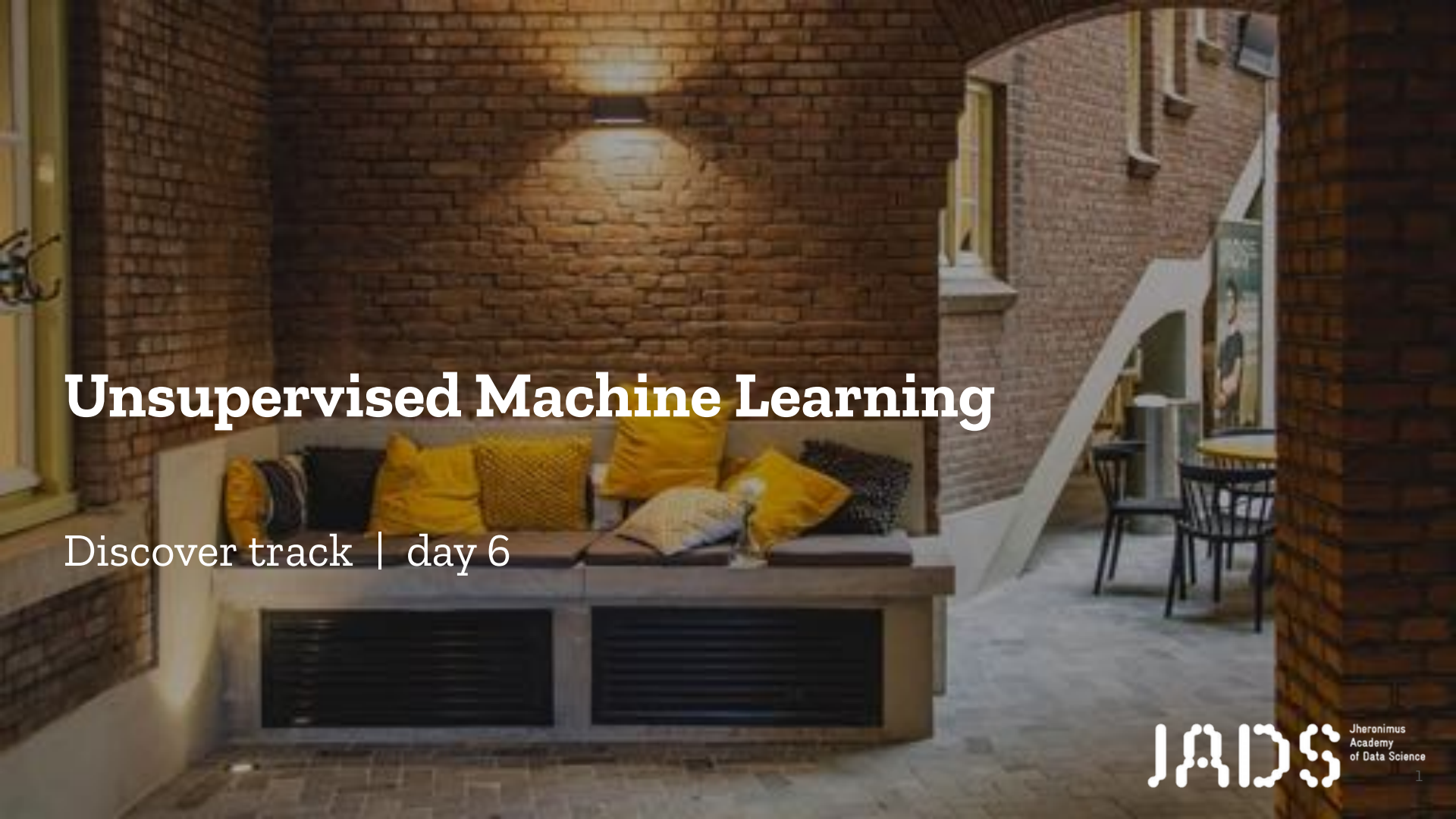


Unsupervised Machine Learning

Discover track | day 6

Jheronimus Academy of Data Science

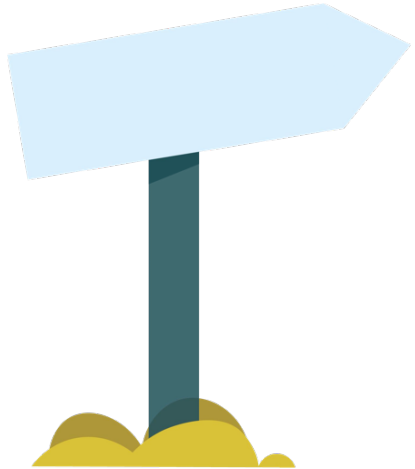
Attribution and copyright notice

This lecture is based on the following material available in the commons:

- Introduction to Statistical Learning, by James, Witten, Hastie and Tibshirani
- Lecture notes by Abass Al Sharif
- Paper dissected: visualizing data using t-SNE explained by Keita Kurita
- T-SNE Explained — Math and Intuition by Achinoam Soroker
- t-SNE clearly explained. An intuitive explanation of t-SNE... by Kemal Erde

Introduction to Data Science Projects & Machine Learning

- To learn how to leverage the strengths and mitigate the limitations of different unsupervised learning algorithms.
- To learn how to tune and optimize unsupervised machine learning models.
- To learn how to program and interpret unsupervised machine learning algorithms in Python.



Supervised vs. unsupervised learning

Supervised	Unsupervised
X and Y are known	Only X is known
Prediction	Exploratory data analysis, pre-processing
Assess quality of output with cross-validation and test set	No hard performance metric, subjective interpretation

Most common unsupervised tasks and algorithms

Dimensionality reduction	Clustering
Transformation of data from a high-dimensional space into a low-dimensional space whilst retaining some meaningful properties of the original data, ideally close to its intrinsic dimension.	*Set of techniques for finding subgroups, or clusters, in a data set*
• Principal component analysis (PCA)	• K-means clustering
• t-SNE	• Hierarchical clustering

Principal Component Analysis (PCA)

What are principal components?

- Reduce the features X_1, X_2, ... X_p to a smaller number of components with the highest variance

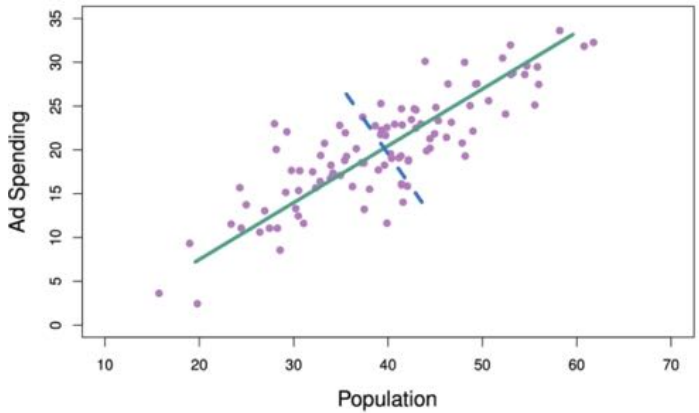
$$Z_1 = \phi_{11} X_1 + \phi_{21} X_2 + \ldots \phi_{p1} X_p$$

