DNA Classification and Clustering

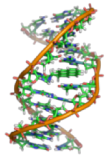
Gail Rosen

Supervised vs. Unsupervised

- Supervised
  - Assumes some training set (in our setting, usually a database of known organisms/genes)
- Unsupervised
  - De Novo
  - Criterion for cutoff

Classification (Supervised)

- Identification (assigning a label) of a particular object to its correct category based on the (features of) data collected from that object.
  - Classify DNA into **previously fixed set** of DNA sequences



3mer database vector		635 Genomes →								
		1	2	3	4	5	6	7	8	9
↓ 4^3 words	AAA	404678	46709	130727	23717	56987	47535	322726	267241	183408
	AAC	268730	69466	160299	47906	86451	86088	153043	123254	88580
	AAG	301373	71200	176900	40775	104539	105499	167045	128796	87628
	AAT	361902	51123	120176	32122	43381	42325	242445	214811	146447
	ACA	204690	54852	115124	37242	82858	77402	124053	115709	56475
	ACC	248053	124371	166366	101044	153358	172183	108899	96032	66789
	ACG	146356	125583	215166	116174	162055	194818	78039	72974	68208

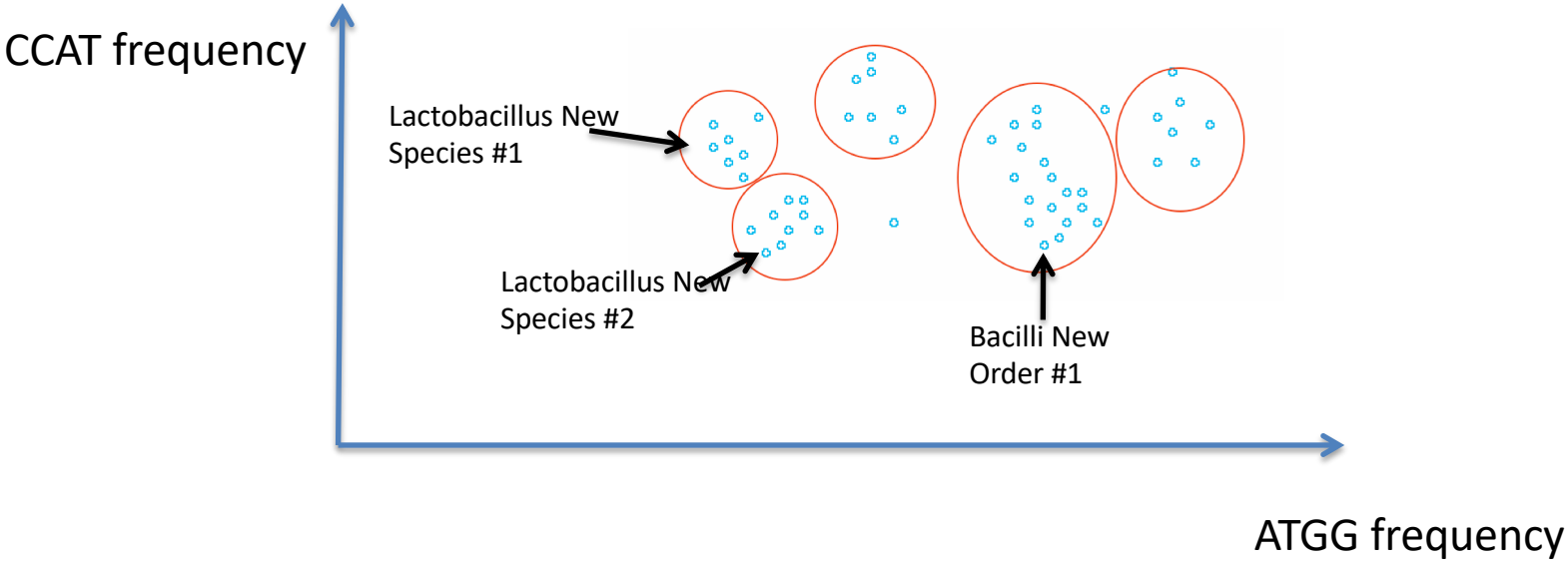
E. Coli!

Supervised Classifiers

- Bayesian (Naïve Bayes, Bayesian Networks)
- k-Neighest Neighbors
- Discriminant Classifiers (Linear/Quadratic)
- Tree (Simple Classification and Regression Tree (CART), Random Forests)
- Functions (Logistic regression, multilayer perceptron, Support Vector Machines)
- Neural Networks (Multilayer Perceptron, Probabilistic, etc).

Clustering (unsupervised)

- Given data of objects obtained from an unknown number and nature of categories, group such data into clusters based on some measure of similarity
  - Data Mining: Given large volumes of data obtained from metagenomes, group the corresponding datainto logically meaningful sets (e.g. organisms/genes/protein families/metabolic pathways)



Unsupervised Clusterers

- K-means
- Self-organizing Maps
- Hierarchical Clustering
- Adaptive Resonance Theory

Terminology

Supervised learning: Given ***training data*** with previously labeled classes, learn the mapping between the data and their correct classes.

- Associated with "classification," typically involves adaptively changing the parameters of a model (classifier) until the model output fits the data

Unsupervised learning: Given ***unlabeled data*** obtained from unknown number of categories, learn how to group such data into meaningful clusters based on some measure of similarity

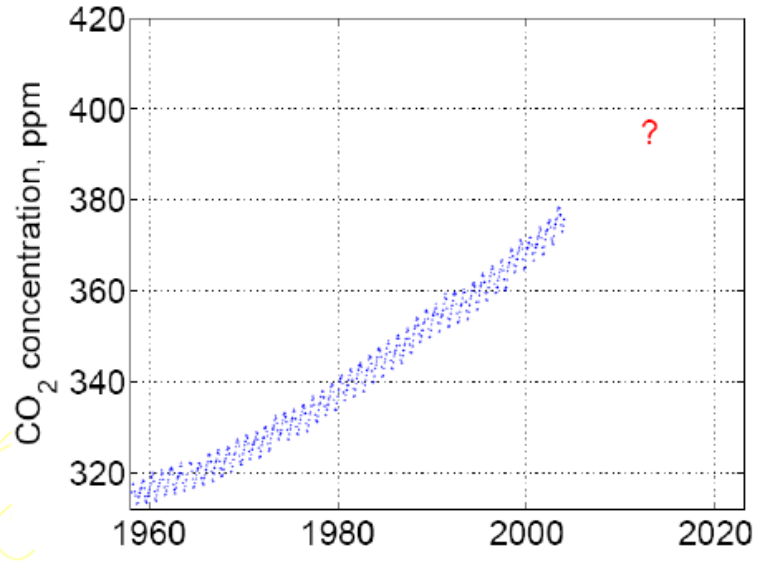
- Typically associated with "clustering" and "density estimation"

Reinforcement learning: Given a sequence of outputs, learn a policy to obtain the desired output. Typically associated with credit assignment and game playing problems

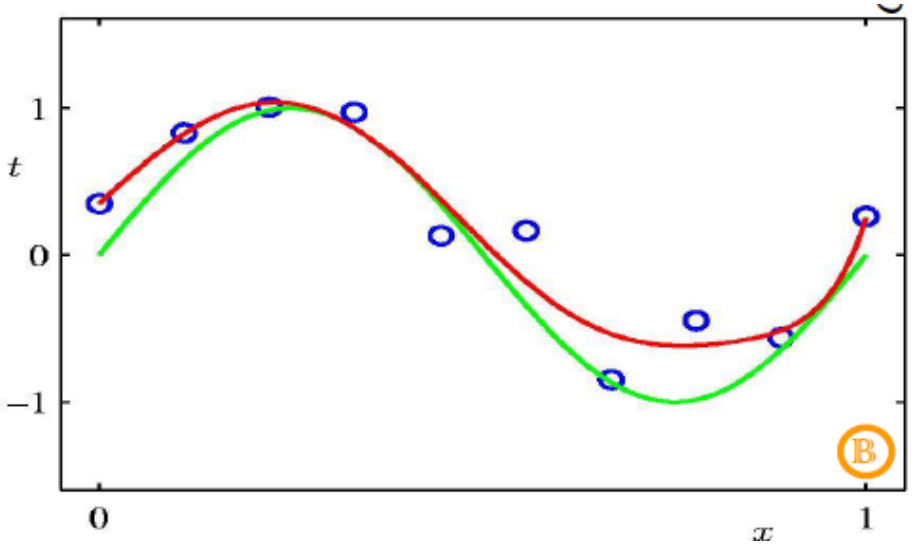
- Learn how to play chess – Given only the rules of the game (how different pieces can move) and the final outcome of the game (you won or you lost), learn the objective and strategies of playing chess.
- No single good move - game is won, if the sequence of moves are collectively good!

