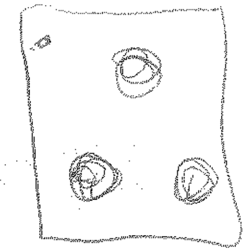
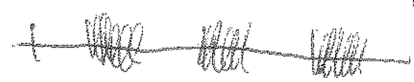


Project from 2D to 1D



Visualizing High-D Structure / Projections

What do we do when we have **many** continuous variables?

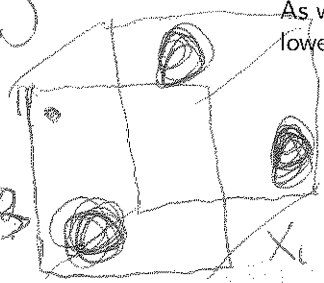
Example situations when we have many continuous variables:

Projections: Sometimes we want to project the high dimensional data into a smaller subspace without losing "important structure".

associations, group structure, outliers

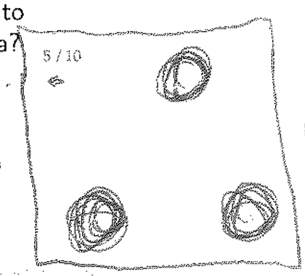
As we will see on the upcoming lab and HW, we can project the data into lower-dim space and visualize the results. Why might this be a bad idea?

3D



project into 2D $\longrightarrow$

goal: preserve the order of distances between pairs of obs.



Distance = Metric = Distance Metric = Distance Function

Function that defines distance between pairs of observations in a dataset

$\binom{n}{2} = \frac{n(n-1)}{2}$

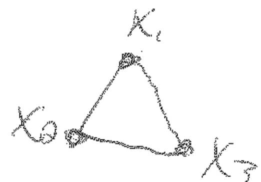
Properties: X_i, X_j are observations in our dataset (rows)

Interested in distance between X_i and X_j, $d(X_i, X_j)$

Non-negativity: $d(X_i, X_j) \geq 0$

Identity: $d(X_i, X_j) = 0 \iff X_i = X_j$

Symmetry: $d(X_i, X_j) = d(X_j, X_i)$



Triangle inequality: $d(X_1, X_3) \leq d(X_1, X_2) + d(X_2, X_3)$

Examples:

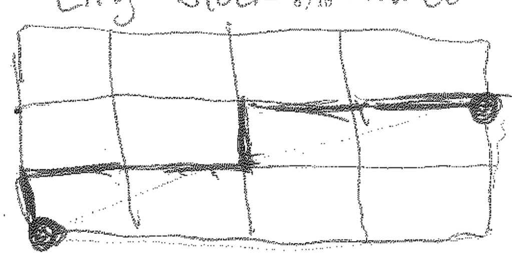
Euclidean Distance

$$d(X_i, X_j) = \sqrt{\sum_{k=1}^{p} (X_{ik} - X_{jk})^2}$$

Manhattan Distance

"city-block distance"

6/10



$$D = \begin{array}{c} \\ 1 \\ \vdots \\ n \end{array}\begin{array}{c} 1 \; \cdots \; n \\ \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix} \end{array}$$

n rows
n colums

Distance Matrices

A **distance matrix** is a data structure that efficiently organizes the pairwise distances between all observations in a dataset.

Pairwise distances are organized into the lower-triangle of a matrix, D

The $(i,j)^{th}$ element of the matrix contains the distance between x_i and x_j:

$D[i,j] = d(x_i, x_j)$ $= d(x_j, x_i) = D[j,i] \rightarrow D$ is symmetric

Examples: Suppose data has 4 obs. $\Rightarrow$ D will be 4×4

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 3 & \pi & x \\ 2 & 3 & 0 & & y \\ 3 & \pi & & 0 & z \\ 4 & x & y & z & 0 \end{array}$$

$\rightarrow$ Diagonal of D is 0

$\rightarrow$ cell(i,j) = cell(j,i)

$\rightarrow$ D is "symmetric"

$\rightarrow$ all values are non-negative

$x \geq 0$, $y \geq 0$, $z \geq 0$, ≥ 0

Projections: MDS and PCA

Multi-dimensional scaling: looks for a configuration in a k-dimensional subspace such that the distances between observations in the subspace best match the distances in the original p-dimensional space.