- Components Z_1, Z_2, ... Z_n are linear combinations of the original features

$$\sum_{j=1}^{p} \phi_{j1}^2 = 1$$

PCA illustrated

- The population size and ad spending for 100 different cities are shown as purple circles
- The green solid line indicates the first principal component
- The blue dashed line indicates the second principal component

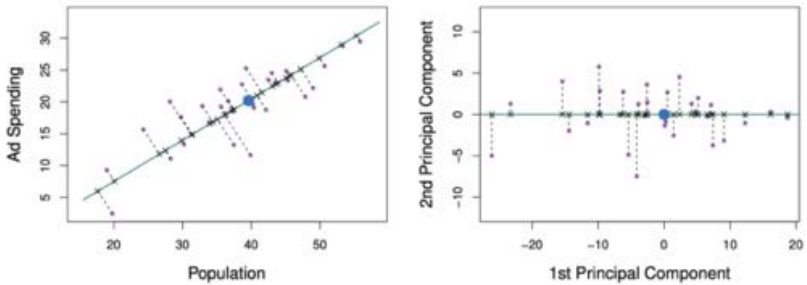


ISLR, figure 6.14

Interpretations of PCA

- The **blue dot** represents the **mean** population and ad spending. The first principal component is a single number summary: $Z_1 < 0$ indicates the city is below average

- The first principal component vector defines the line that is as *close as possible to the data*: the projections (dashed lines) are shortest



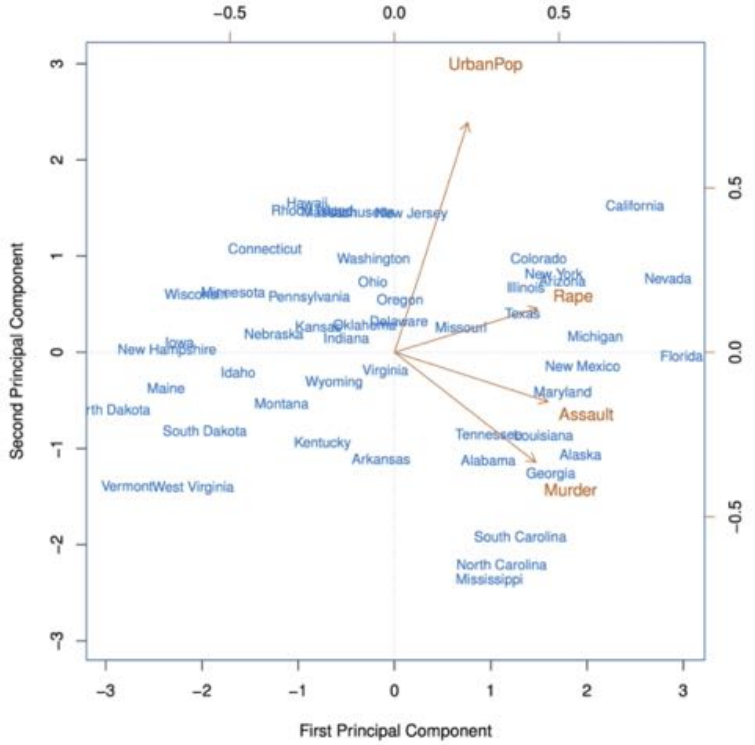
$$Z_1 = 0.839 \times (pop - \overline{pop}) + 0.544 \times (ad - \overline{ad})$$

ISLR, figure 6.15

Using PCA: biplot

- Consider the USArrests data with four features transformed to two components (table below)

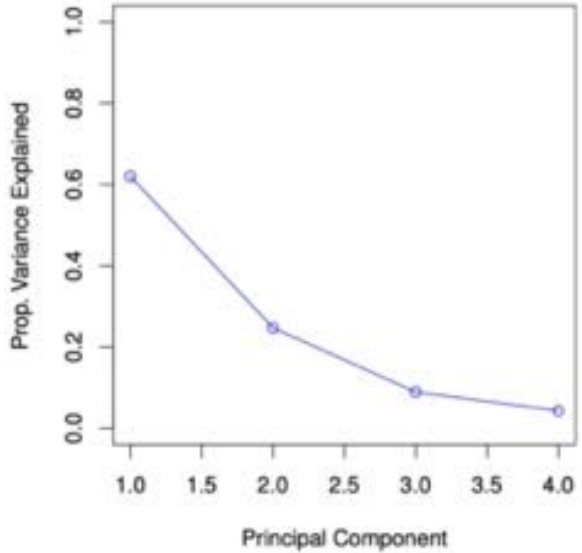
	PC1	PC2
Murder	0.536	-0.418
Assault	0.583	-0.188
UrbanPop	0.278	0.873
Rape	0.543	0.167

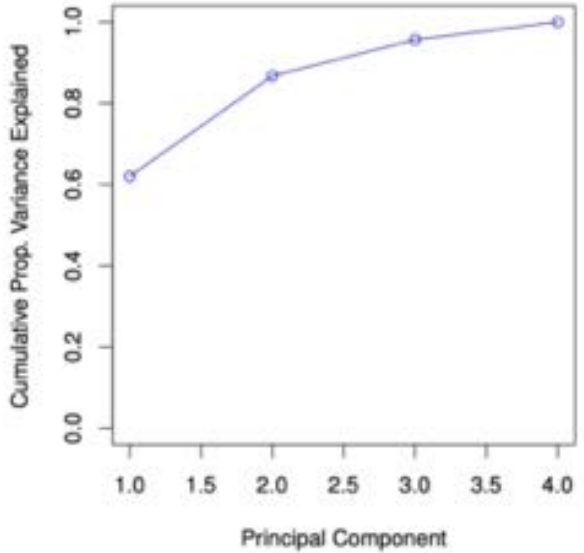


ISLR, figure 10.1

Using PCA: proportion of variance explained

- Scree plot (left): proportion of variance explained by each component
- Cumulative proportion (right)