Terminology

Prediction: Given historical data obtained from previous behavior of a system / object, predict the future behavior of the same object.



Regression: Given data obtained from an object / system at discrete time points, predict (estimate) the behavior at other (unobserved) time points. Regression is used to determine unknown functions, $t=f(x)$ from its samples (system identification).



Terminology

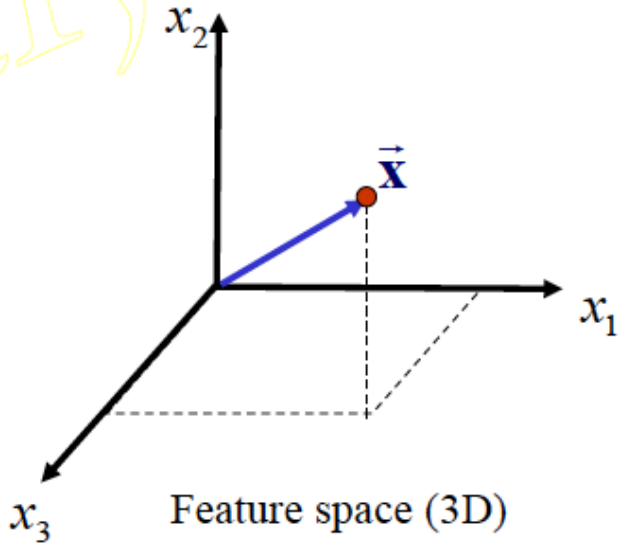
Feature: a variable (predictor) believed to carry discriminating and characterizing information about the objects under consideration

Feature vector: A collection of *d* features, ordered in some meaningful way into a *d-dimensional* column vector, that represents the signature of the object to be identified.

Feature space: The *d-dimensional* space in which the feature vectors lie. A *d-dimensional* vector in a *d-dimensional* space constitutes a point in that space.

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix} \begin{matrix} \textit{feature 1} \\ \textit{feature 2} \\ \\ \textit{feature d} \end{matrix}$$

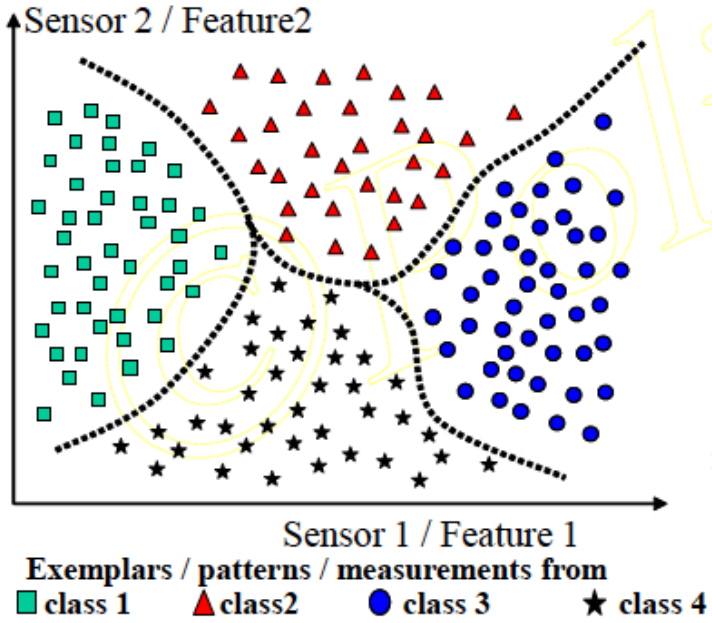
$$\mathbf{x} = \begin{bmatrix} 56 \\ 80 \\ 120 \\ 220 \end{bmatrix} \begin{matrix} \leftarrow \text{Age (years)} \\ \leftarrow \text{Diastolic BP (mmHg)} \\ \leftarrow \text{Systolic BP (mmHg)} \\ \leftarrow \text{Total lipids (ml/dl)} \end{matrix}$$



Feature space (3D)

Terminology

- Class: The category to which a given object belongs, typically denoted by ω
- Pattern: A collection of features of an object under consideration, along with the correct class information of that object. In classification, a pattern is a pair of variables, $\{\vec{\mathbf{x}}, \omega\}$ where $\vec{\mathbf{x}}$ is the feature vector and ω is the corresponding label
- Instance/ Exemplar: Any given example pattern of an object
- Decision boundary: A boundary in the *d-dimensional* feature space that separates patterns of different classes from each other.



- Training Data: Data used during training of a classifier for which the correct labels are *a priori* known
- Test / Validation Data: Data not used during training, but rather set aside to estimate the true (generalization) performance of a classifier, for which correct labels are also *a priori* known
- Field Test Data: Unknown data to be classified for which the classifier is ultimately trained. The correct class labels for these data are not known *a priori.*

Terminology

- Cost Function: A quantitative measure that represents the cost of making an error. The classifier is trained to minimize this function.
- Classifier: A parametric or nonparametric model which adjusts its parameters or weights to find the correct decision boundaries through a learning algorithm using a training dataset – such that a cost function is minimized.
- Model: A simplified mathematical / statistical construct that mimics (acts like) the underlying physical phenomenon that generated the original data.
- Parametric Model: A probabilistic / statistical model that assumes that the underlying phenomenon follows a specific known probability distribution. The parameters of such a model are the parameters of the distribution.
  - A classifier based on determining the parameters of a distribution is also called a ***generative model*** as the underlying distribution can be generated from the parameters.
  - Examples: Bayes classifier, expectation-maximization algorithm.
- Nonparametric model: A model that does not assume a specific distribution, and that typically follows an optimization algorithm to minimize error.
  - A classifier based on using a nonparametric approach is also called a ***discriminative model***, as the decision is then based on a ***discriminant*** (or discriminant function).
  - Examples: Neural networks, decision trees, support vector machines.

Terminology

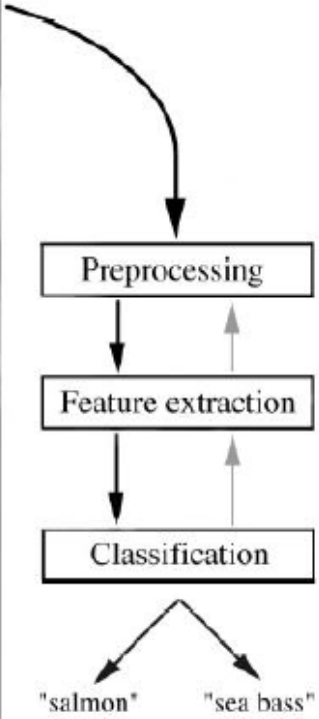
- Error: Incorrect labeling of the data by the classifier
- Cost of error: Cost of making a decision, in particular an incorrect one – not all errors are equally costly!
- Training Performance: The ability / performance of the classifier in correctly identifying the classes of the training data, which it has already seen. It may not be a good indicator of the generalization performance.
- Generalization (Test Performance): The ability / performance of the classifier in identifying the classes of previously unseen patterns.
- Confusion Matrix: The matrix obtained from test performance of the classifier that shows how many instances of each class are classified into different classes.

$$\mathbf{CM} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1K} \\ c_{21} & c_{22} & \cdots & c_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ c_{K1} & c_{K2} & \cdots & c_{KK} \end{bmatrix}$$