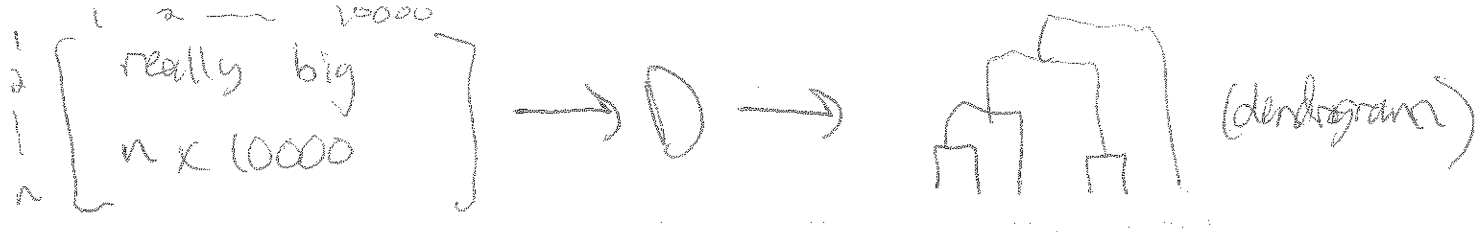
$$\begin{array}{c} 1, 2, \cdots \\ p = 10{,}000 \end{array}$$
$$\begin{bmatrix} \text{really big} \\ n \times 10{,}000 \end{bmatrix} \longrightarrow D \longrightarrow \begin{array}{c} 1, 2, \cdots k \\ \begin{bmatrix} n \times 2 \end{bmatrix} \end{array}$$

$k << p$

k is typically chosen to be 2

Principal Components Analysis: tries to represent large number of correlated continuous variables with a (usually) smaller number of uncorrelated "principal components" (new variables)

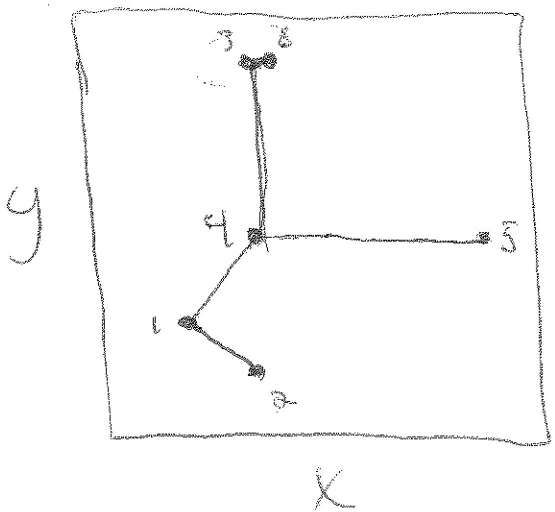
$$\begin{array}{c} 1, 2, \cdots \quad 10{,}000 \\ \begin{bmatrix} \text{really big} \\ n \times 10{,}000 \end{bmatrix} \end{array} \xrightarrow[\text{matrix algebra}]{\text{stuff}} \begin{array}{c} 1, 2, \cdots k \\ \begin{bmatrix} \\ \\ \end{bmatrix} \end{array}$$

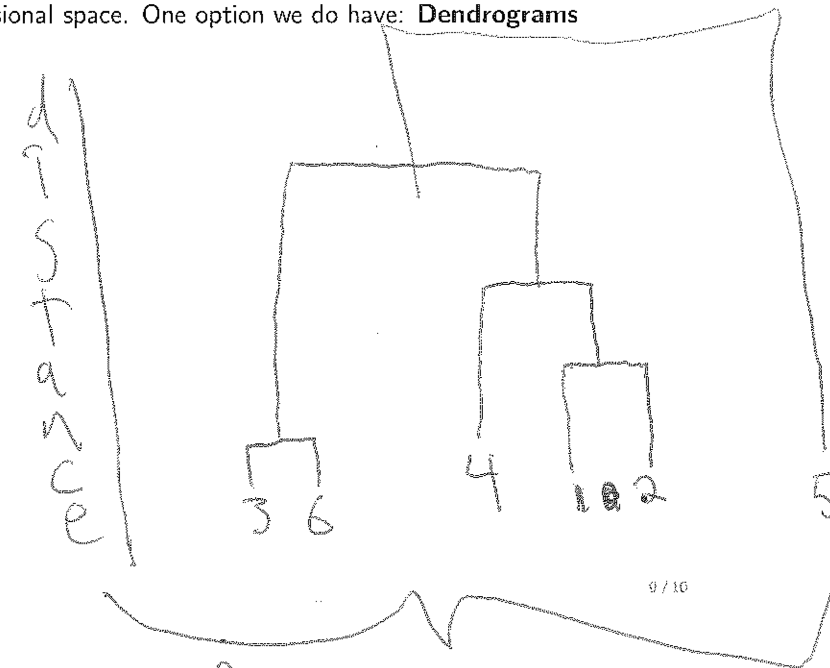


Visualizing High Dimensional Structure with Dendrograms

There is no easy way to visualize how far apart observations are in high-dimensional space. One option we do have: **Dendrograms**

2 variables, 6 obs





Dendrogram

Goal: link observations into groups / clusters

Hierarchical Clustering to Obtain Dendrograms

similar to minimal spanning tree

We can get different dendrograms via **hierarchical linkage clustering**

0. Start w/ all obs. in their own group / cluster
1. find distance between all pairs of obs in dataset
2. Link the two closest obs / "groups"
3. Re-calculate all of the distances between "groups"
4. repeat steps 2, 3 until all obs. are linked