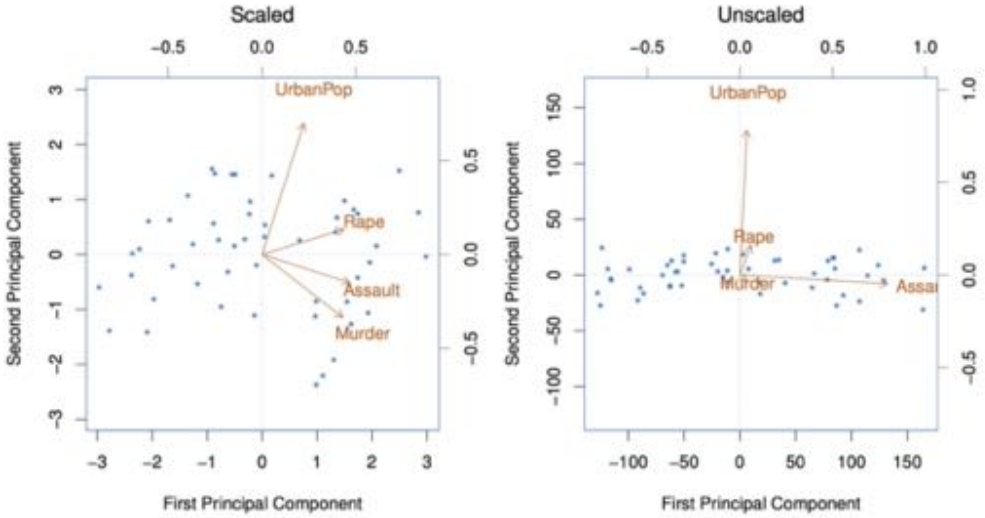




ISLR, figure 10.4

Using PCA in practice

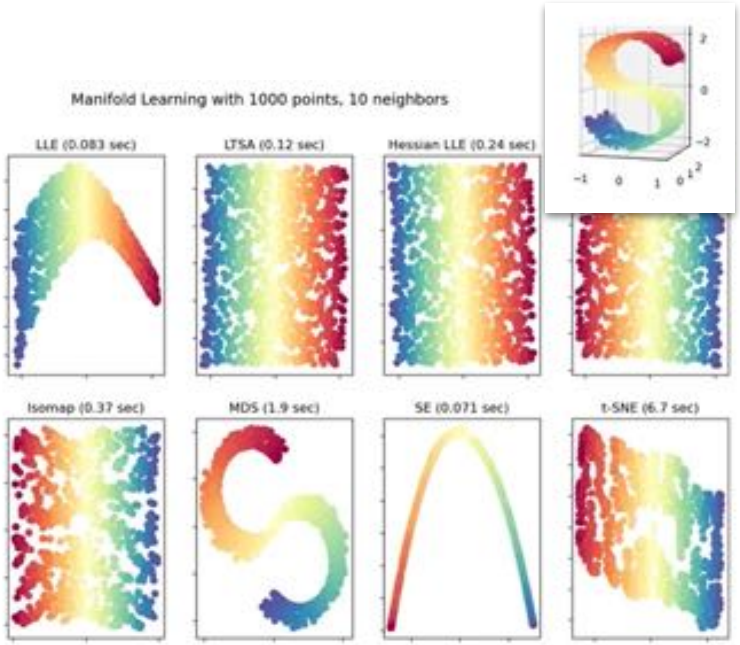
- Only applicable to continuous features
- All variables need to be
  - Centered (mean = zero), and
  - Scaled (variance = 1)
- PCA vectors are unique, up to a sign flip



ISLR, figure 10.3

t-SNE

Why do we need t-SNE?



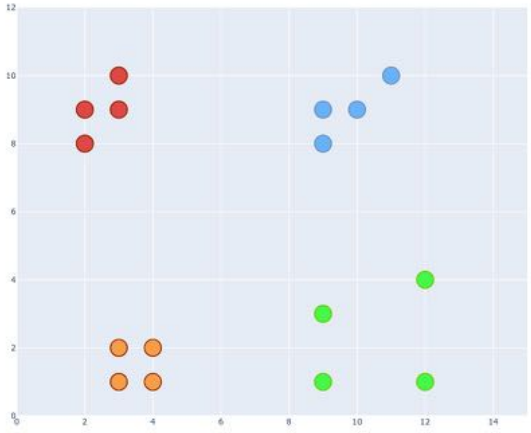
- Can handle non-linear relationships
- Solves the **'crowding problem'** by projecting into lower dimensional space with a heavy-tailed distribution
- Able to capture local and global structure via stochastics embedding with gradient descent

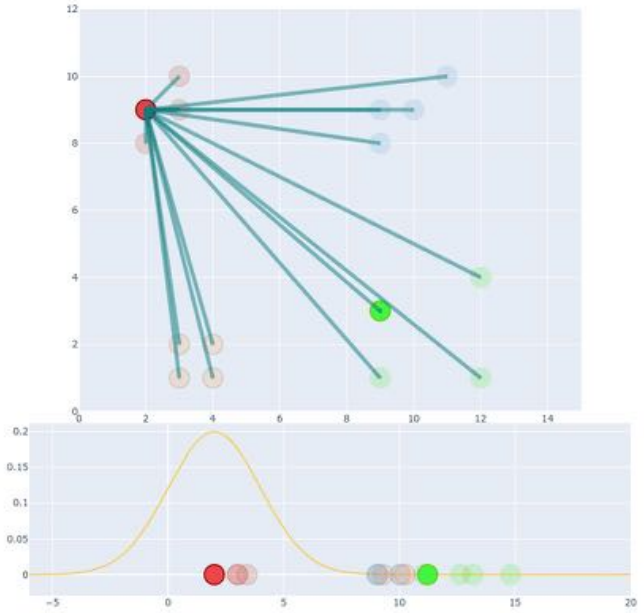
sci-kit learn documentation - Comparison of Manifold Learning Methods
Visualizing Data using t-SNE (original paper by Hinton and van der Maaten)

How does it work - compare with PCA

steps	t-SNE	PCA
1.	local proximity with Gaussian probabilities	global covariance
2.	t-distribution for mapping onto lower dimensional space to `spread out` the data	look for linear combinations of dimensions that contain largest variance

1. Local proximity with Gaussian variables

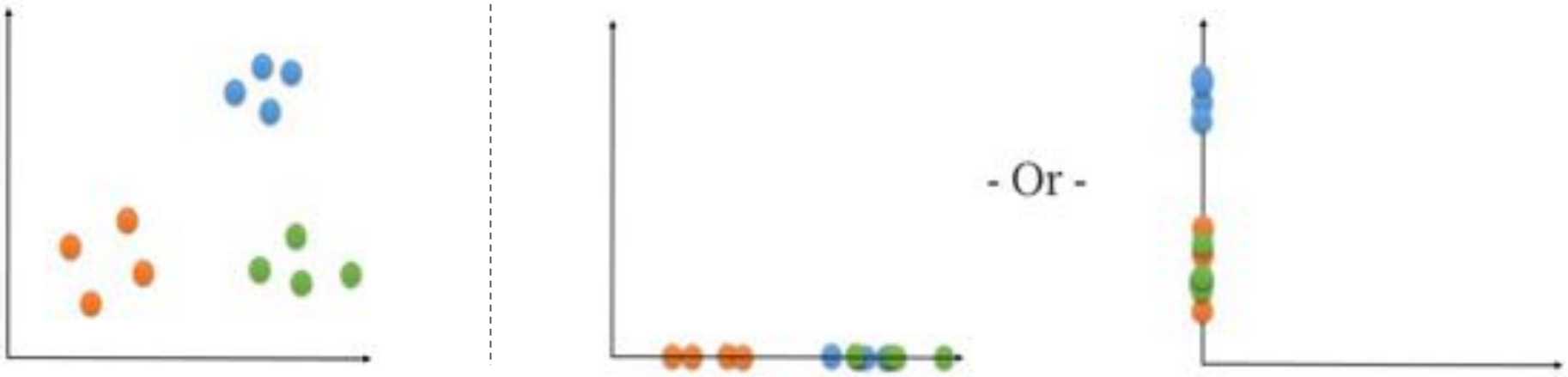




t-SNE clearly explained. An intuitive explanation of t-SNE... | by Kemal Erdem (burnpiro)