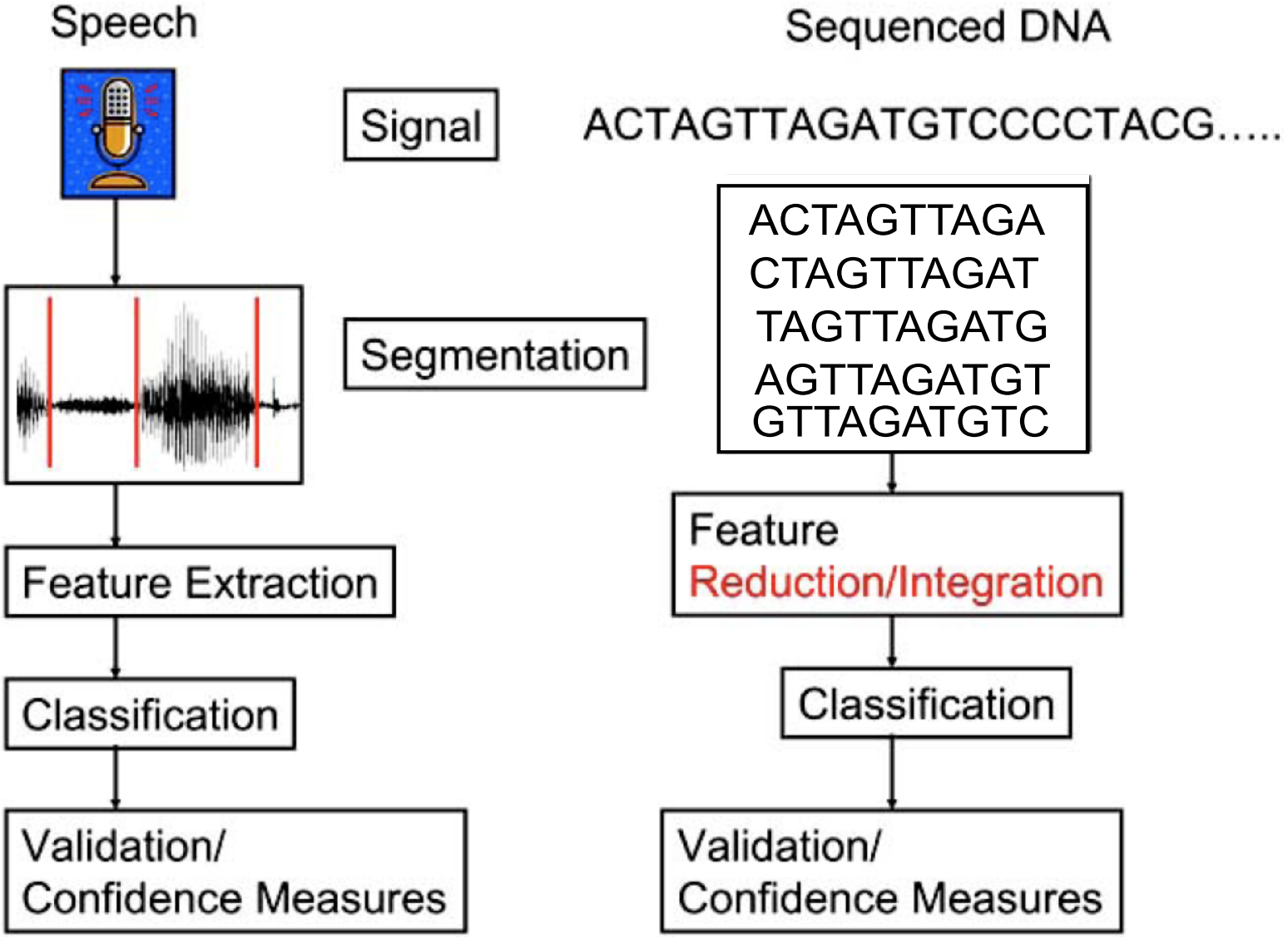
Number of correctly classified instances

c_{ij}: Number of class ω_i instances classified as class ω_j by the classifier.

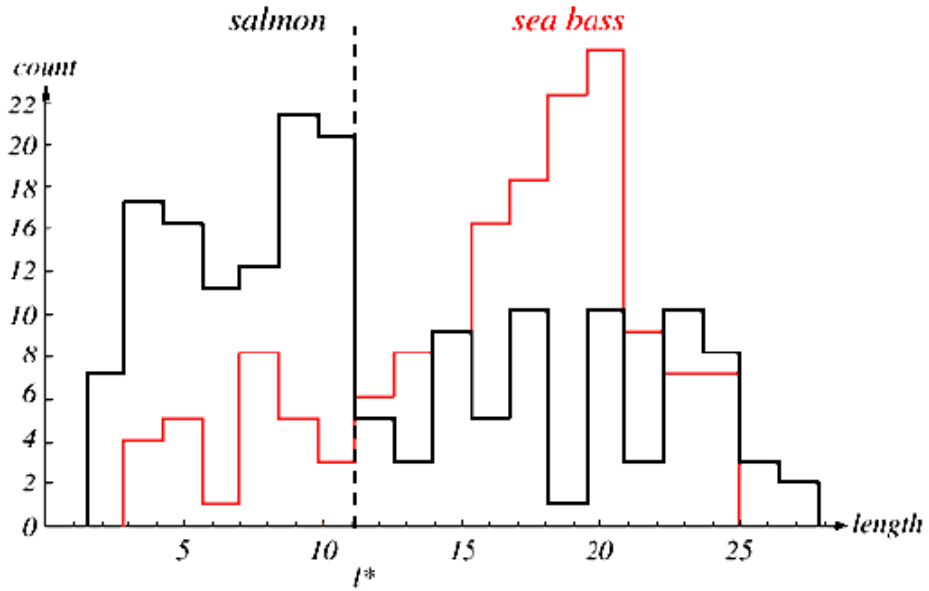
- Salmon or Sea Bass?



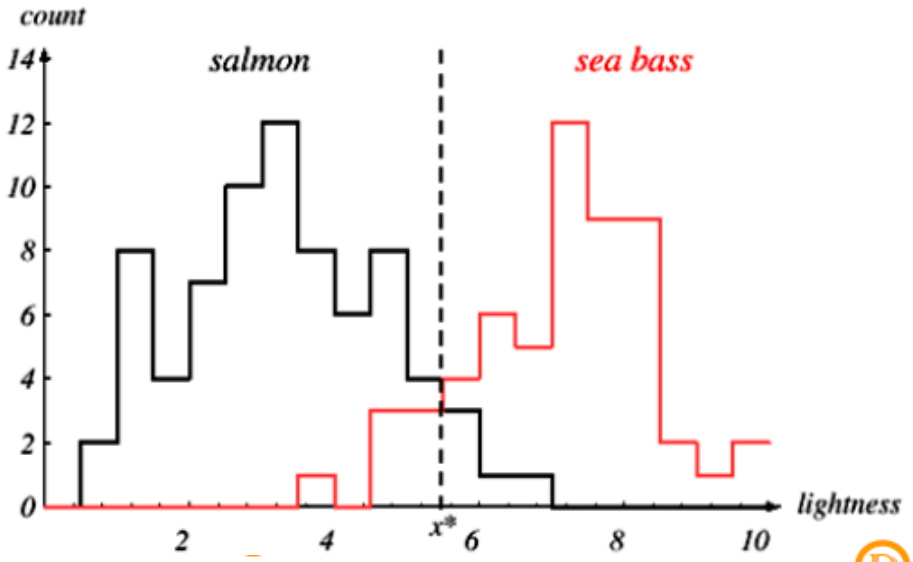
Analogue of speech recognition to Nmer-Based feature Metagenomics



