Chapter 10: Unsupervised learning

Yu-Tzung Chang and Hsuan-Wei Lee

Department of Political Science, National Taiwan University

2018.12.27.

Outline

- The challenge of unsupervised learning
- Principal component analysis
- Clustering methods
    - K-means clustering
    - Hierarchical clustering
    - Practical issues in clustering
- Example: NC160 Data

Supervised and unsupervised learning

- Most of this course focuses on *supervised learning* methods such as regression and classification.

- In that setting we observe both a set of features $X_1, X_2, \ldots, X_p$ for each object, as well as a response or outcome variable Y. The goal is then to predict Y using $X_1, X_2, \ldots, X_p$.

- Here we instead focus on *unsupervised learning*, we where observe only the features $X_1, X_2, \ldots, X_p$. We are not interested in prediction, because we do not have an associated response variable Y.

The goals of unsupervised learning

- The goal is to discover interesting things about the measurements: is there an informative way to visualize the data? Can we discover subgroups among the variables or among the observations?
- We discuss two methods:
  - *principal components analysis*, a tool used for data visualization or data pre-processing before supervised techniques are applied
  - *clustering*, a broad class of methods for discovering unknown subgroups in data.

The challenge of unsupervised learning

- Unsupervised learning is more subjective than supervised learning, as there is no simple goal for the analysis, such as prediction of a response.
- But techniques for unsupervised learning are growing importance in a number of fields:
  - subgroups of breast cancer patients groups by their gene expression measurements,
  - groups of shoppers characterized by their browsing and purchase histories,
  - movies grouped by the ratings assigned by movie viewers.

Another advantage

- It is often easier to obtain *unlabeled data* – from a lab instrument or a computer – than *labeled data*, which can require human intervention.
- For example it is difficult to automatically assess the overall sentiment of a movie review: is it favorite or not?

Principal components analysis

- PCA produces a low-dimensional representation of a dataset. It finds a sequence of linear combinations of the variables that have maximal variance, and are mutually uncorrelated.

- Apart from producing derived variables for use in supervised learning problems, PCA also serves as a tool for data visualization.

Principal components analysis: details

- The *first principal component* of a set of features $X_1, X_2, \ldots, X_p$ is the normalized linear combination of the features

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

  that has the largest variance. By *normalized*, we mean that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.

- We refer to the elements $\phi_{11}, \ldots, \phi_{p1}$ as the loadings of the first principal component; together, the loading make up the principal component loading vector,

  $$\phi_1 = (\phi_{11}\phi_{21}\ldots\phi_{p1})^T.$$

- We constrain the loadings so that their sum of squares is equal to one, since otherwise setting these elements to be arbitrarily large in absolute value could result in an arbitrarily large variance.

Principal components analysis: example
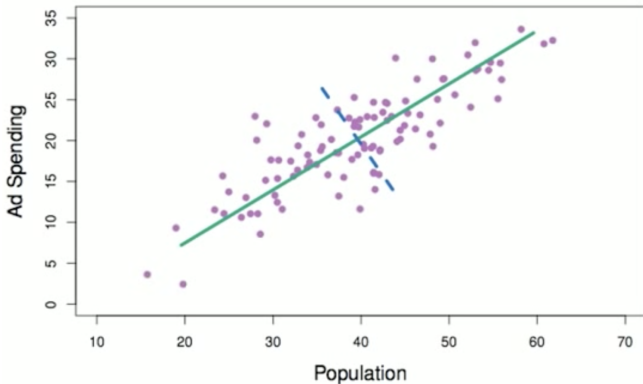
The population size (*normalized*) and ad spending (*ad*) for 100 different cities are shown as purple circles. The green solid line indicates the first principal component direction, and the blue dashed line indicates the second principal component direction.

Computation of principal components

- Suppose we have a $n \times p$ data set X. Since we are only interested in variance, we assume that each of the variables in X has been centered to have mean zero (that is, the column means of X are zero).

- We then look for the linear combination of the sample feature values of the form

$$z_{i1} = \phi_{11}x_{I1} + \phi_{21}x_{i2} + \cdots + \phi_{p1}x_{ip}$$

for $i = 1, \ldots, n$ that has largest sample variance, subject to the constraint that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.

- Since each of the x_{ij} has mean zero, then so does z_{i1} (for any values of ϕ_{j1}). Hence the sample variance of the z_{i1} can be written as $\frac{1}{n} \sum_{i=1}^{n} z_{i1}^2$.

