Cluster analysis

Associated with each object is a set of G measurements w the **feature vector**, $\mathbf{X} = (X_1, \ldots, X_G)$. The feature vect belongs to a feature space $\mathcal{X}$ (e.g. $\Re^G$).

The task is to identify groups of *similar* objects on the ba set of feature vectors, $\mathbf{X}_1 = \mathbf{x}_1, \ldots, \mathbf{X}_n = \mathbf{x}_n$.

Clustering involves several distinct steps. First, a suitable between objects (based on the features) must be defined. clustering algorithm must be selected and applied to the c data. The results of a clustering procedure can include bo number of clusters K (if not prespecified) and a set of n c labels $\in \{1, \ldots, K\}$ for the objects.

Cluster analysis

Clustering is probably a more difficult problem than class
In general, all the issues that must be addressed for classi
must also be addressed for clustering.

With clustering there is generally no *a priori* notion of wh
features are important.

Often the number of clusters is unknown as well.

Additionally, the goals can be quite vague: *Find some int*
and important clusters in my data.

Most of the algorithms that are appealing are computatio
complex to have exact solutions. Approximate solutions a
instead and reproducibility becomes an issue.

Cluster analysis

Clustering algorithms fall into two broad categories, **hiera** **methods** and **partitioning methods**.

Hierarchical methods are either **divisive** or **agglomerati** methods provide a hierarchy of clusters, from the smallest all objects are in one cluster, through to the largest set, w observation is in its own cluster.

Most methods used in practice are agglomerative hierarch methods. In large part this is due to the fact that efficien algorithms exist for performing these calculations.

Partitioning methods usually require the specification of t number of clusters. Then, cluster centers must be determi finally a mechanism for apportioning objects to the cluste

Distance

The feature data are often transformed to an $n \times n$ **dista**
similarity matrix, $\mathbf{D} = (d_{ij})$, between the n objects.

One of the most important factors that determines which
will be found is the choice of distance between objects.

Once a distance measure between individual observations
chosen, one must often also define a distance measure bet
clusters or groups of observations

Different choices here can greatly affect the outcome.

More details in the lecture *Distances and expression meas*

Gene expression data

Most efforts to date have involved clustering only the exp
data collected on a number of different genes and samples

However, there is likely to be a need for incorporating oth
such as sample level covariates into the algorithm.

For example, a common task is to determine whether or n
expression data can reliably identify or classify different ty
disease. However, one might ask as well whether such dat
our ability to classify over already available sample level c
data.

Gene expression data

Gene expression data on G genes (features) for n mRNA (observations)

$$X_{G\times n} = \begin{array}{cc} \text{mRNA samples} & \\ \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1n} \\ x_{21} & x_{22} & \ldots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{G1} & x_{G2} & \ldots & x_{Gn} \end{bmatrix} & \text{Genes} \end{array}$$

x_{gi} = expression measure for gene g in mRNA samp

An array of conormalized arrays.

Gene expression data

Features correspond to expression levels of different genes correspond to, for e.g., tumor types (e.g. ALL, AML), cli outcomes (survival, non–survival), and are labeled by $\{1,$

Gene expression data on G genes (features) for n mRNA (observations)

$$\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iG})$$

– gene expression profile / feature vector for s

$$y_i = \text{response for sample } i, \qquad i = 1, \ldots, n.$$

Other covariates such as age, sex may also be important and included in the analysis. However, it is worth noting that the distance should reflect the covariates being used (e.g. the Eucl distance is generally not suitable for categorical variables).

Clustering gene expression data

- One can cluster genes and/or samples (arrays).
- Clustering leads to readily interpretable figures.
- Clustering strengthens the signal when averages are ta within clusters of genes (Eisen et al., 1998).
- Clustering can be helpful for identifying gene expressi patterns in time or space.
- Clustering is useful, perhaps essential, when seeking n subclasses of cell samples (tumors, etc).

Clustering gene expression data

Cluster genes (rows)

- to identify groups of co–regulated genes, e.g. using la numbers of yeast experiments;
- to identify spatial or temporal expression patterns;
- to reduce redundancy (cf. feature selection) in predict models;
- for display purposes.