2. t-distribution for mapping to lower dimensional space

Crowding problem



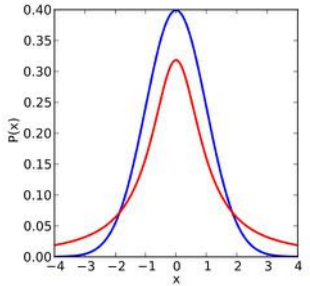
original data
(higher dimension)

mapping to lower dimension

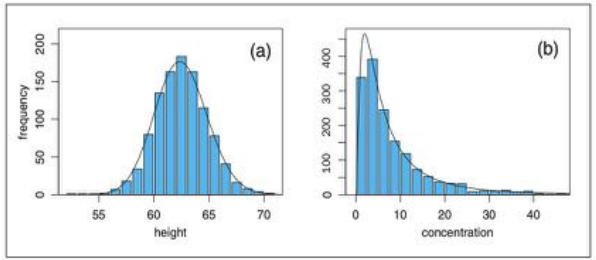
T-SNE Explained — Math and Intuition

Intermezzo: heavy-tailed distributions

Short-tailed distributions	heavy-tailed distributions
probabilities drop to zero exponentially	probabilities far beyond center
integral is finite	integral is not always infinite
example: normal distribution	examples: Student's t-distribution, log-normal distribution, Zipf distribution



blue: normal distribution
red: t-distribution



normal distribution (a) vs
log-normal distribution (b)

Zipf's law, frequency of words in language:

1. *the* 7%
2. *of* 3.5%
3. *and* 2.9%

→ proportional to 1/rank

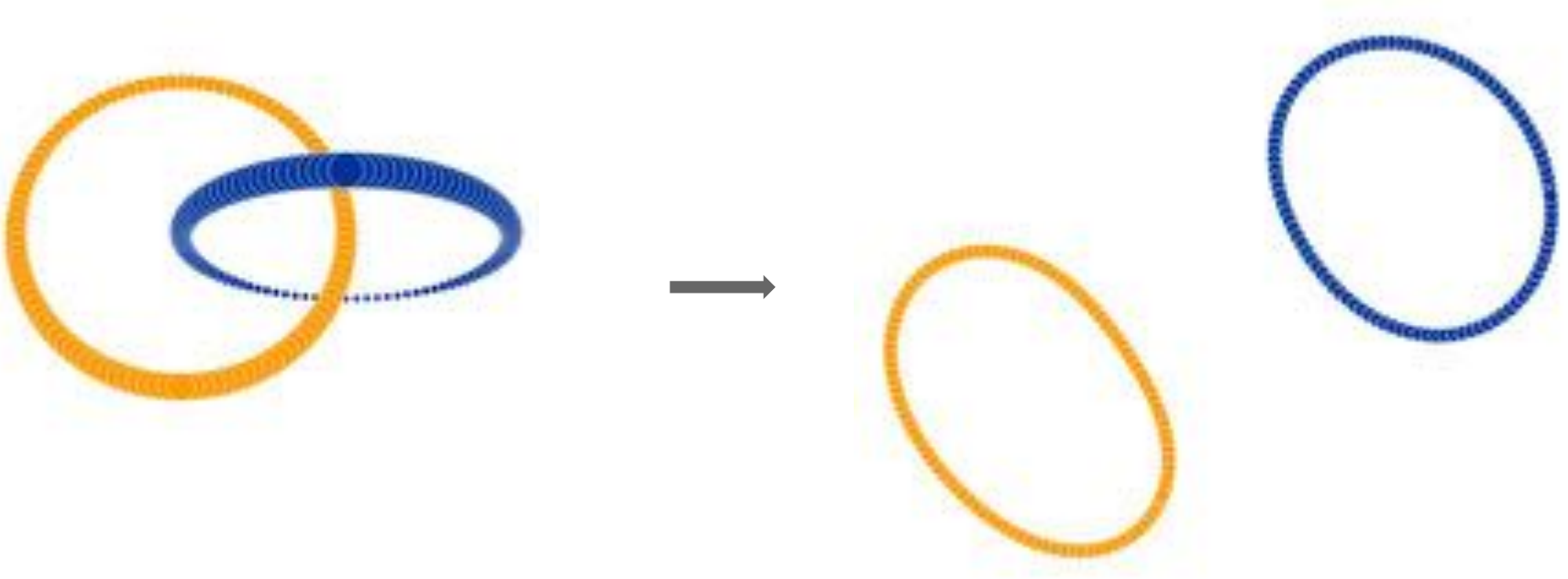
2. t-distribution for mapping to lower dimensional space

Convert pairwise distances to probabilities p_{ij}
using normal distribution in original **higher** dimension space

- Look for solution where $p_{ij} = q_{ij}$
- Heavy-tails compensates for crowding problem

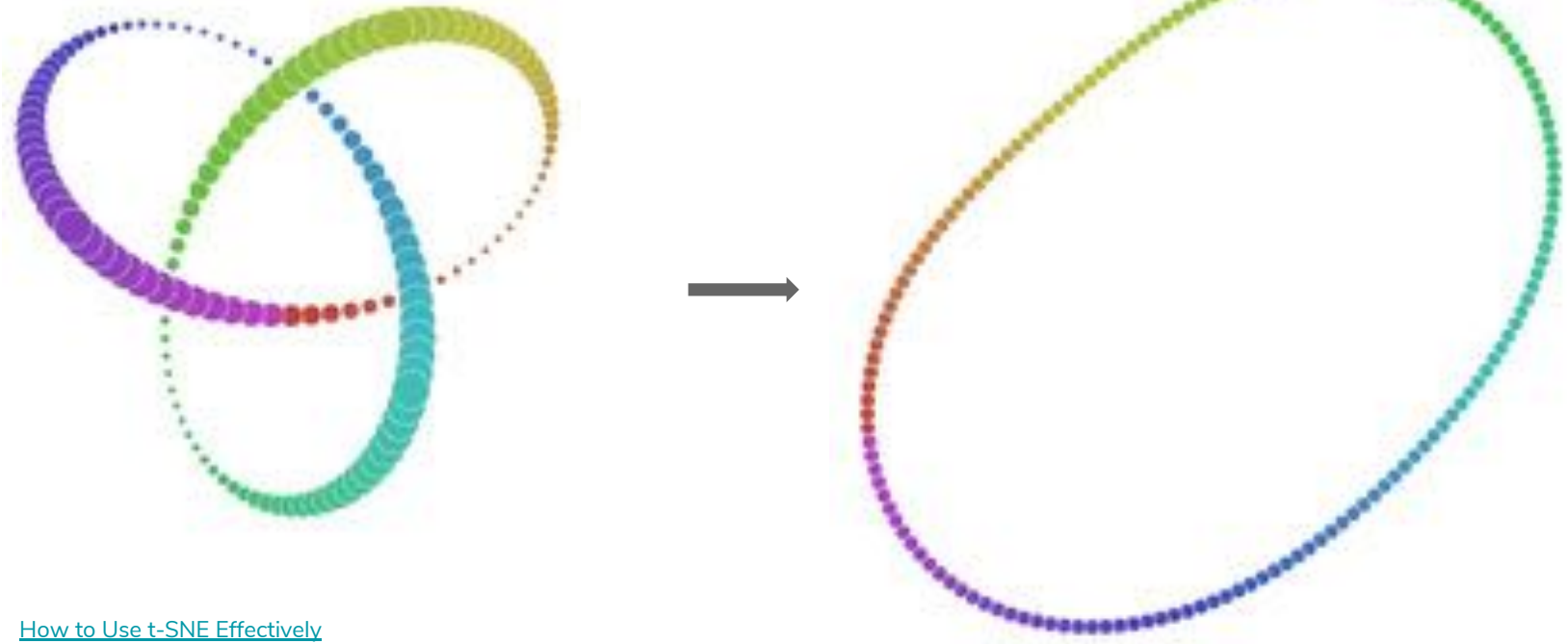
Convert pairwise distances to probabilities q_{ij}
using Student's t-distribution in **lower** dimensional space