Computation of principal components
(continued)

- Plugging in the first principal component equation loading vector solves the optimization problem

$$\text{maximize}_{\phi_{11},\ldots,\phi_{p1}} \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 \text{ subject to } \sum_{j=1}^{p} \phi_{j1}^2 = 1.$$

- This problem can be solved via a singular-value decomposition of the matrix X, a standard technique in linear algebra.

- We refer to Z_1 as the first principal component, with realized values $z_{11}, \ldots, z_{n1}$.

Geometry of PCA

- The loading vector ϕ_1 with elements $\phi_{11}, \ldots, \phi_{p1}$ defines a direction in feature space along which the data vary the most.

- If we project the n data points $x_1, \ldots, x_n$ onto this direction, the projected values are the principal component scores $z_{11}, \ldots, z_{n1}$ themselves.

Further principal components

- The second principal component is the linear combination of $X_1, \ldots, X_p$ that has maximal variance among all linear combinations that are *uncorrelated* with Z_1.

- The second principal component scores $z_{12}, z_{22}, \ldots, z_{n2}$ take the form

$$z_{i2} = \phi_{12}x_{i1} + \phi_{22}x_{i2} + \cdots + \phi_{p2}x_{ip},$$

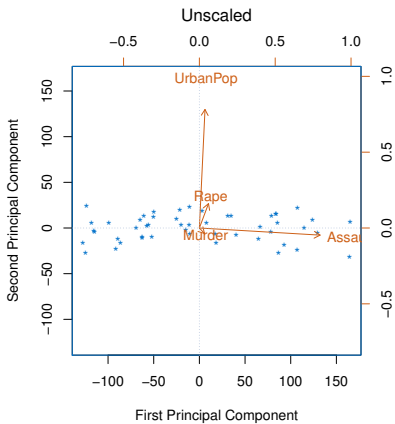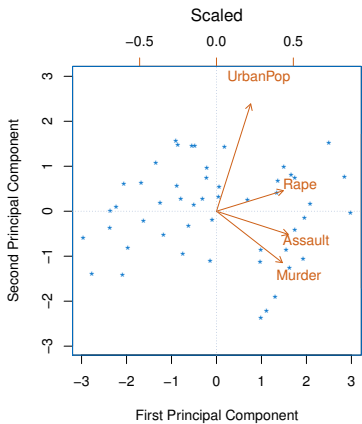where ϕ_2 is the second principal component loading vector with elements $\phi_{12}, \phi_{22}, \ldots, \phi_{p2}$.

Further principal components (continued)

- It turns out that constraining Z_2 to be uncorrelated with Z_1 is equivalent to constraining the direction ϕ_2 to be orthogonal (perpendicular) to the direction ϕ_1. And so on.
- The principal component directions $\phi_1, \phi_2, \phi_3, \ldots$ are the ordered sequence of right singular vectors of the matrix X, and the variances of the components are $\frac{1}{n}$ times the squares of the singular values. There are at most $\min(n-1, p)$ principal components.

Illustration

- *USAarrests* data: For each of the fifty states in the United States, the data set contains the number of arrests per 100,000 residents for each of three crimes: *assault*, *murder*, and *rape*. We also record *UrbanPop* (the percent of the population in each state living in urban areas).

- The principal components score vectors have length $n = 50$, and the principal component loading vectors have length $p = 4$.

- PCA was performed after standardizing each variable to have mean zero and standard deviation one.

USAarrests data: PCA plots

The first two principal components for the *USAarrests* data:

- The blue dots are the scores for the first two principal components of 50 states.

- The orange arrows indicate the first two principal component loading vectors (with axes on the top and right). For example, the loading for *rape* on the first component is 0.54, and its loading on the second principal component is 0.17 [the word *rape* is centered at the point (0.54, 0.17)].

- This figure is known as a *biplot*, because it displays both the principal component scores and the principal component loadings.