Transformations of the expression data matrix using linea modeling as in the lecture *Microarray experimental design analysis* may be useful in this context:

$$\text{genes} \times \text{arrays} \Longrightarrow \text{genes} \times \text{estimated effects.}$$

Clustering gene expression data

Cluster samples or arrays (columns)

- to identify new classes of biological samples, e.g. new classes, new cell types;
- to detect experimental artifacts;
- for display purposes.

Cluster both rows and columns at once.

Clustering gene expression data

Clustering can be gainfully employed in an exploratory m
The clusters that obtain from clustering samples/arrays sl
compared with different experimental conditions such as:

- batch or production order of the arrays;
- batch of reagents;
- technician;
- order.

Any relationships observed here should be considered as a
potentially serious source of bias.

Tumor classification using gene expression da

A reliable and precise classification of tumors is essential f
successful diagnosis and treatment of cancer.

Current methods for classifying human malignancies rely
variety of morphological, clinical, and molecular variables.

In spite of recent progress, there are still uncertainties in d

Also, it is likely that the existing classes are heterogeneou
comprise diseases which are molecularly distinct and follo
different clinical courses.

Tumor classification using gene expression da

DNA microarrays may be used to characterize the molecu
variations among tumors by monitoring gene expression p
a genomic scale.

This may lead to a finer and more reliable classification o
and to the identification of marker genes that distinguish
these classes.

Eventual clinical implications include an improved ability
understand and predict cancer survival.

Tumor classification using gene expression da

There are three main types of statistical problems associa
tumor classification:

1. the identification of new tumor classes using gene exp profiles – **unsupervised learning**;
2. the classification of malignancies into known classes – **supervised learning**;
3. the identification of marker genes that characterize th different tumor classes – **feature selection**.

Example: Row and column clustering

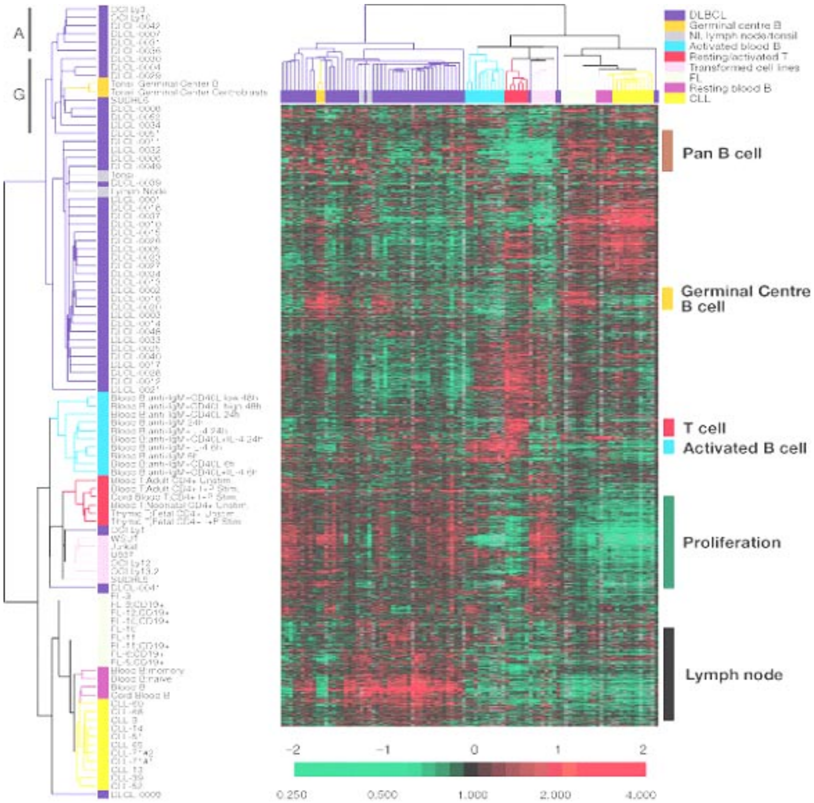

Figure 1: Alizadeh et al. (2000). Distinct types of dif
B–cell lymphoma identified by gene expression profiling.

Clustering gene expression data

Preliminary questions

- Which genes / arrays to use?
- Which transformation/standardization?
- Which distance function?
- Which clustering algorithm?