Single linkage: the distance between two groups is the shortest possible distance between two points, one from each group

Complete linkage: the distance between two groups is the largest possible distance between two points, one from each group

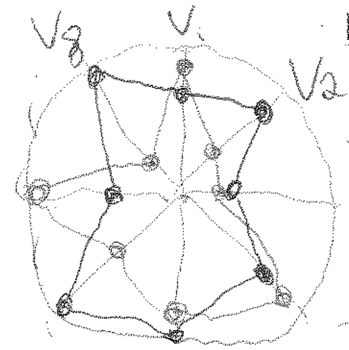
minimax linkage

average linkage

Lots of continuous variables at once

Radar chart: Adv
- many vars at once
- easy to compare two obs. on the same variable
- Kind of looks cool → bad reason

Radar Charts and Parallel Coordinates

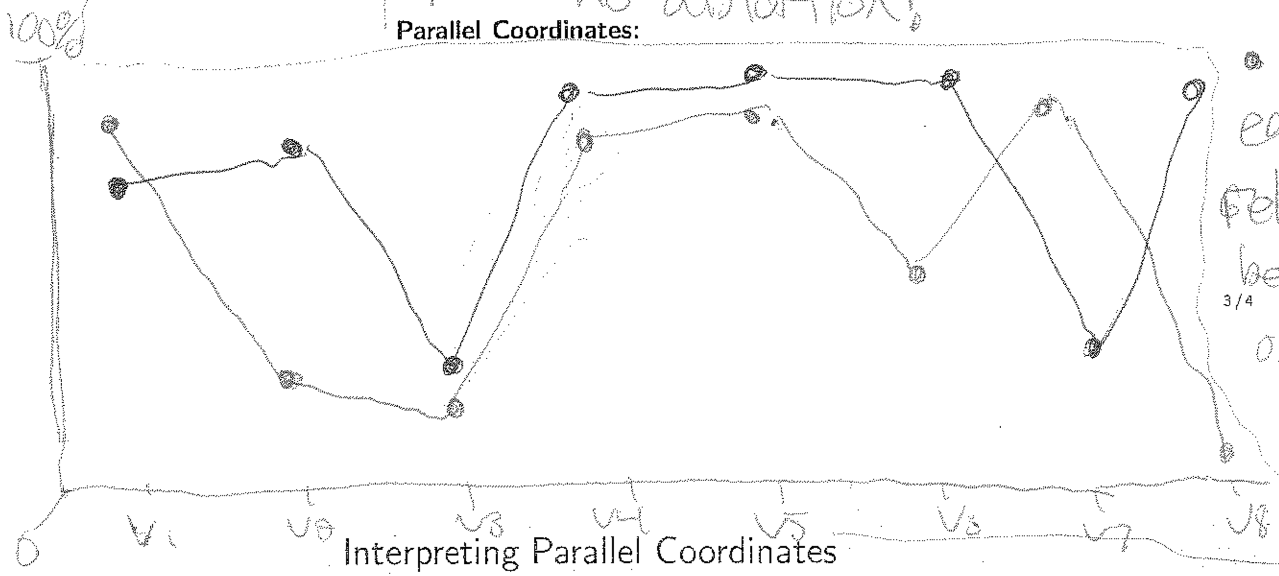


Radar Charts: Disadvantages:
- DISTORTION*

  *if used as an area plot
- Hard to compare more than 2-3 obs.

→ Adv: All of the advantages of radar, plus there's no distortion!

Parallel Coordinates:

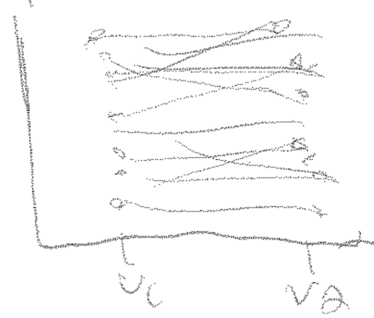


- We can more easily determine relationships between pairs of adjacent vars
- We can identify group structure

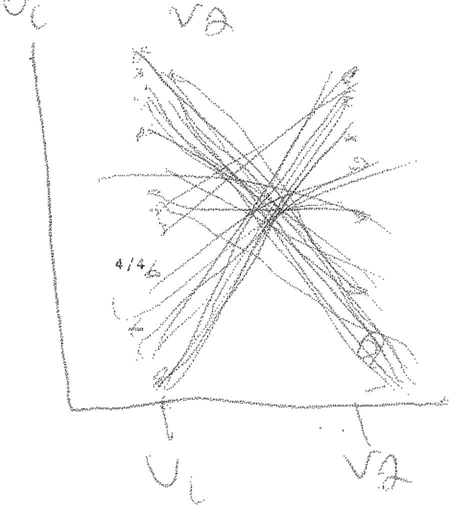
Interpreting Parallel Coordinates

* We can gain insight into the relationships between adjacent variables

1) high positive correlation



2) high negative correlation



3) No correlation