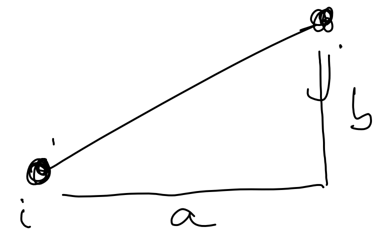
Dissimilarity Measures for Quantitative Data

- **Euclidean Distance**

$$d_{ij} = \left\{ \sum_{k=1}^{p} (x_{ik} - x_{jk})^2 \right\}^{1/2}$$



$$d_{ij} = \sqrt{a^2 + b^2}$$

- **Manhattan (taxi-cab) distance**

$$d_{ij} = \sum_{k=1}^{p} | x_{ik} - x_{jk} |$$

$$d_{ij} = a + b$$

- **Scaled Euclidean Distance**

$$d_{ij} = \left\{ \sum_{k=1}^{p} w_k^2 (x_{ik} - x_{jk})^2 \right\}^{1/2}$$

where w_k are suitable weights

e.g. $\left(\text{std. dev of variable } k\right)^{-1}$ or $\left(\text{range of } k\text{th variable}\right)^{-1}$

Metric vs. Non-metric

A non-negative function, $g(x,y)$, is **metric** if:

i) Satisfies the triangle inequality:

$$g(x,y) \leq g(x,z) + g(y,z)$$

ii) Symmetric:

$$g(x,y) = g(y,x)$$

iii) $g(x,y) = 0$ only if $x = y$

Euclidean Dist. is a metric function
(as is Manhattan distance)

Other Quantitative Measures of Dissimilarity

- Minkowski Metric

$$d_{ij} = \left\{ \sum_{k=1}^{p} \left| x_{ik} - x_{jk} \right|^{\lambda} \right\}^{1/\lambda} \quad \text{for integers } \lambda$$

$\lambda = 1$ is Manhattan distance, $\lambda = 2$ is Euclidean Dist.

- Canberra Metric

$$d_{ij} = \sum_{k=1}^{p} \frac{|x_{ik} - x_{jk}|}{(x_{ik} + x_{jk})}$$

[Accts for distance btw. points & relationship to origin]
→ only for non-negative values

- Czekanowski Coefficient

$$d_{ij} = 1 - \frac{2 \sum_{k=1}^{p} \min(x_{ik}, x_{jk})}{\sum_{k=1}^{p} (x_{ik} + x_{jk})}$$

[% dissimilarity over all variables]

Quantitative Dissimilarity for Variables

Correlation provides a suitable measure of similarity

$d_{kl} = 1 - r_{kl}$ if $r_{kl} = -1$ is taken to indicate maximum disagreement

$d_{kl} = 1 - r_{kl}^2$ is appropriate if $r_{kl} = 1$ and $r_{kl} = -1$ are treated equivalently (predictive power)

$$d_{kl} = 1 - \frac{\sum_{i=1}^{n} x_{ik} x_{il}}{\left(\sum_{i=1}^{n} x_{ik}^2 \sum_{i=1}^{n} x_{il}^2 \right)} \quad \leftarrow \text{uncentered correlation}$$

Dissimilarity for Dichotomous Data

For each pair of objects of interest form a 2×2 contingency table

		obj 2 +	obj 2 −
obj 1	+	a	b
	−	c	d

$a + b + c + d = p$

Simple Matching: $d_{ij} = 1 - \dfrac{a+d}{p} = \dfrac{b+c}{p}$

Jaccard Coefficient: $d_{ij} = \dfrac{b+c}{a+b+c}$ (joint absence does not contribute)

Czekanowski Coeff: $d_{ij} = \dfrac{b+c}{2a+b+c}$

Dissimilarity btwn. Variables

$a + b + c + d = n$ (# of objects/individuals)

a = # of objects showing + for both variables, k & l

... etc.

	+	−
+	a	b
−	c	d

$$\chi^2 = \frac{(ad - bc)^2 (a+b+c+d)}{(a+b)(a+c)(c+d)(b+d)}$$

$$d_{kl} = 1 - \sqrt{\frac{\chi^2}{n}}$$

Dissimilarities for mixed data types

Gower (1971) suggests:

$$S_{ij} = \frac{\sum_{k=1}^{p} W_{ijk} S_{ijk}}{\sum_{k=1}^{p} W_{ijk}}$$

where S_{ijk} is the similarity for i & j based on variable k

- recommends

$S_{ijk} = 1$ for binary data w/ positive match & categorical data when i and j in same category

$$S_{ijk} = 1 - \frac{|X_{ik} - X_{jk}|}{R_k}$$

for continuous variables where R_k is range of variable k

$W_{ijk} = 0$ when k missing on i or j

$W_{ijk} = W_k$ otherwise (often 1)

Define dissimilarity as:

$$d_{ij} = (1 - S_{ij})^{1/2}$$

Introduction to Clustering

Goal of Clustering

- Find "natural groups" in data

→ one definition:

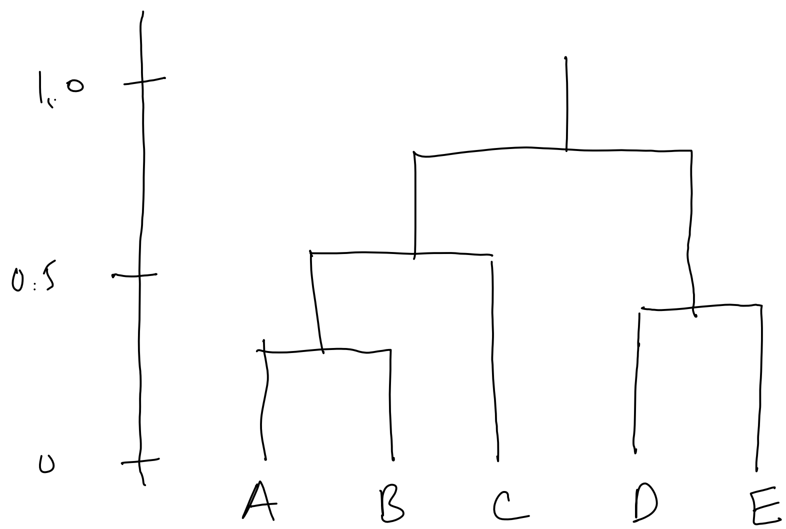
Patches of high dimensity surrounded by patches of lower density in the p-dimensional space defined by the variates

Hierarchical Clustering

Agglomerative / Divisive methods

- In practice almost always agglomerative

For n data points define a set of $n-1$ joins that represent groupings of objects @ different levels of similarity



Simple Algorithm for Hierarchical Clustering

1) Calculate a dissimilarity matrix for the n items

2) Join the two nearest items, i & j

3) Delete the i^{th} & j^{th} row and column of the dissimilarity matrix; add a new row/column
* that represents dissimilarity of new group (i,j) to all other items

4) Repeat from step 2 until there is a single group

Methods of Hierarchical Clustering

The different methods are determined by the function used to determine the distance between groups

Some Common Group Distance Criteria

Single linkage (nearest neighbor)

Complete linkage (furthest neighbor)

Group average

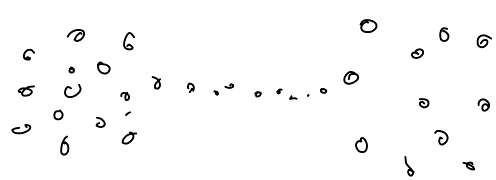
Centroid

Single Linkage Clustering

n_i, n_j are # of objects in groups i & j

☆ D_{ij} is the <u>smallest</u> of the $n_i n_j$ dissimilarities between each element of i & each element of j

- → Invariant under monotonic transformation of the d_{ij}
- → Unaffected by ties
- → Provably nice assymptotic properties
- → Susceptible to "chaining"



Complete Linkage

D_{ij} is the maximum of the $n_i n_j$ dissimilarities between the two groups

→ also invariant under monotonic transformation

Group average

D_{ij} is the average of the $n_i n_j$ dissimilarities between the two groups (UPGMA, WPGMA)

Centroid method

D_{ij} is the squared euclidean distance between the centroids of groups i & j

Hierarchical Clustering, A worked Example

	A	B	C	D	E
A	0				
B	4	0			
C	1	4	0		
D	4	2	4	0	
E	5	5	3	4	0

Single Linkage

	(A, C)	B	D	E
(A, C)				
B	4			
D	4	2	0	
E	3	5	4	0

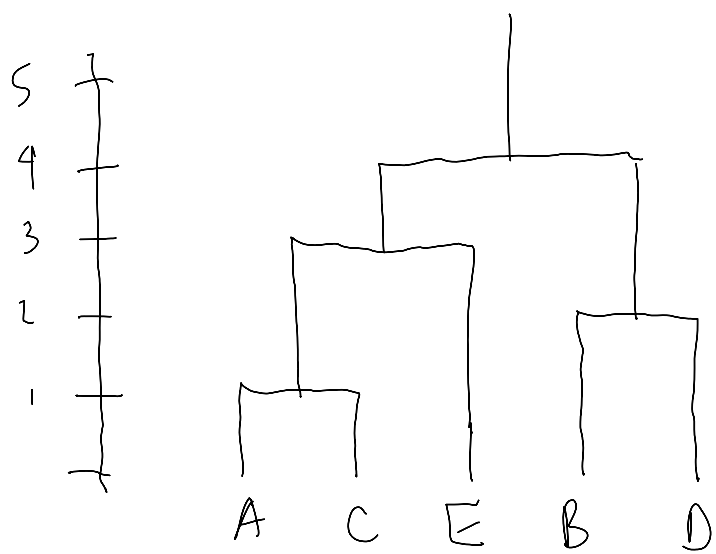
	(A, C)	(B, D)	E
(A, C)	0		
(B, D)	4	0	
E	3	4	0

Worked Example, cont.

	((A,C),E)	(B,D)
((A,C),E)	0	
(B,D)	4	0

$\rightarrow$ Only one Choice

$(((A,C),E),(B,D))$



Single Linkage Clustering

K-means Clustering

General idea: assign the n data points to one of K clusters so as to optimize some criterion of interest

→ Most common criteria is to minimize the sum-of-squares from the group centroid

$$V = \sum_{i=1}^{K} \sum_{j \in S_i} |x_j - \mu_i|^2$$

Simple Algorithm for K-Means Clustering

1. Decide on K, the number of groups
2. Randomly pick K of the objects to act as the initial "centers"
3. Assign each object to the group whose center it's closest to
4. Recalculate the K "centers" as the centroids of the objects assigned to them
5. Repeat from step 3 until centers no longer move (convergence)

Things to Note re: K-means

- The algorithm described above does not necessarily find the global optimum
- The algorithm is sensitive to choice of initial cluster centers
  - → often run multiple times w/ different initial centers