Answers will depend on the biological problem.

Clustering gene expression data

Important questions (which are generic)

- How many clusters?
- How reliable are the clustering results?
  - Statistical inference: distributional properties of cl results.
  - Assessing the strength/confidence of cluster assign individual observations;
  - Assessing cluster homogeneity.

Partitioning methods

- Partition the data into a **prespecified** number K of r exclusive and exhaustive groups.
- Iteratively reallocate the observations to clusters until criterion is met, e.g. minimize within–cluster sums–of
- Examples:
  - k–means; fuzzy k–means;
  - Partitioning Around Medoids – PAM (Kaufman & Rousseeuw, 1990);
  - Self–Organizing Maps – SOM (Kohonen, 2001);
  - model–based clustering,
    e.g. Gaussian mixtures in Fraley & Raftery (1998,2 McLachlan et al. (2001).

Partitioning around medoids

Partitioning around medoids or **PAM** of Kaufman ar Rousseeuw (1990) is a partitioning method which operate distance matrix, e.g. Euclidean distance matrix.

For a prespecified number of clusters K, the PAM proced based on the search for K representative objects, or **med** among the observations to be clustered.

After finding a set of K medoids, K clusters are construct assigning each observation to the nearest medoid.

Partitioning around medoids

The goal is to find K medoids, $\mathbf{M} = (\mathbf{m}_1, \ldots, \mathbf{m}_K)$, which minimize the sum of the distances of the observations to t closest medoid, that is,

$$\mathbf{M}^* = \mathrm{argmin}_{\mathbf{M}} \sum_i \min_k d(\mathbf{x}_i, \mathbf{m}_k).$$

PAM can be applied to general data types and tends to b robust than k–means.

Silhouette plots

Rousseeuw (1987) suggested a graphical display, the **silho plot**, which can be used to: (i) select the number of clust (ii) assess how well individual observations are clustered.

The **silhouette width** of observation i is defined as

$$sil_i = (b_i - a_i)/\max(a_i, b_i),$$

where a_i denotes the average distance between i and all o observations in the cluster to which i belongs, and b_i deno minimum average distance of i to objects in other clusters

Intuitively, objects with large silhouette width sil_i are well–clustered, those with small sil_i tend to lie between cl

Silhouette plots

For a given number of clusters K, the overall **average sil**
width for the clustering is simply the average of sil_i over
observations i, $\bar{sil} = \sum_i sil_i/n$.

Kaufman & Rousseeuw suggest estimating the number of
K by that which gives the largest average silhouette widt

Note that silhouette widths may be computed for the resu
partitioning clustering algorithm.