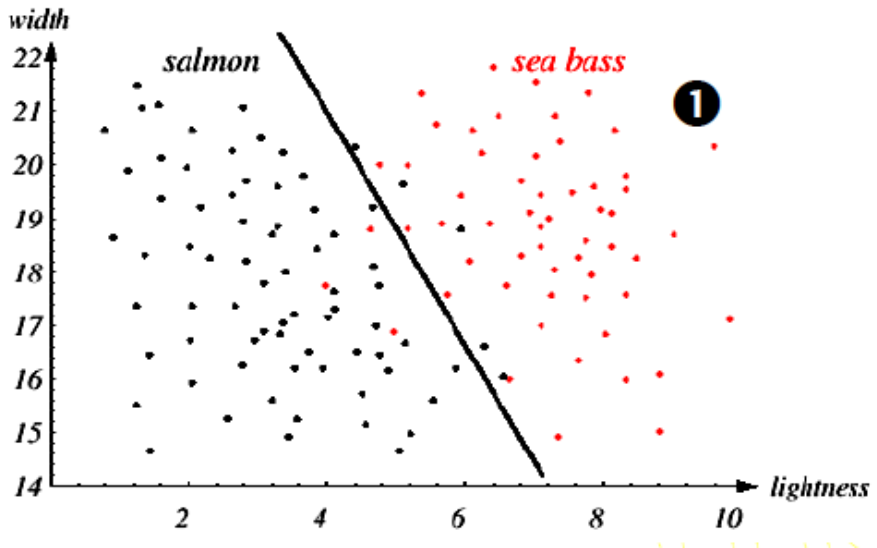
Features

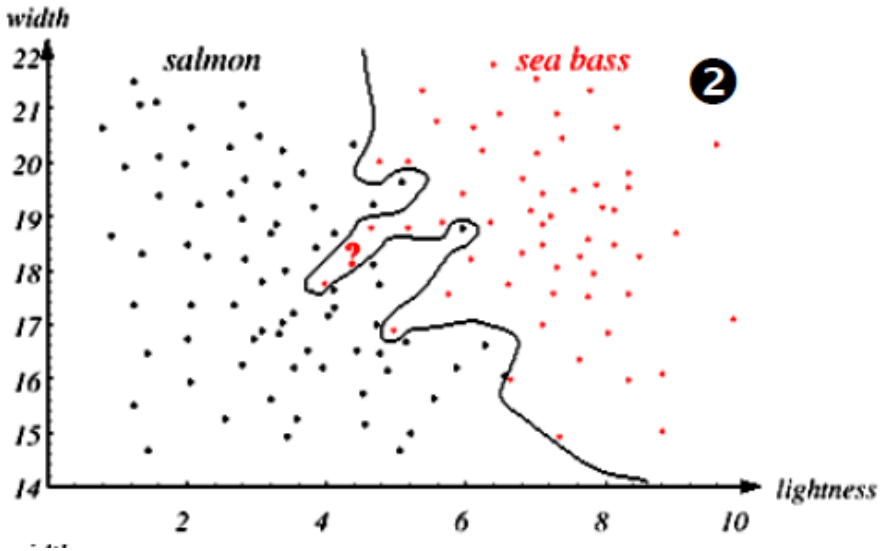


- Length or Lightness (weight), which one is a better feature?
- If you were to make the decision based on the value of a single feature, which one would you choose and what would the decision value be?
- No value of either feature will "classify" all fish correctly… What to do?



Decision Boundary



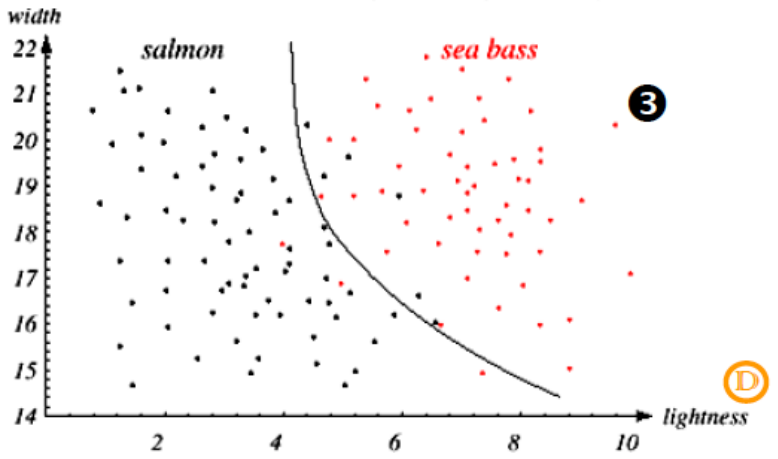


➲ Which of the boundaries would you choose?

❶ **Simple linear boundary – training error > 0**

❷ **Nonlinear complex boundary – tr. error = 0**

❸ **Simpler nonlinear boundary – tr. error > 0**



Probability Theory

Here are the most important things to know in probability

- Probabilities are nonnegative and normalize to 1

$$P(x) \geq 0, \sum_x P(x) = 1 \text{ (disc.)}$$
$$\int P(x)dx = 1 \text{ (cont.)}$$

- If you have two r.v. X and Y, they have a joint distribution $P(X,Y)$

$$P(X,Y) = P(Y,X), \quad 0 \leq P(X = x_i, Y = y_j) \leq 1, \quad \sum_i \sum_j P(X = x_i, Y = y_j) = 1, \text{ or } \iint P(x,y)dydx = 1$$

- If X and Y are independent (and only then) ➔ $P(X,Y) = P(X)P(Y)$
- **The sum rule:** The marginal probability of a single r.v. can always be obtained by summing (integrating) the pdf over all values of all other variables, for example

$$P(x) = \int P(x,y)dy \quad P(y) = \sum_i P(X = x_i, Y)$$
$$P(X) = \iint P(x,y,z)dydz \quad P(A,C) = \sum_B \sum_D \sum_E P(A,B,C,D,E)$$

- **The product rule**: The joint probability can always be obtained by multiplying the conditional probability (conditioned on one of the variables) with the marginal probability of the conditioned variable:

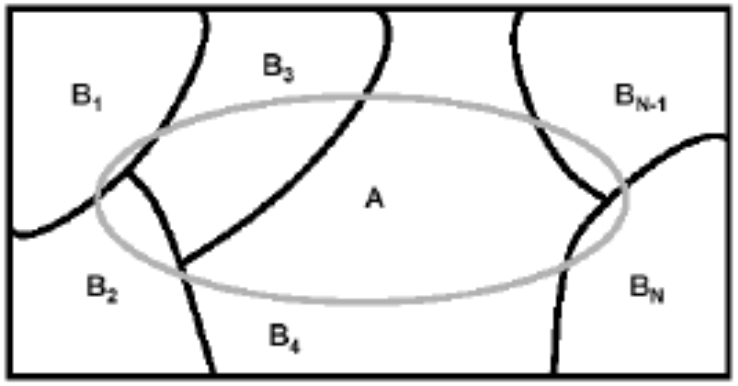
$$P(X,Y) = P(Y|X)P(X) = P(X|Y)P(Y)$$

- which gives rise to **Bayes rule**:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} = \frac{P(X|Y)P(Y)}{\sum_Y P(X,Y)} \propto P(X|Y)P(Y)$$

Bayes Rule

- We pose the following question: Given that the event A has occurred. What is the probability that any single one of the event B's occur?



$$P(B_j \mid A) = \frac{P(A \cap B_j)}{P(A)} = \frac{P(A \mid B_j) \cdot P(B_j)}{\sum_{k=1}^{N} P(A \mid B_k) \cdot P(B_k)}$$

Rev. Thomas Bayes,
(1702-1761)

This is known as the Bayes rule

$$posterior = \frac{likelihood \times prior}{evidence}$$

Bayes Classifier

Statistically, the best classifier you can build !!!

Based on quantifying the trade offs betweens various classification decisions using a probabilistic approach

The theory assumes:

- Decision problem can be posed in probabilistic terms
- All relevant probability values are known or can be estimated (in practice this is not true)

Back to our fish example:

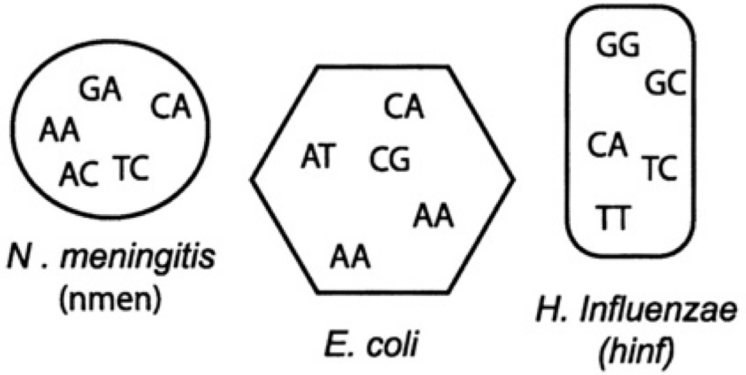
- Assume that we know the probabilities of observing sea bass and salmons, $P(\omega_1)$ and $P(\omega_2)$, for a particular location of fishing and time of year
  - *Prior probability*
- Based on this information, how would you guess the type of the next fish to be caught?

$$\omega = \omega_1 \quad if \quad P(\omega_1) > P(\omega_2)$$
$$\omega = \omega_2 \quad if \quad P(\omega_2) > P(\omega_1)$$

A reasonable decision rule ?