PCA loadings

|	PC1	PC2
Murder	0.5358995	-0.4181809
Assault	0.5831836	-0.1879856
UrbanPop	0.2781909	0.8728062
Rape	0.5434321	0.1673186

Another interpretation of principal components
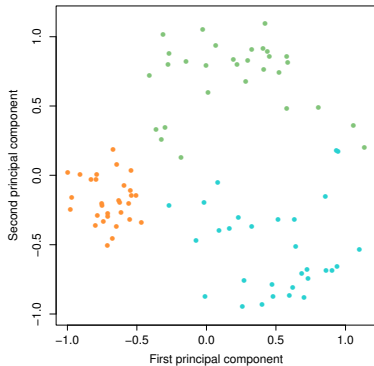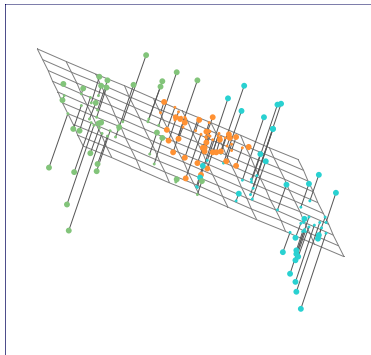
PCA finds the hyperplane closest to the observations

- The first principal component loading vector has a very special property: it defines the line in p-dimensional space that is *closest* to the n observations (using average squared Euclidean distance as a measure of closeness).

- The notion of principal components as the dimensions that are closet to the n observations extends beyond just the first principal component.

- For instance, the first two principal components of a data set span the plane that is closest to the n observations, in terms of average squared Euclidean distance.

Scaling of the variable matters

- If the variables are in different units, scaling each to have standard deviation equal to one is recommended.
- If they are in the same units, you might or might not scale the variables.
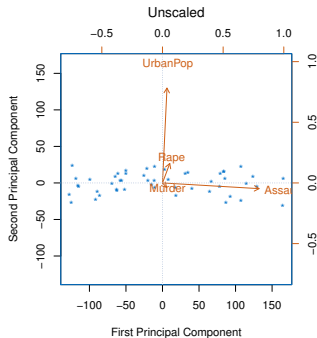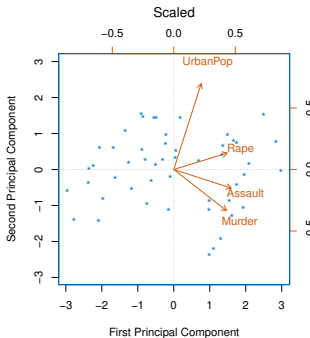
Proportion variance explained

- To understand the strength of each component, we are interested in knowing the proportion of variance explained (PVE) by each one.
- The *total variance* present in a data set (assuming that the variables have been centered to have mean zero) is defined as

$$\sum_{j=1}^{p} Var(X_j) = \sum_{j=1}^{p} \frac{1}{n} \sum_{i=1}^{n} x_{ij}^2,$$

and the variance explained by the mth principal component is

$$Var(Z_m) = \frac{1}{n} \sum_{i=1}^{n} z_{im}^2.$$

- It can be shown that
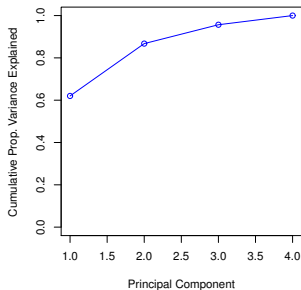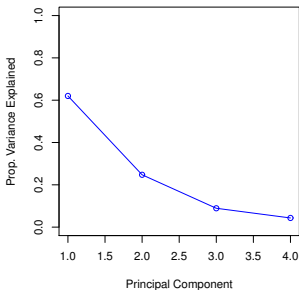
$$\sum_{j=1}^{p} Var(X_j) = \sum_{m=1}^{M} Var(X_m),$$

with $M = \min(n-1, p)$.

Proportion variance explained (continued)

- Therefore, the PVE of the mth principal component is given by the positive quantity between 0 and 1

$$\frac{\sum_{i=1}^{n} z_{im}^2}{\sum_{j=1}^{p} \sum_{i=1}^{n} x_{ij}^2}.$$

- The PVEs sum to one. We sometimes display the cumulative PVEs.

How many principal components should we use?

If we use principal components as a summary of our data, how many components are sufficient?

- No simple answer to this question, as cross-validation is not available for this purpose.
    - Why not?
    - When could we use cross-validation to select the number of components?

- The "scree plot" on the previous slide can be used as a guide: we look for an "elbow".

Clustering

- *Clustering* refers to a very broad set of techniques for finding *subgroups*, or *clusters*, in a data set.
- We seek a partition of the data into distinct groups so that the observations within each group are quite similar to each other.
- To make this concrete, we must define what it means for two or more observations to be *similar* or *different*.
- Indeed, this is often a domain-specific consideration that must be made based on knowledge of the data being studied.