Partitioning around medoids

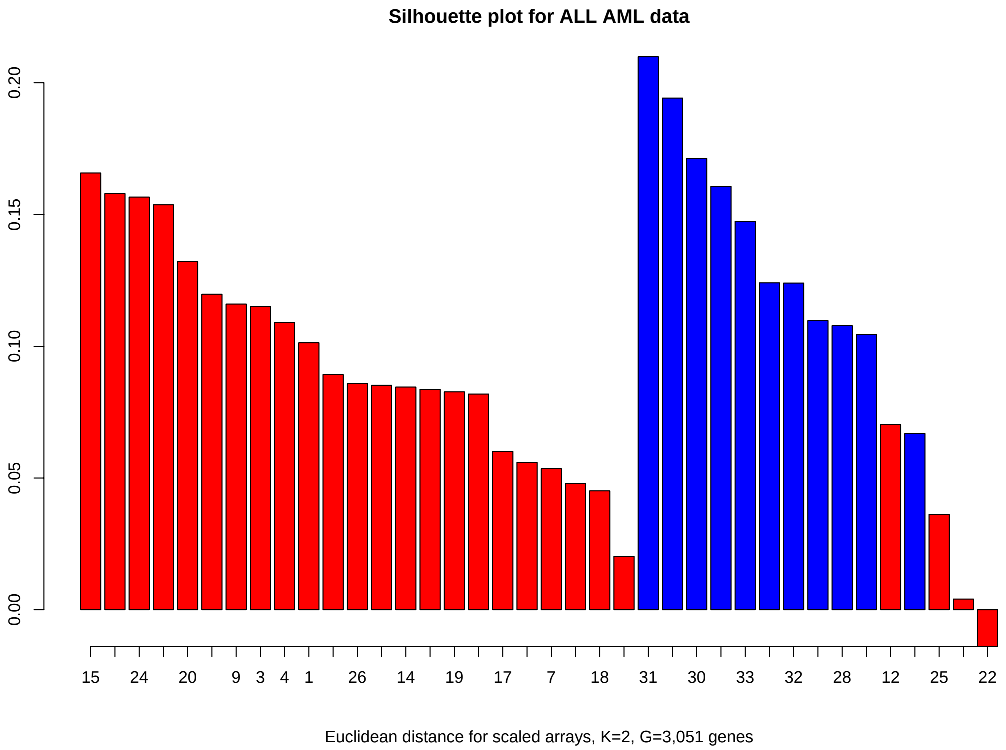

Figure 2: Golub et al. (1999) ALL AML data. Silhouett
PAM, red=ALL, blue=AML.

PAMSIL

PAMSIL. van der Laan, Pollard, & Bryan (2001).

Replace PAM criteria function with average silhouette.

	PAM	PAMS
Criteria	$-\sum_i \min_k d(\mathbf{x}_i, \mathbf{m}_k)$	$\sum_i si$
Algorithm	Steepest ascent	Steepest a
Starting values	Build	PAM, ran
K	Given or data–adaptive	Given or data
Overall performance	”Robust”	”Efficie
Splitting large clusters	Yes	No
Outliers	Ignore	Identi

Hierarchical methods

- Hierarchical clustering methods produce a **tree** or **dendrogram**.
- They avoid specifying how many clusters are appropr providing a partition for each K. The partitions are o from cutting the tree at different levels.
- The tree can be built in two distinct ways
  - bottom–up: **agglomerative** clustering;
  - top–down: **divisive** clustering.

Hierarchical methods

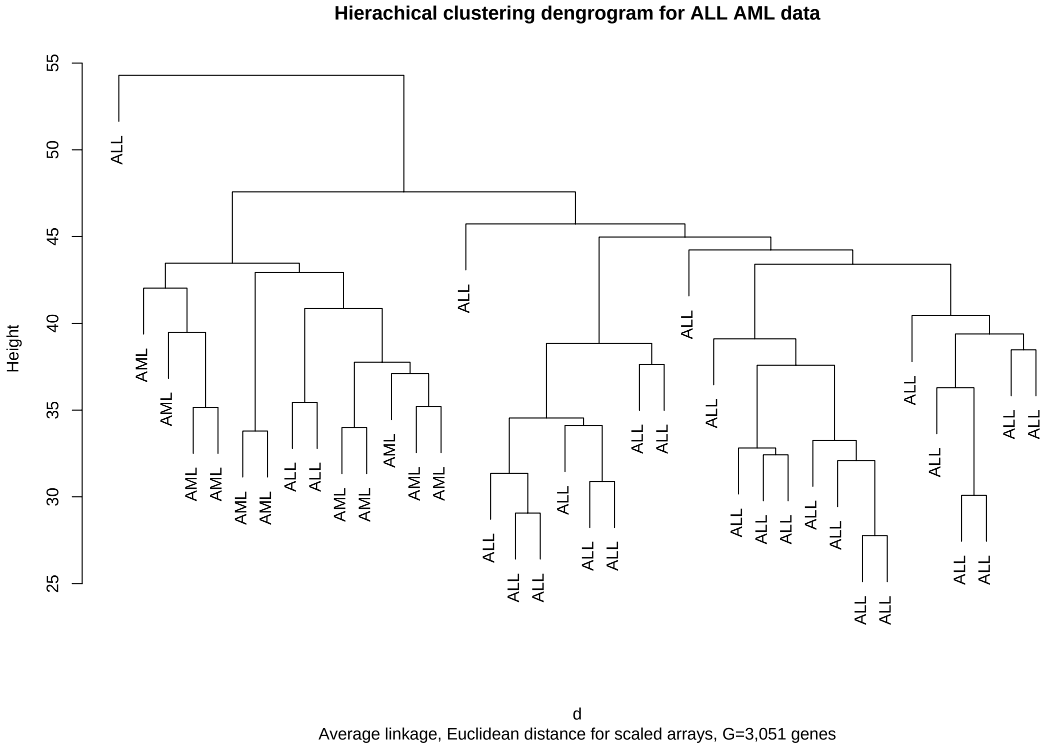

Figure 3: Golub et al. (1999) ALL AML data. Dendr
agglomerative hierarchical clustering.