Naïve Bayes Classifier on Metagenomes

(A) Genomes:



(B) Motif Occurrence Profile:

Motif	nmen	ecoli	hinf
AA	1	2	0
AT	0	1	0
AC	1	0	0
.....			
GG	0	0	1
Sum	N_n	N_e	N_h

(C) Motif Frequency Profile:

Motif	nmen	ecoli	hinf
AA	$1/N_n$	$2/N_e$	$0/N_h$
AT	$0/N_n$	$1/N_e$	$0/N_h$
AC	$1/N_n$	$0/N_e$	$0/N_h$
.....			
GG	$0/N_n$	$0/N_e$	$1/N_h$
Sum	1	1	1

(D) Any genomic sequence, S, with j overlapping motifs:

(E) Probability of obtaining the motif distribution S in the different genomes:

Genome	$Motif_1$	....	$Motif_j$	P(S:G)
nmen	$P(M_1:G_n)$		$P(M_j:G_n)$	$\Pi P(M:G_n)$
ecoli	$P(M_1:G_e)$		$P(M_j:G_e)$	$\Pi P(M:G_e)$
hinf	$P(M_1:G_h)$		$P(M_j:G_h)$	$\Pi P(M:G_h)$

(F)

Prediction = max P(S:G)

Algorithm Assumptions:

Motifs are independent of each other.
(Obviously not true if overlapping)

Sandberg, 2001.

Maximize Posterior Probabilities

$$\underset{i}{\text{argmax}}\, P(G_i|S) = \frac{P(S|G_i) \cdot P(G_i)}{P(S)}$$

Probability of Genome i given a Fragment, $P(G_i|S)$
Probability of a Fragment given Genome i, $P(S|G_i)$

Assumptions:

- We don't know the $P(G_i)$ in our mixture so we assume it's EQUI-PROBABLE
- We assume that getting a particular fragment in our sample, P(S), is completely EQUI-PROBABLE (probably not the case)

Maximize Apriori Scores

$$\underset{i}{\text{argmax}}\ P(S|G_i) = \prod_{j=1:N-(m-1)} P(M_j|G_i)$$

M_j is the motif (jth Nmer of the fragment)

N = 6

$P(S|G_i) = P(M_1=\text{ATGTAC}|G_i) \cdot P(M_2=\text{TGTACA}|G_i) \cdot P(M_3=\text{GTACAC}|G_i) \cdots$

Estimate $Pr(M_j|G_i) = \dfrac{\text{Frequency of } M_j}{\text{Total M in } G_i}$

$P(M_j|G_i)$: Probability of Nmer given Genome

N = 3 example

3 mers		635 Genomes →								
		1	2	3	4	5	6	7	8	9
4^3 words ↓	AAA	404678	46709	130727	23717	56987	47535	322726	267241	183408
	AAC	268730	69466	160299	47906	86451	86088	153043	123254	88580
	AAG	301373	71200	176900	40775	104539	105499	167045	128796	87628
	AAT	361902	51123	120176	32122	43381	42325	242445	214811	146447
	ACA	204690	54852	115124	37242	82858	77402	124053	115709	56475
	ACC	248053	124371	166366	101044	153358	172183	108899	96032	66789
	ACG	146356	125583	215166	116174	162055	194818	78039	72974	68208

Sum of 5th column is Total words(M) in G_i

$$Pr(M_j|G_i) = \frac{\text{Frequency of } M_j}{\text{Total M in } G_i}$$

Frequency of ACG in 5th Genome

We get a score for each Genome, i

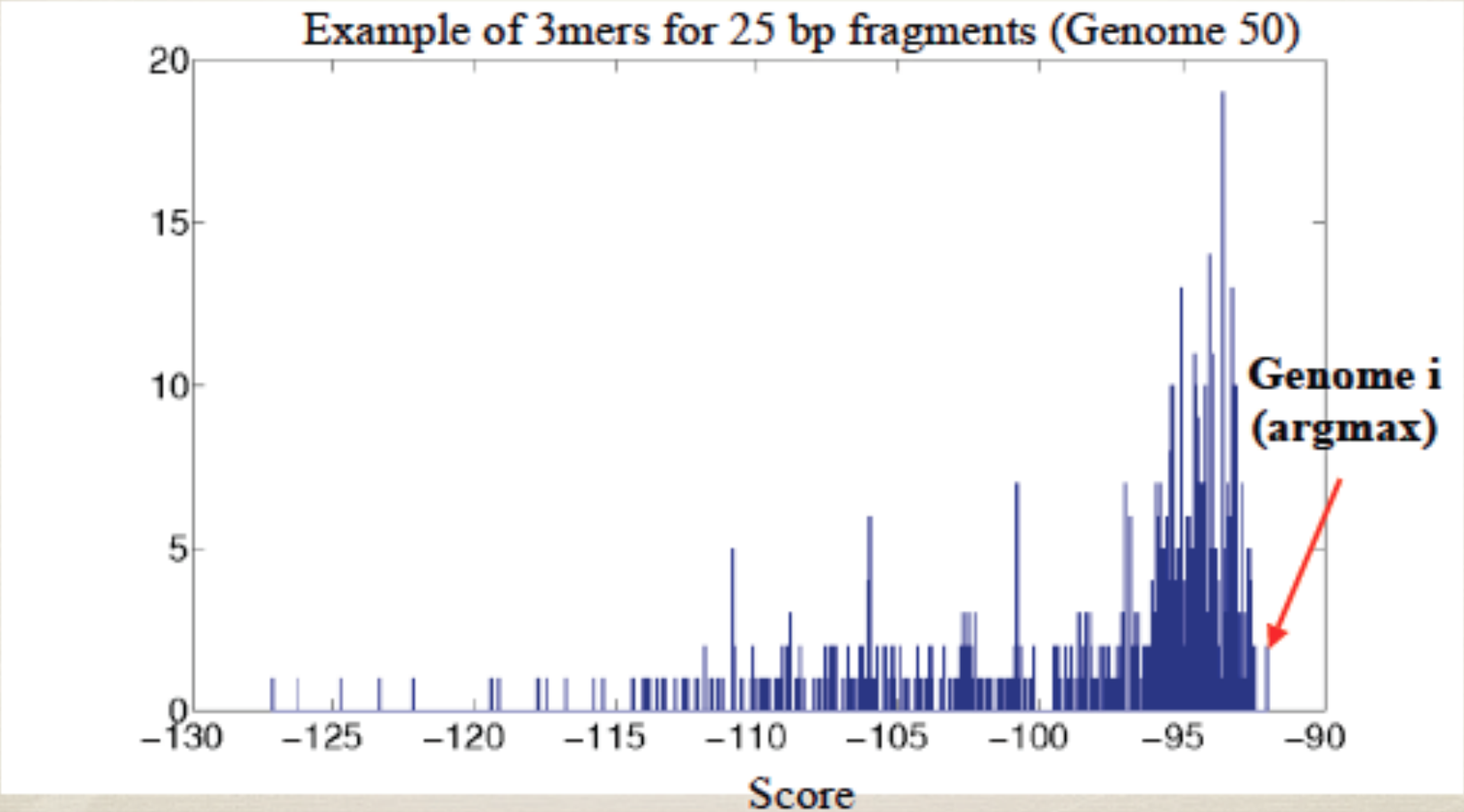
$$\underset{i}{\text{argmax}}\, P(S|G_i) = \prod_{i=1:N-(m-1)} P(M_j|G_i)$$

- The Max($P(S|G_i)$) (get the ith genome which maximizes the probability of the fragment i)