PCA versus clustering

- PCA looks for a low-dimensional representation of the observations that explains a good fraction of the variance.
- Clustering looks for homogeneous subgroups among the observations.

Clustering for market segmentation

- Suppose we have access to a large number of measurements (e.g. median household income, occupation, distance from nearest urban area, and so forth) for a large number of people.

- Our goal is to perform *market segmentation* by identifying subgroups of people who might be more receptive to a particular form of advertising, or more likely to purchase a particular product.

- The task of performing market segmentation amounts to clustering the people in the data set.

Two clustering methods

- In *K-mean clustering*, we seek to partition the observations into a pre-specified number of observations.

- In *hierarchical clustering*, we do not know in advance how many clusters we want; in fact, we end up with a tree-like visual representation of the observations, called a *dendrogram*, that allows us to view at once the clusterings obtained for each possible number of clusters, from 1 to n.

K-means clustering
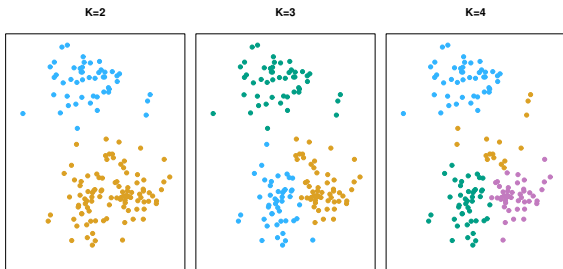
A simulated data set with 150 observations in 2-dimensional space. Panels show the results of applying *k*-means clustering with different values of *K*, the number of clusters. The color of each observation indicates the cluster to which it was assigned using the *K*-means clustering algorithm. Note that there is no ordering of the clusters, so the cluster coloring is arbitrary. These cluster labels were not used in clustering; instead, they are the outputs of the clustering procedure.

Details of K-means clustering

Let $C_1, \ldots, C_k$ denotes sets containing the indices of the observations in each cluster. These sets satisfy two properties:

1. $C_1 \cup C_2 \cup \cdots \cup C_K = \{1, \ldots, n\}$. In other words, each observation belongs to at least one of the K clusters.

2. $C_k \cap C_{k'} = \emptyset, \quad \forall k \neq k'$. In other words, the clusters are non-overlapping: no observation belongs to more than one cluster.

For instance, if the ith observation is in the kth cluster, then $i \in C_k$.

Details of K-means clustering (continued)

- The idea behind K-means clustering is that a *good* clustering is one for which the *within-cluster variation* is as small as possible.

- The within-cluster variation for cluster C_k is a measure $WCV(C_k)$ of the amount by which the observations within a cluster differ from each other.

- Hence we want to solve the problem

$$\text{minimize}_{C_1,\ldots,C_K} \left\{ \sum_{k=1}^{K} WCV(C_k) \right\}.$$

- In words, this formula says that we want to partition the observation into K clusters such that the total within-cluster variation, summed over all K clusters, is as small as possible.

How to define within-cluster variation?

- Typically we use Euclidean distance

$$WCV(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2,$$

  where $|C_k|$ denotes the number of observations in the kth cluster.

- Combining the previous two equations gives the optimization problem that defines K-means clustering,

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

K-means clustering algorithm

1. Randomly assign a number, from 1 to K, to each of the observations. These serve as initial cluster assignments for the observations.
2. Iterate until the cluster assignments stop changing:
   1. For each of the K clusters, compute the cluster *centroid*. The kth cluster centroid is the vector of the p feature means for the observations in the kth cluster.
   2. Assign each observation to the cluster whose centroid is closest (where *closest* is defined using Euclidean distance).

K-means clustering example
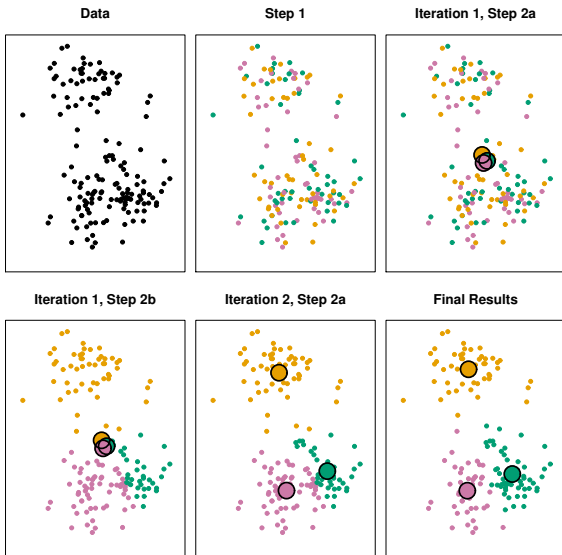
Properties of the K-means algorithm

- This algorithm is guaranteed to decrease the value of the objective function at each step. Note that

$$\frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = 2 \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{ij})^2,$$

where $\bar{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$ is the mean for feature j in cluster C_k.