Examples (perplexity 5)



How to Use t-SNE Effectively

Examples (perplexity 5)



How to Use t-SNE Effectively

Final thoughts

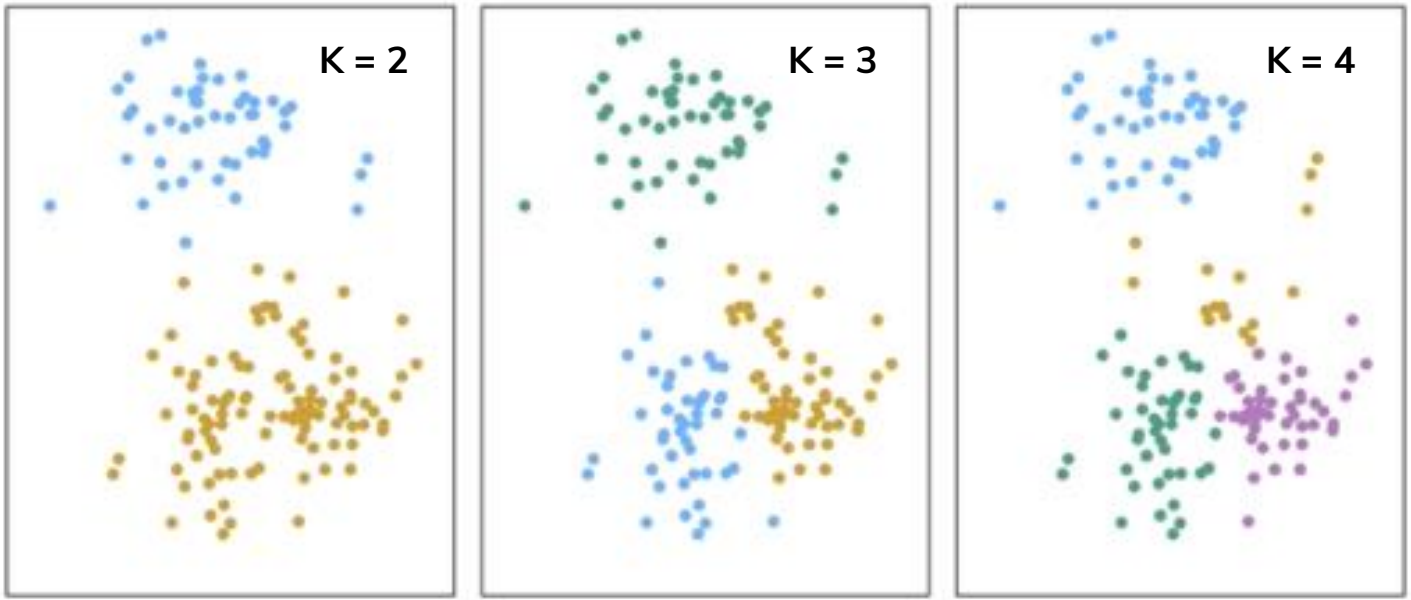
- t-SNE is stochastic, so it will yield different results for the same parameters!
  - Useful for exploratory data analysis
  - Not suitable for preprocessing (like PCA)

- Look at this paper and online simulation from Google Brain to get a feel of how it works

K-Means Clustering

K-Means clustering

- Specify number of cluster *K*
- Algorithm will create clusters



ISLR, figure 10.5

How does K-Means work?

- We would like to partition that data set into K clusters: $C_1, \ldots C_K$
  - Each observation belong to at least one of the K clusters
  - The clusters are non-overlapping, i.e. no observation belongs to more than one cluster

- The objective is to have a minimal “within-cluster-variation”, i.e. the elements within a cluster should be as similar as possible

- One way of achieving this is to minimize the sum of all the pairwise squared Euclidean distances between the observations in each cluster.

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \right\}$$

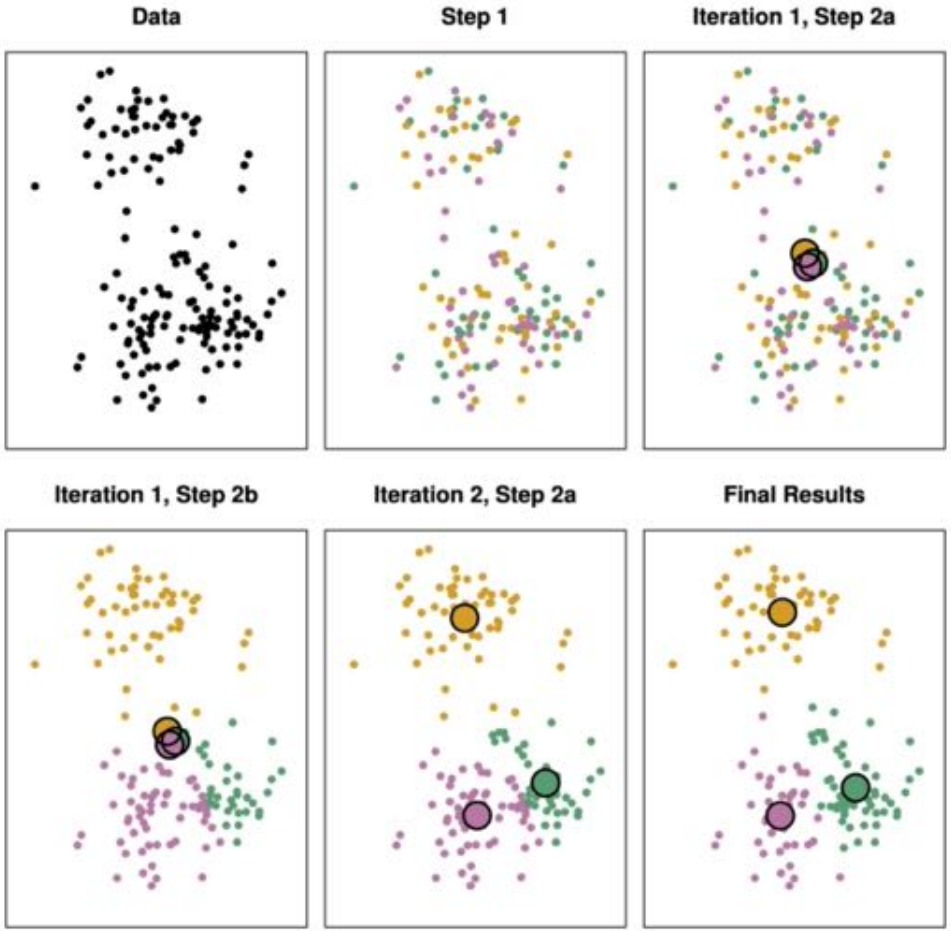
K-Means algorithm

- Initial Step: Randomly assign each observation to one of K clusters

- Iterate until the cluster assignments stop changing:
  - For each of the K clusters, compute the cluster centroid. The k^{th} cluster centroid is the mean of the observations assigned to the k^{th} cluster
  - Assign each observation to the cluster whose centroid is closest (where “closest” is defined using Euclidean distance)