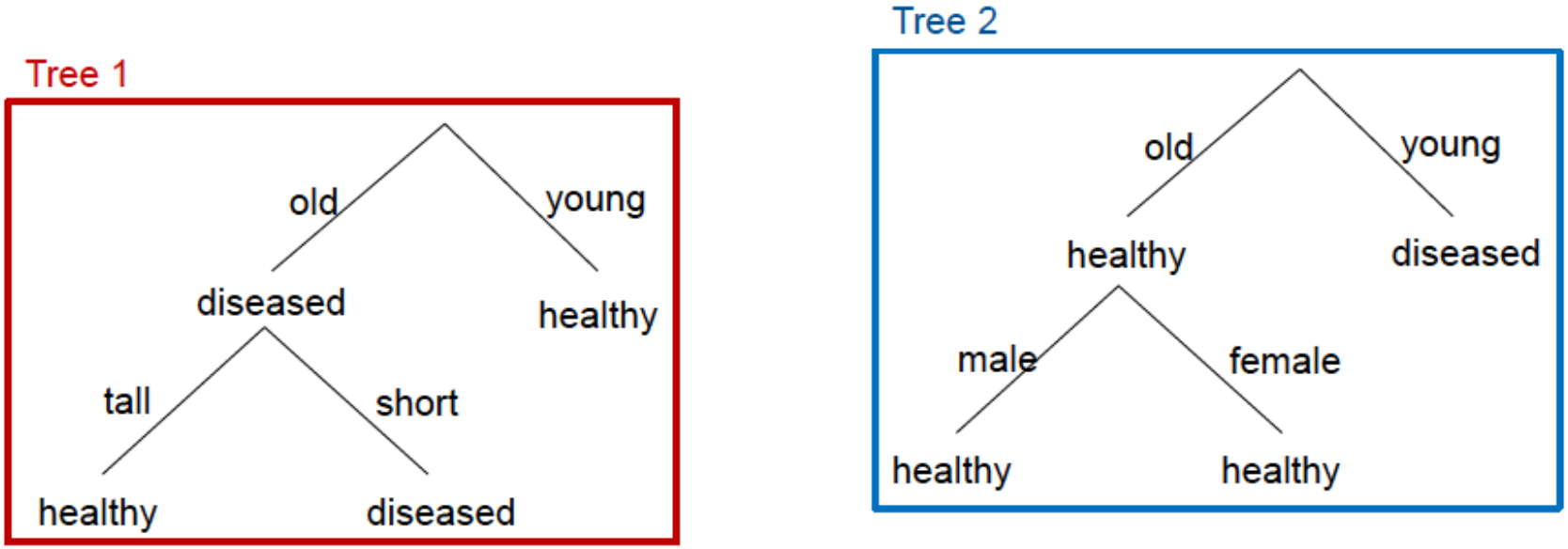
- Take log to reduced numerical error

"Predicted Genome"

NB Scoring - (a lot of close organisms for 3mers)



Random Forests



New sample:

old, retired, male, short

Tree predictions:

diseased, healthy, diseased

Majority rule:

diseased

Random Forest

- Train each tree on bootstrap resample of data
  (Bootstrap resample of data set with N samples:
  Make new data set by drawing **with replacement** N samples; i.e., some samples will probably occur multiple times in new data set)
- For each split, consider only m randomly selected variables
- Don't prune
- Fit B trees in such a way and use average or majority voting to aggregate results

If trees are sufficiently deep, they have very small bias

Traditional Query model

- One Query
- Result

How about estimate abundances of whole sample?

Quikr


David Koslicki Simon Foucart

A= (Database k-mer training matrix, k=3)

3mer database vector		635 Genomes →								
		1	2	3	4	5	6	7	8	9
↓ 4^3 words	AAA	404678	46709	130727	23717	56987	47535	322726	267241	183408
	AAC	268730	69466	160299	47906	86451	86088	153043	123254	88580
	AAG	301373	71200	176900	40775	104539	105499	167045	128796	87628
	AAT	361902	51123	120176	32122	43381	42325	242445	214811	146447
	ACA	204690	54852	115124	37242	82858	77402	124053	115709	56475
	ACC	248053	124371	166366	101044	153358	172183	108899	96032	66789
	ACG	146356	125583	215166	116174	162055	194818	78039	72974	68208

(Sample k-mer vector)

s=

322726
153043
167045
146447
56475
66789
68208

(Proportions/ weights)

x=

0.3
0.2
0.0001
0.1
0.05
…

Wish to solve:

$$A^{(k)}x = s^{(k)}$$

Subject to:

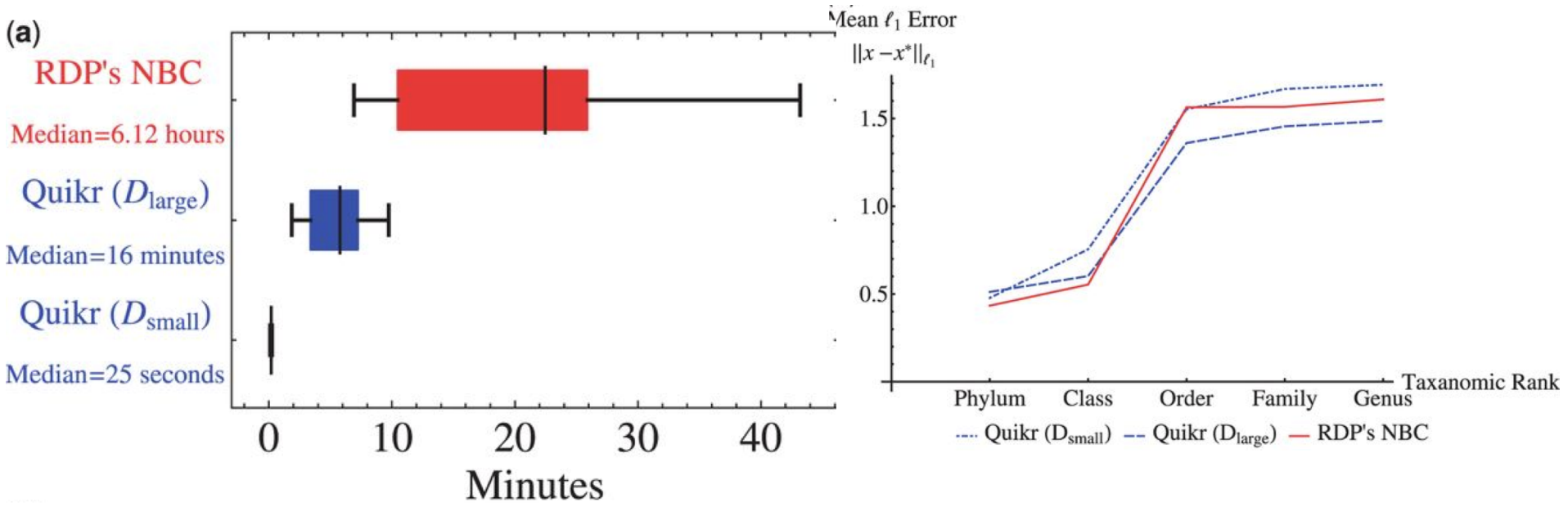
$$x_j \geq 0 \quad \sum_{j=1}^{M} x_j = 1$$

D. Koslicki, S. Foucart, G. Rosen. "Quikr: a Method for Rapid Reconstruction of Bacterial Communities via Compressive Sensing," Bioinformatics, 2013.

Non-negative Least Squares

$$\underset{z \in \mathbb{R}^M}{\text{minimize}} \; ||z||_1^2 + \lambda^2 ||A^{(k)} z - s^{(k)}||_2^2$$

Fast Lawson-Hanson Algorithm



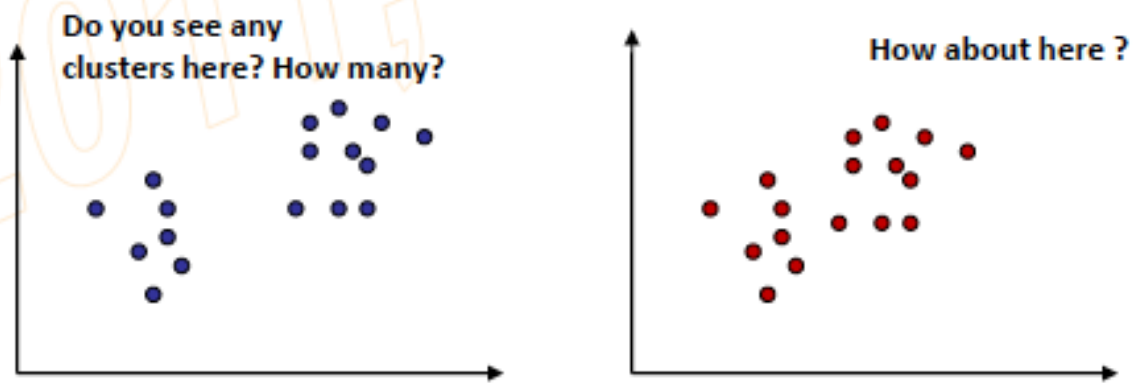
D. Koslicki, S. Foucart, G. Rosen. "Quikr: a Method for Rapid Reconstruction of Bacterial Communities via Compressive Sensing," Bioinformatics, 2013.