- However, it is not guaranteed to give the global minimum. Need random initialization!

K-means clustering example (different starting values)

Hierarchical clustering

- K-means clustering requires us to pre-specify the number of clusters K. This can be a disadvantage (later we discuss strategies for choosing K).

- *Hierarchical clustering* is an alternative approach which does not require that we commit to a particular choice of K.

- In this section, we describe *bottom-up* or *agglomerative* clustering. This is the most common type of hierarchical clustering, and refers to the fact that a dendrogram is built starting from the leaves and combining clusters up to the trunk.
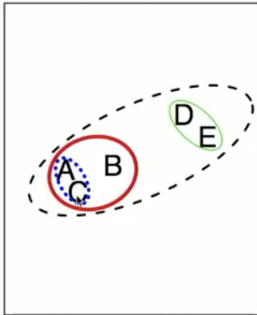
Hierarchical clustering: illustration
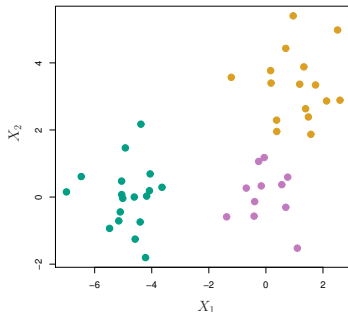
Builds a hierarchy in a "bottom-up" fashion.
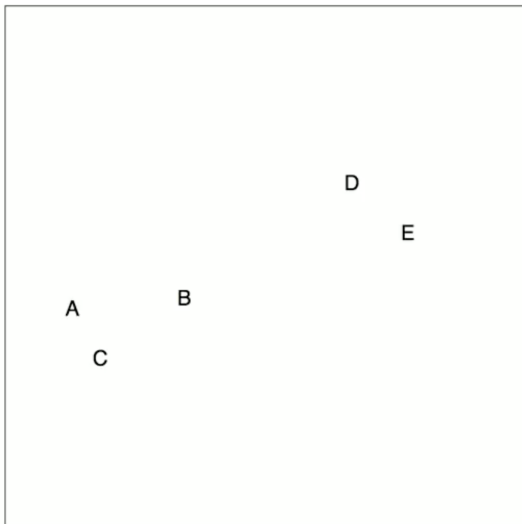
Hierarchical clustering algorithm

The approach in words:

- Start with each point in its own cluster.
- Identify the closet two clusters and merge them.
- Repeat.
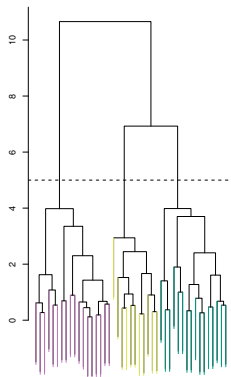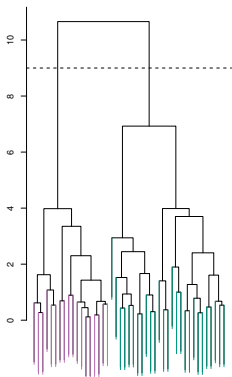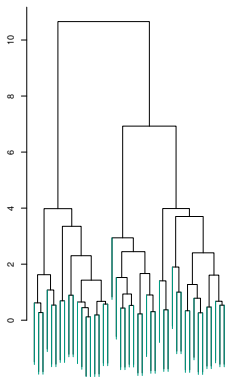- Ends when all points are in a single cluster.
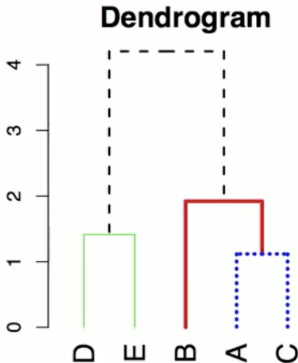
45 observations generated in 2-dimensional space. In reality there are three distinct classes, shown in separate colors. However, we will treat these class labels as unknown and will seek to cluster the observations in order to discover the classes from the data.

