFA00135 gpdA glycerol-3-phosphate dehydrogenase

2115 399 ORF00399 ORF00399 GB|AAG20147.1|1 glycerol-3-phosphate dehydrogenase chain

B
```

In the clustering, line 2112 clusters with the lines below it. Maybe the unknown gene is a glycerol-3-phosphate dehydrogenase?

Better Performance Through Simhash

We computed the Jaccard dissimilarity between each of the $O(m^2)$ pairs of genes. To compute the union and intersection of a pair takes $O(n)$ time.

Complexity: $O(m^2 n)$

Hashing: Compute a hash function which takes each high-dimensional feature vector to a low-dimensional hash (possibly just a number). The hash function should be relatively quick to compute.

Similarity Hash: A similarity hash takes feature vectors that are close together to hashes that are close together.

Simhash-based Clustering: Estimate the distance between feature vectors by the distance between their hashes. Cluster together items whose simhashes are within a threshold of each other.

Complexity: $O(m) \times$ time to compute the simhash for each item

Minhash Approximates Jaccard Measure

- Randomly permute the features
- Hash each feature vector to the index of the first feature (in the permuted ordering) for which it is nonzero
- Fix a pair of feature vectors. For each, the column index of the minhash lies in the union of the two feature vectors. So it only matters what the random permutation does to this union.
- The probability that the two feature vectors will have the same minhash is the probability that the permutation brings a column lying in their intersection to the first position in the union, i.e., their Jaccard measure.
- Cluster together items with the same minhash

Complexity: $O(mn)$

Incorporating Phylogenetic Distance

If the columns are in random order, and two rows have different minhashes, then we must put them in different clusters.

Idea: Take advantage of the fact that minhash(gene) is a species, and species are related by the phylogenetic tree.

Let dist(gene A, gene B) = PhylogeneticDistance(minhash(A), minhash(B))

Then we can cluster together genes within a threshold distance.

Another way to do the same thing is to allow only permutations (orderings) that are compatible with the phylogenetic tree, and cluster together genes whose minhashes are close together.

We can also reduce the complexity from $O(mn)$ to $O(mb)$ by projecting the phylogenetic profiles onto b higher-level clades (with a 1 if the gene occurs in any species in the clade).