K-means algorithm illustrated

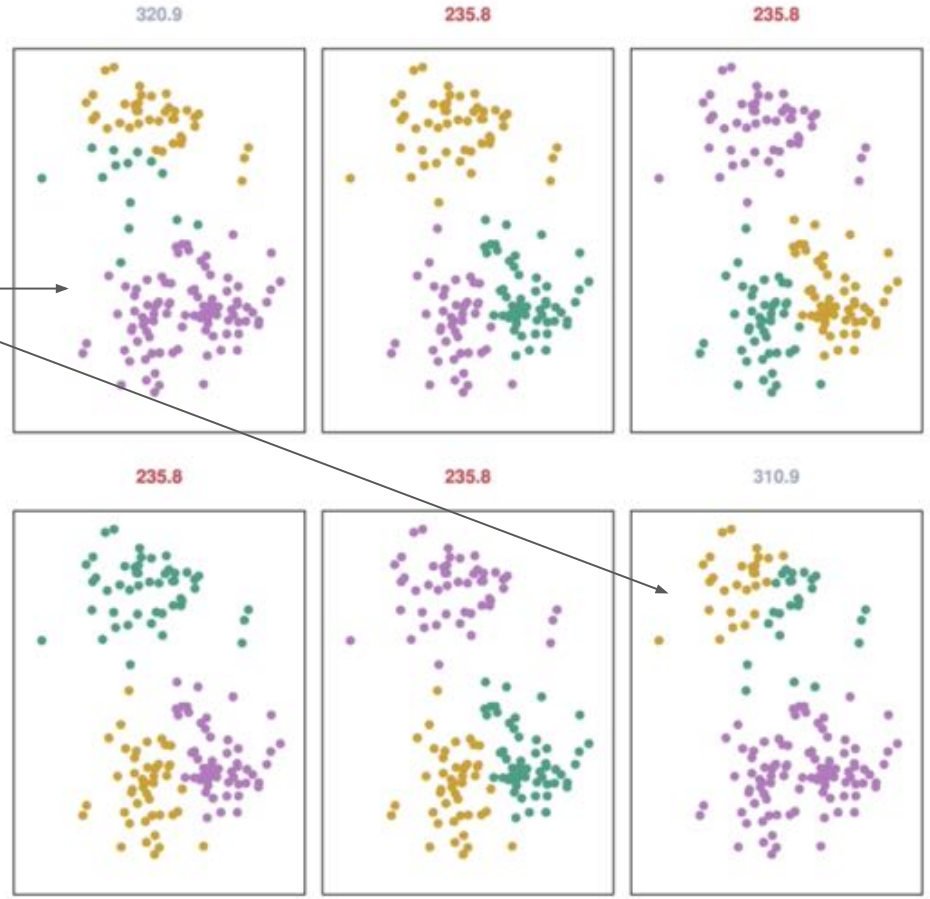
- Random assignment
- Compute cluster centers from random assignment
- Assign points to closes center
- Compute new centers
- Stop when there are no changes



ISLR, figure 10.6

Beware of local optimums

- Bad solutions: stuck in local optimum
- Hence, it is important to run the algorithm multiple times with random starting points to find a good solution



ISLR, figure 10.6

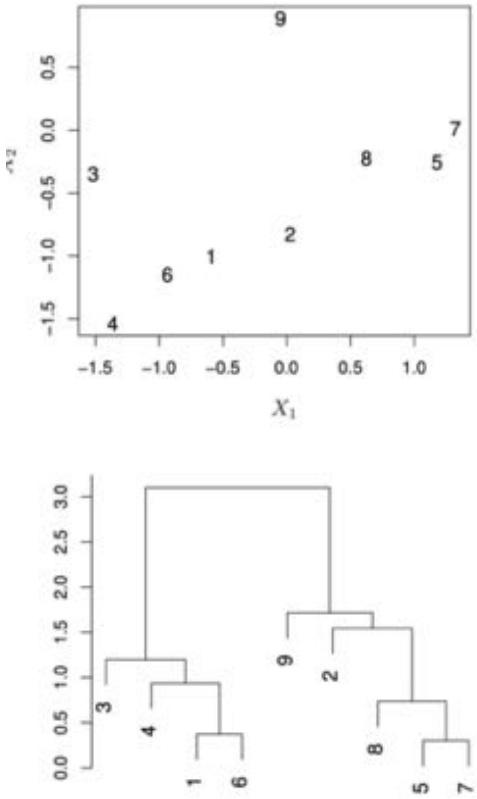
Hierarchical Clustering

Hierarchical Clustering

- K-Means clustering requires choosing the number of clusters.
- If we don’t want to do that, an alternative is to use Hierarchical Clustering
- Hierarchical Clustering has an added advantage that it produces a tree based representation of the observations, called a Dendogram

Dendograms

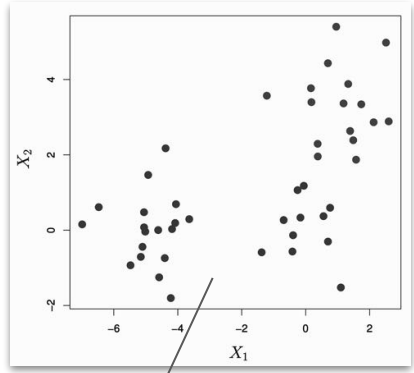
- First join closest points (5 and 7)
- Height of fusing/merging (on vertical axis) indicates how similar the points are
- After the points are fused they are treated as a single observation and the algorithm continues

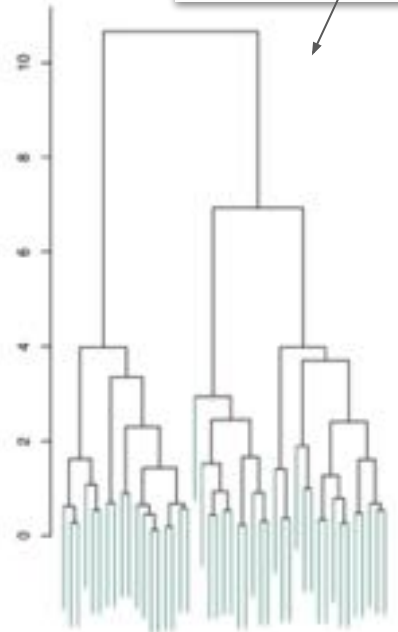


ISLR, figure 10.10

Interpretation

- Each “leaf” represents one observation
- At the bottom of the dendogram, each observation is a distinct leaf. However, as we move up the tree, some leaves begin to fuse. These correspond to observations that are similar to each other.
- As we move higher up the tree, an increasing number of observations have fused. The earlier (lower in the tree) two observations fuse, the more similar they are to each other.
- Observations that fuse later are quite different