Other novel new ways

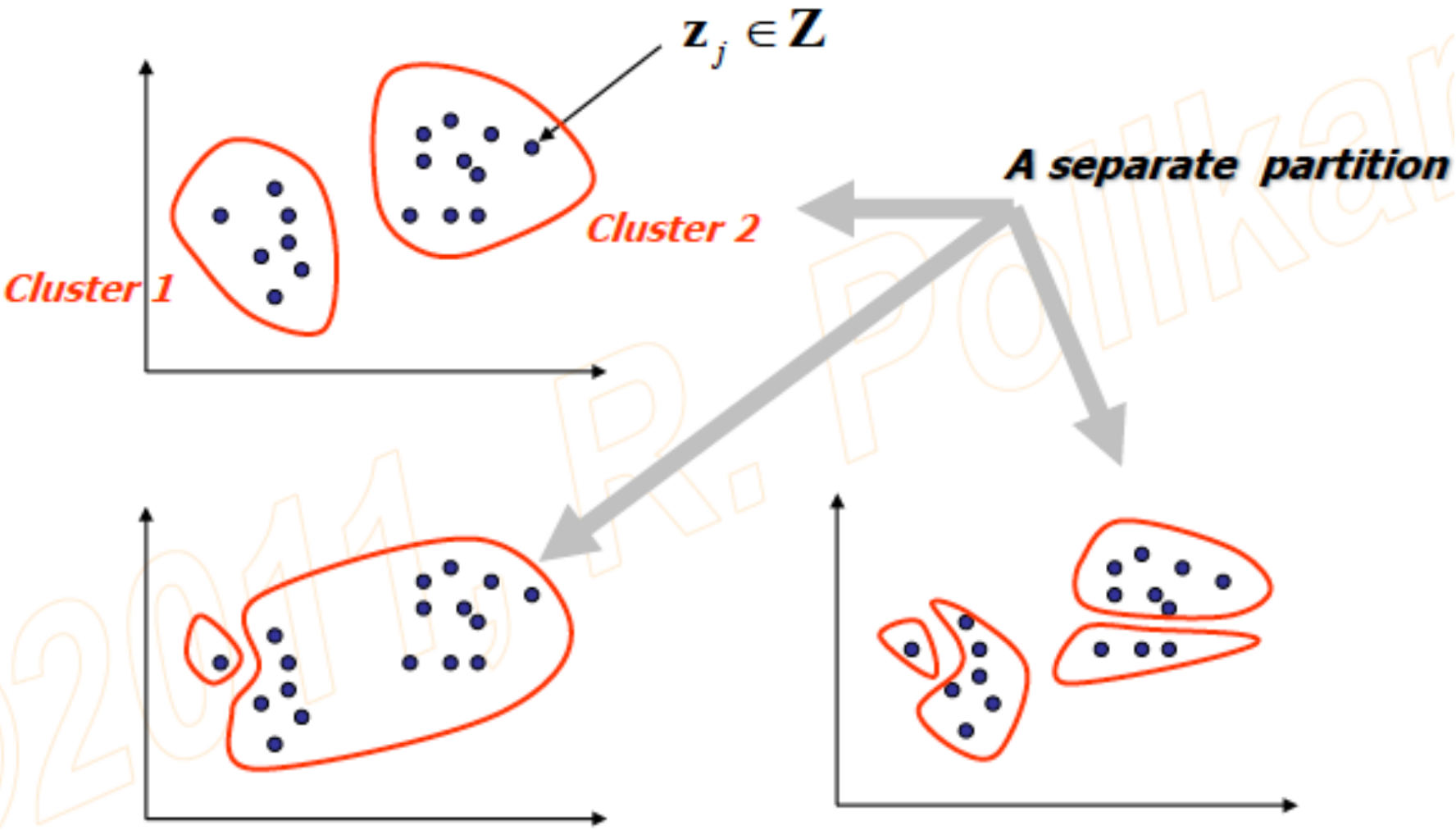
- MetaPhlan
  - Based on marker genes in sample
  - Blasting against fraction of database (that captures differences) speeds things up
- Phylosift
  - Placing reads into a phylogenetic tree
  - Disadvantage: slow

Clustering

- Clustering is the identification of subsets of data that can be grouped together based on some similarity measure.
- Also called unsupervised learning, clustering can be thought of as grouping the points in an unlabeled data set in order to discover structures in the data.
  - Hence, for any clustering algorithm, a similarity (or distance) measure, along with a clustering criterion is needed. Often Dissimilarity = Distance
- We would like the individuals in the same group to be similar to each other and dissimilar to the individual from the other groups.
- If the groups are compact and well separated, then they could be labelled and their characteristics interpreted.



Clustering



Hierarchical Clustering

There are two flavors:

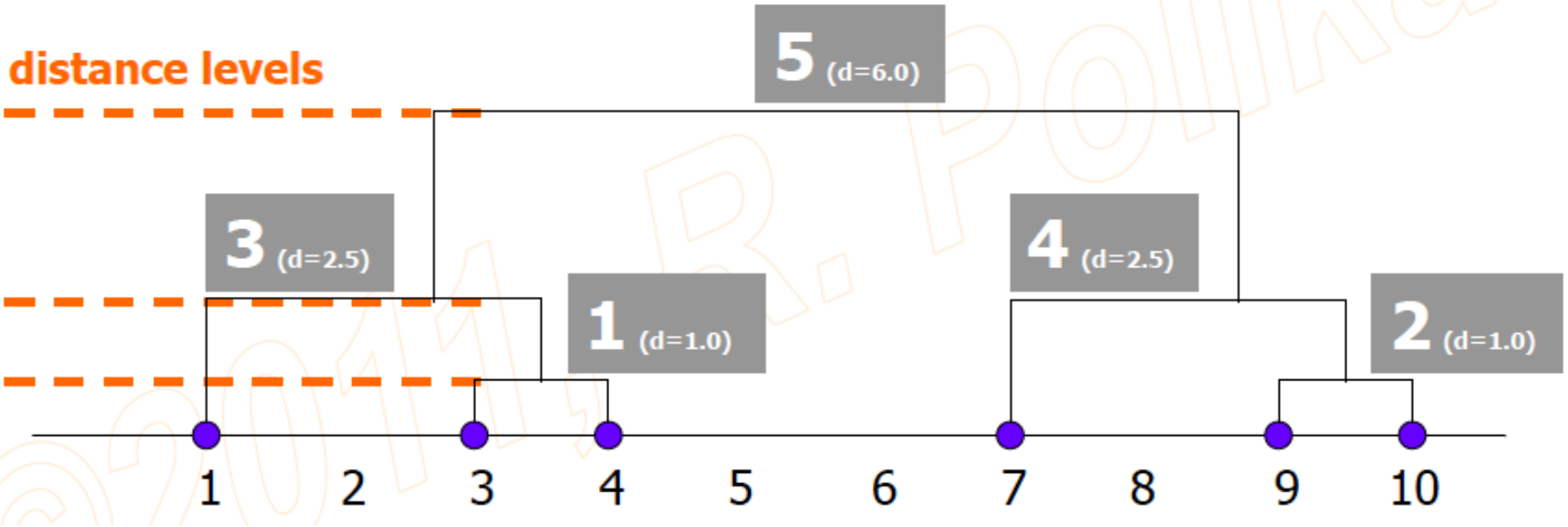
- **Agglomerative:** Start with every object being a cluster of its own. Group clusters successively until all objects fall in one single cluster.
- **Divisive:** Start with all objects in one single cluster. Split clusters successively until every object becomes a cluster of its own.

Similar to UPGMA algorithm, the general approach is

1. Start with a distance matrix between all pairs of data points. This is equivalent to starting with N clusters, $C_1,\ldots,C_N$, each containing a single individual x_j.
2. Iteratively identify closest pairs and merge them into a new cluster. That is, find the nearest pair of distinct clusters, say C_i and C_j, merge them, delete C_j and decrease the number of clusters by 1.
3. Compute the distance between the new cluster and all other points and clusters
4. Repeat 2-3.
5. Stop when only one element is left in the matrix.

Hierarchical Clustering

Hierarchical clustering uses ***dendograms***, 2-d tree structure representing the process of grouping nested clusters.