An example using hierarchical clustering (continued)

Details of the previous figure

- Left: Dendrogram obtained from hierarchically clustering the data from previous slide, with complete linkage and Euclidean distance.

- Center: The dendrogram from the left-hand panel, cut at a height of 9 (indicated by the dashed line). This cut results in two distinct clusters, shown in different colors.

- Right: The dendrogram from the left-hand panel, now cut at a height of 5. This cut results in three distinct clusters, shown in different colors. Note that the colors were not used in clustering, but are simply used for display purposes in this figure.

Types of linkage

Linkage	Description
Complete	Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities.
Average	Mean inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

Choice of dissimilarity measure

- So far we have used the Euclidean distance.
- An alternative is *correlation-based distance* which considers two observations to be similar if their features are highly correlated.
- This is an unusual use of correlation, which is normally computed between variables; here it is computed between the observation profiles for each pair of observations.
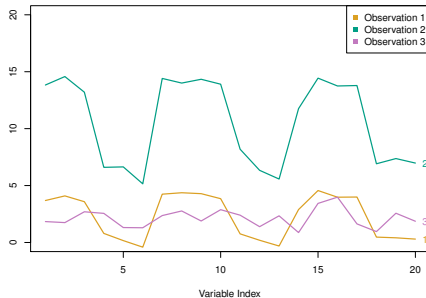
Practical issues

- *Scaling of the variables matters!* Should the observations or features first be standardized in some way? For instance, maybe the variables should be centered to have mean zero and scaled to have standard deviation one.
- In the case of hierarchical clustering
    - What dissimilarity measure should be used?
    - What type of linkage should be used?
- How many clusters to choose? (in both K-means and hierarchical clustering). Difficult problem. No agreed-upon method. See Elements of Statistical Learning, chapter 13 for more details.
- Which features should we use to drive the clustering?

Example: breast cancer microarray study

- "Repeated observations of breast tumor subtypes in independent gene expression data sets;" Sorlie at el, PNAS 2003.

- Gene expression measurements for about ~ 8000 genes, for each of 88 breast cancer patients.

- Average linkage, correlation metric.

- Clustered samples using 500 *intrinsic genes*: each woman was measured before and after chemotherapy. Intrinsic genes have smallest within/between variation.

Results: breast cancer microarray study

Results: breast cancer microarray study

B Norway/Stanford data set
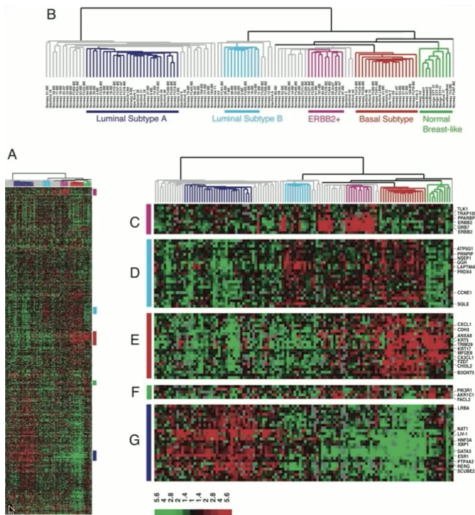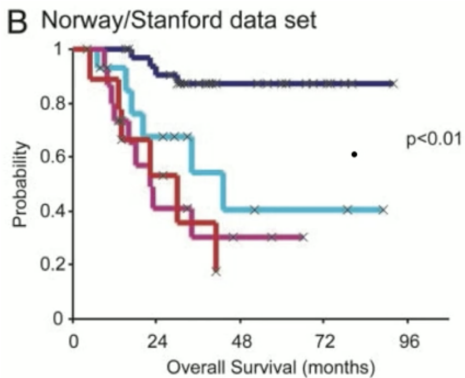
Conclusions

- *Unsupervised learning* is important for understanding the variation and grouping structure of a set of unlabeled data, and can be a useful pre-processor for supervised learning.

- It is intrinsically more difficult than *supervised learning* because there is no gold standard (like an outcome variable) and no single objective (like test set accuracy).

- It is an active field of research, with many recently developed tools such as *self-organizing maps*, *independent components analysis* and *spectral clustering*. See The Elements of Statistical Learning for details.