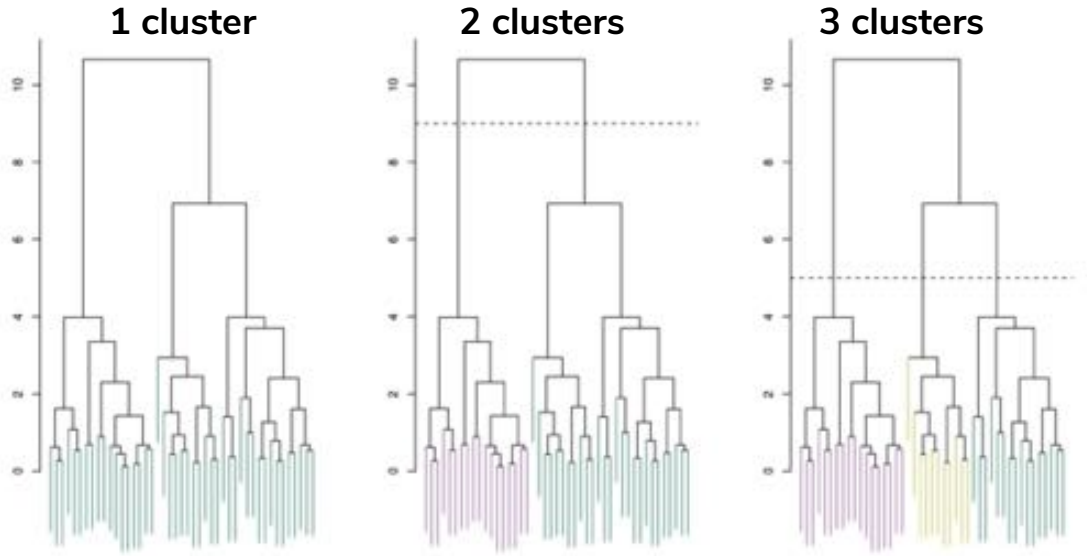




ISLR, figure 10.8 and 10.9

Choosing clusters

- To choose clusters we draw lines across the dendogram
- Number of clusters depends on where we draw the break point.



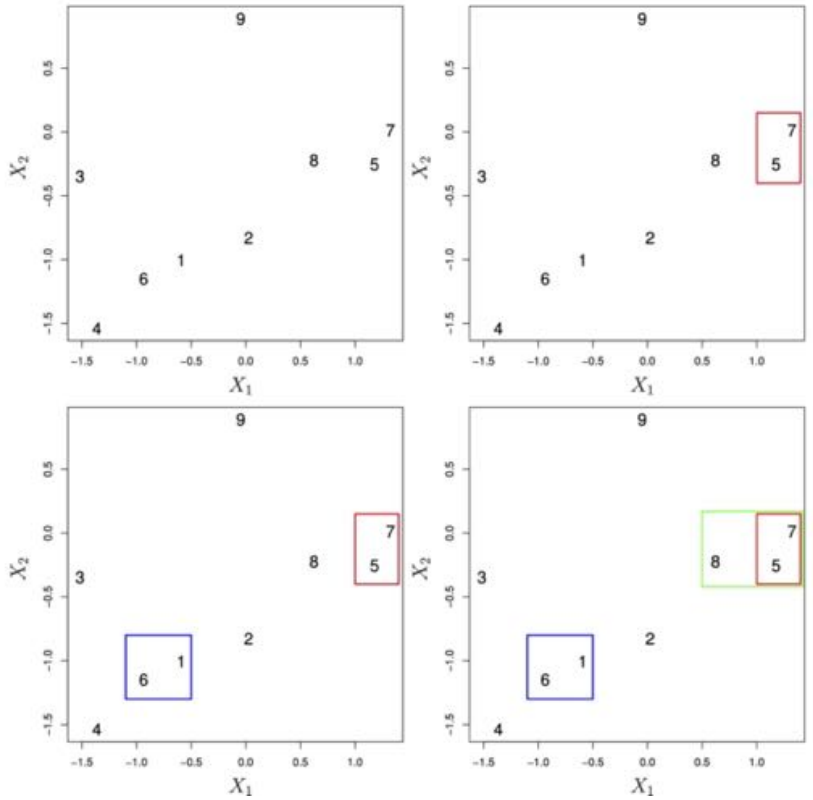
ISLR, figure 10.9

Hierarchical clustering algorithm: agglomerative approach

- Start with each point as a separate cluster (n clusters)
- Calculate a measure of dissimilarity between all points/ clusters
- Fuse two clusters that are most similar so that there are now n-1 clusters
- Fuse next two most similar clusters so there are now n-2 clusters
- Continue until there is only 1 cluster

Agglomerative clustering illustrated

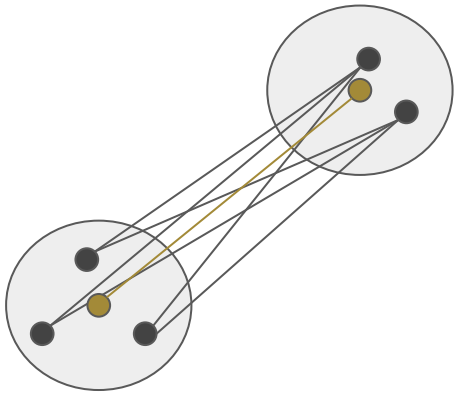
- Start with each observation as separate cluster
- Fuse 5 and 7
- Fuse 6 and 1
- Fuse the (5,7) cluster with 8
- Continue until all observations are fused



ISLR, figure 10.11

How do we define dissimilarity between clusters?