K-means Clustering

One of the most commonly used clustering algorithms in machine learning

- It requires that the number of clusters be predefined.
- It is also sensitive to initial choice of clusters.

Based on the square Euclidean distance metric.

Begin Initialize n, c, $\mu_1, \mu_2, \ldots, \mu_c$ ← Cluster centers

Do Classify n samples according to nearest μ_i

Recompute μ_i

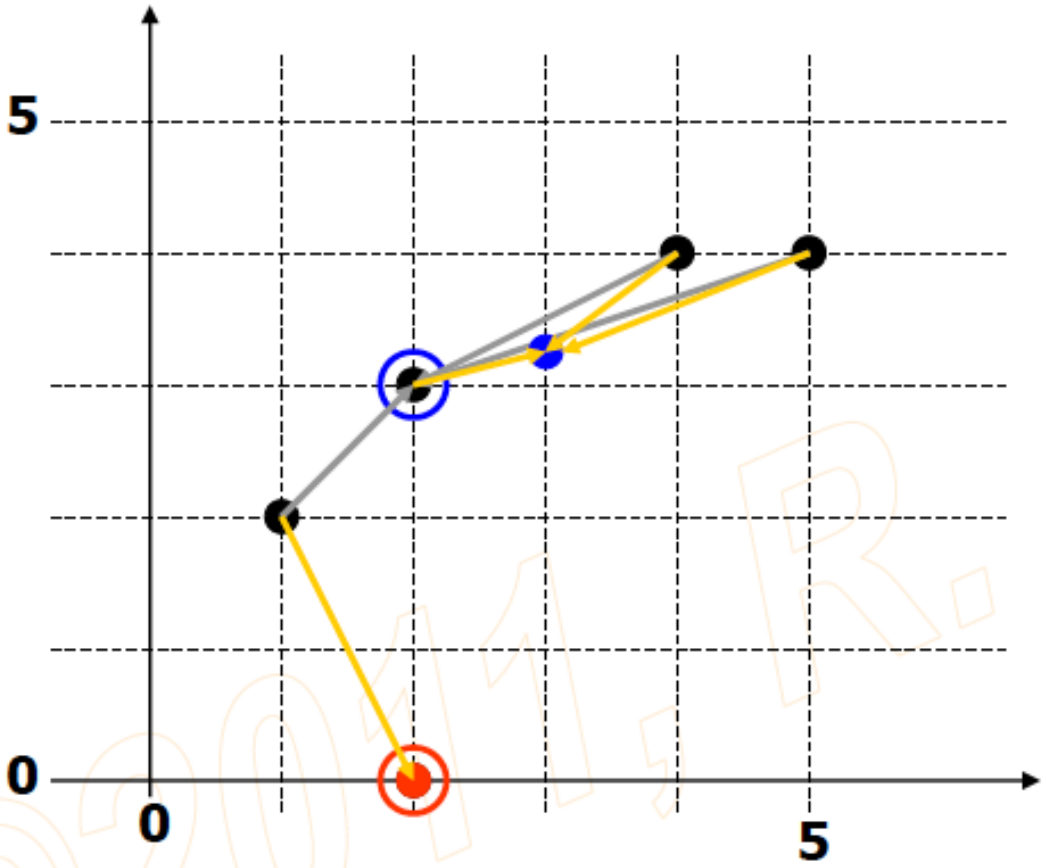
until no change in μ_i

Return $\mu_1, \mu_2, \ldots, \mu_c$

End

$$\boldsymbol{\mu}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in \omega_i}^{n_i} \mathbf{x}$$

K-Means



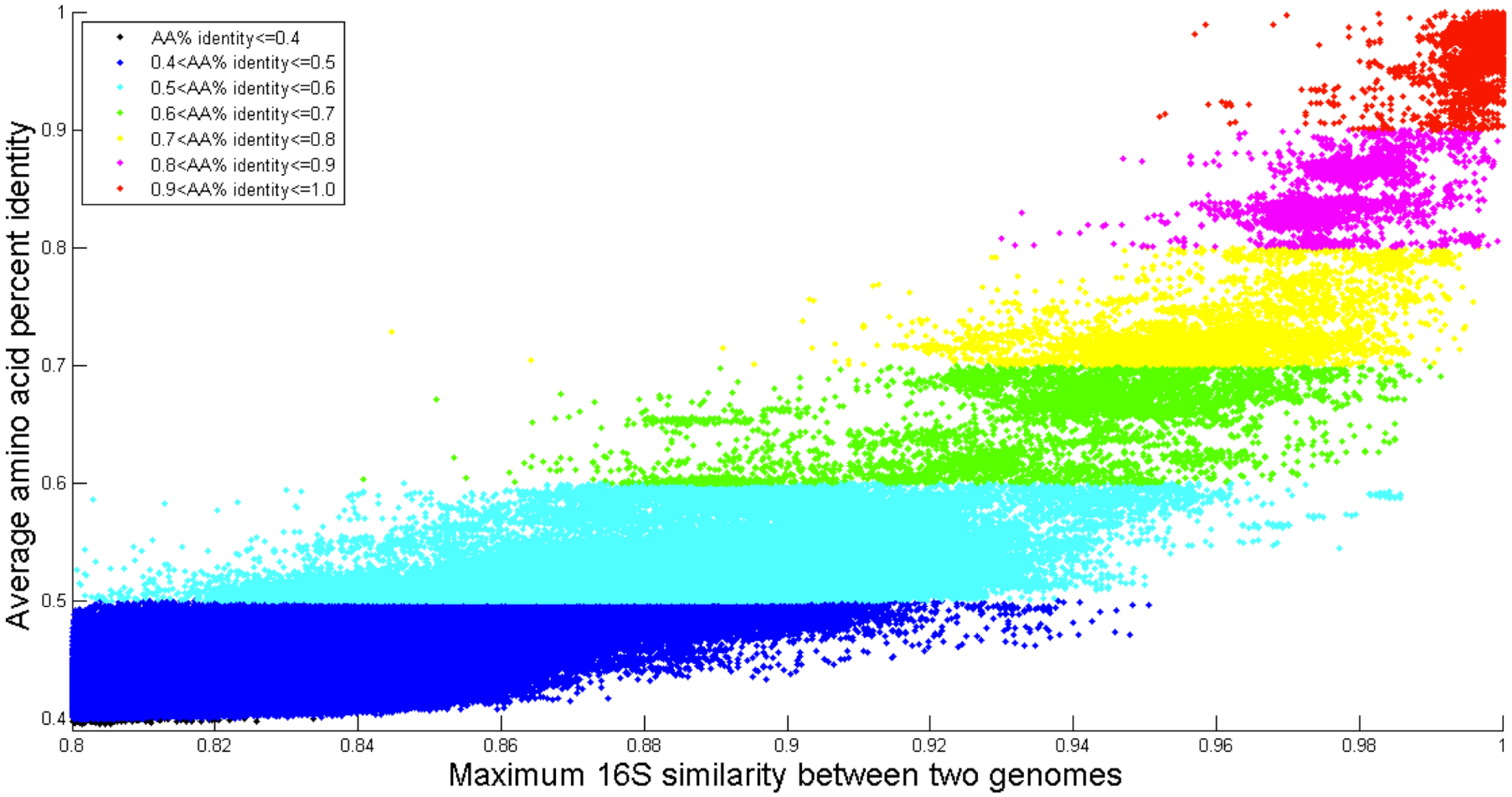
1. Random initial choice of the two means
2. Cluster the data according to the nearest mean
3. Recalculate the means
Change? Yes
2. Cluster the data according to the nearest mean
3. Recalculate the means

Continue at home…

Clustering into Sequence Identity

- Nice to be able to say that every member of group is ~97% identical
  - Counting substitutions and indels
- Rough Rule of Thumb for 16S rRNA
  - 97% Identitcal ~ Within same Species
  - 94% Identical ~ Within same Genus
  - 91/92% Identical ~ Within same Order
  - 88% Identical ~ Within Same Phylum

Linking 16S rRNA to Function



Upcoming comparisons of Clustering Methods

- CD-Hit
  - Alignment but use lots of heuristics to speed up search
- Uclust
  - Heuristics that speed things up but assume that sequences are highly similar
- Other U tools
  - Pipelines to speed up the whole analysis