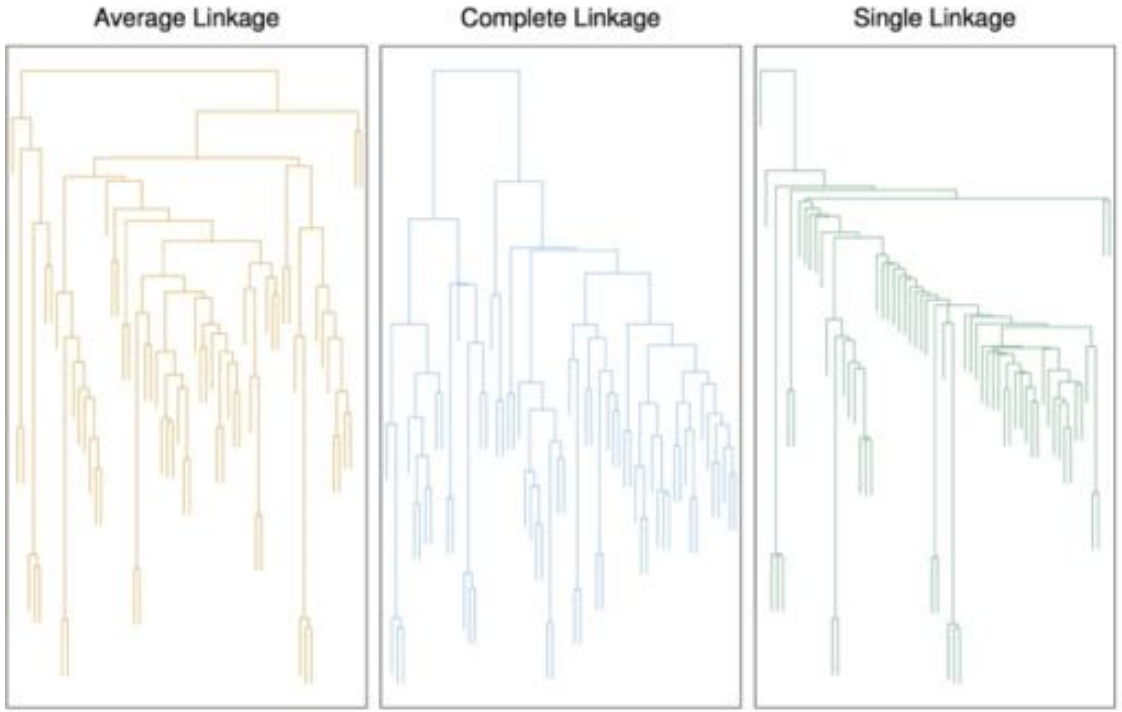
Linkage	Description
Complete	Maximal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations pairs of clusters. Record the ***largest.***
Single	Minimal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in pairs of clusters. Record the ***smallest.***
Average	Mean cluster dissimilarity. Compute all pairwise dissimilarities between observations in pairs of clusters. Record the **average**.
Centroid	Dissimilarity between the **centroids** of pairs of clusters.



ISLR, table 10.2

Linkage can be important

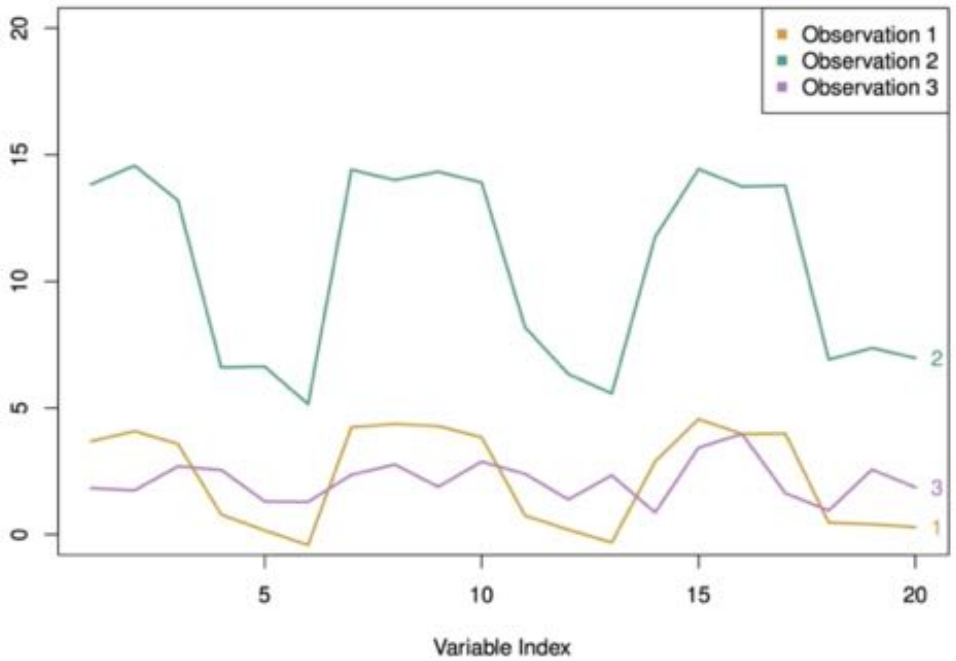
- Example with same data, different results
- Complete and average linkage tend to yield evenly sized clusters whereas single linkage tends to yield extended clusters to which single leaves are fused one by one



ISLR, figure 10.12

Choice of dissimilarity measure

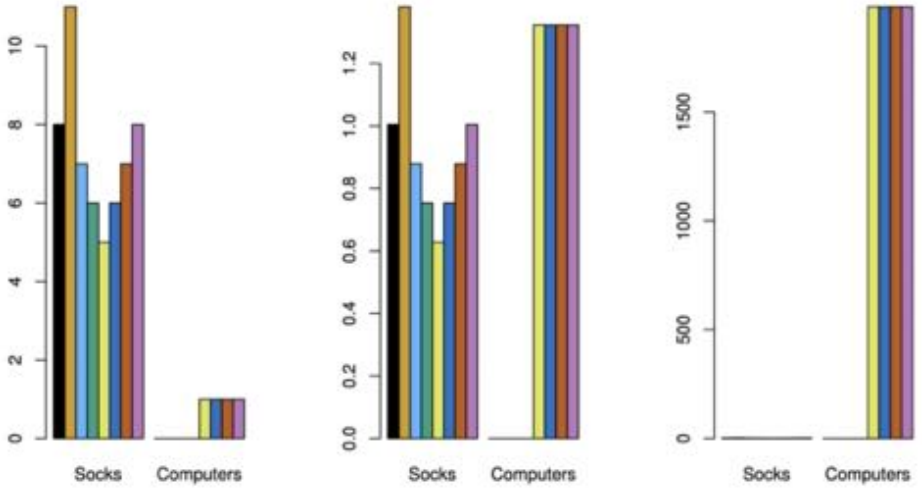
- Euclidean distance by default, but there are others like correlation-based distance
- For example:
  - Using Euclidean distance, observation 1 and 3 would be clustered
  - Using correlation-based distance, observation 1 and 2 would be clustered



ISLR, figure 10.13

Standardizing variables need to be considered, too

- Consider an online shop that sells two items: socks and computers
- Left: In terms of quantity, socks have higher weight
- Center: After standardizing, socks and computers have equal weight
- Right: In terms of expenditure (euros), computers have higher weight



ISLR, figure 10.14

Thoughts on clustering

Practical considerations for clustering (1 of 2)

- Should the features first be standardized? Preferred to center the variables centered with mean of zero and standard deviation of one
- In case of K-means clustering:
  - How many clusters should we look for the data?
- In case of hierarchical clustering:
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
  - Where should we cut the dendogram in order to obtain clusters?

In practice, we try several different choices, and look for the one with the most useful or interpretable solution. There is no single right answer!

Practical considerations for clustering (2 of 2)

- Most importantly, one must be careful about how the results of a clustering analysis are reported
- These results should not be taken as the absolute truth about a data set
- Rather, they should constitute a starting point for the developments of a scientific hypothesis and further study (exploratory data analysis), preferably on independent data (e.g. by using the test set)

Exercise

- Suppose that we have 5 observations, for which we compute a similarity (distance) matrix as shown
- On the basis of the similarity matrix, sketch the dendogram that results from hierarchically clustering these 5 observations using complete linkage.

	A	B	C	D	E
A	0				
B	9	0			
C	3	7	0		
D	6	5	9	0	
E	11	10	2	8	0