Agglomerative methods

- Start with n mRNA sample (or G gene) clusters.
- At each step, merge the two closest clusters using a m between–cluster distance which reflects the shape of t clusters.
- Between–cluster distance measures:
  - *Unweighted Pair Group Method with Arithmetic m (UPGMA)*: average of pairwise distances;
  - *Single–link*: minimum of pairwise distances;
  - *Complete–link*: maximum of pairwise distances.

More details are given in the lecture *Distances and expres measures.*

Divisive methods

- Start with only one cluster.
- At each step, split clusters into two parts.
- Advantages: Obtain the main structure of the data, i.
  on upper levels of dendrogram.
- Disadvantages: Computational difficulties when consi
  possible divisions into two groups.
- Examples
  - Self–Organizing Tree Algorithm – SOTA (Dopazo
    Carazo, 1997);
  - DIvisive ANAlysis – DIANA (Kaufman & Roussee
    1990).

Dendrograms

Dendrograms are often used to visualize the output of a hierarchical clustering.

However, they can be criticized on a number of grounds.

Good graphics reveal structure that might not be found b standard analytic methods.

Hierarchical clustering imposes structure, whether it is th not. Dendrograms then reflect that imposed structure.

It will be important to determine whether the dendrogran reasonable reflection of the structure in the data.

Dendrograms

The **cophenetic distance** between two observations, i ar
defined to be the intergroup distance at which observation
are first put into the same cluster.

These distances have a great deal of structure, there are m
and some other structure.

The extent to which the cophenetic distances reflect the t
distances (as decided by our choice of metric) determines
usefulness of the dendrogram as a tool for visualization.

The agreement can be assessed by the **cophenetic corre**
coefficient which is simply the correlation between the tr
distances and the cophenetic distances.

Partitioning vs. hierarchical

- **Partitioning**
  - Advantages: Provides clusters that satisfy an optin criterion (approximately).
  - Disadvantages: Need initial K, long computation t
- **Hierarchical**
  - Advantages: Fast computation (for agglomerative clustering).
  - Disadvantages: Rigid, cannot correct later for erro decisions made earlier.

Estimating the number of clusters

- **Internal indices.** Statistics based on within– and between–clusters matrices of sums–of–squares and cross–products (30 methods reviewed in Milligan & C (1985)). Estimate is the number of clusters K which or maximizes one of these indices.
- **Average silhouette width.** (Kaufman & Rousseeuw
- **Model–based methods.** EM algorithm for Gaussia mixtures, Fraley & Raftery (1998,2000) and McLachla (2001).
- **Gap statistic.** (Tibshirani et al., 2001). Resampling for each K compare an observed internal index to its value under a reference distribution and look for K w maximizes the difference.

MSS

Mean Silhouette Split – MSS. (Pollard & van der La

Given K clusters, consider each cluster $k = 1, \ldots, K$ sepa

- Apply the clustering algorithm to the elements of clus
- Choose the number of child clusters that maximizes t average silhouette width. Call this maximum the **spli** **silhouette**, SS_k.

Define the **mean split silhouette** as a measure of averag heterogeneity.

$$MSS(K) = \frac{1}{K} \sum_{k=1}^{K} SS_k.$$

Choose the number of clusters K which minimizes $MSS($

MSS

- Identifies finer structure in gene expression data. Whe clustering genes, existing criteria tend to identify glob structure only.
- Provides a measure of cluster heterogeneity.
- Computationally easy.

Clest

Clest. (Dudoit & Fridlyand 2001). Resampling method w
estimates the number of clusters based on prediction accu

- For each number of clusters k, repeatedly randomly d
original learning set into two non–overlapping sets, a
set $\mathcal{L}^b$ and a test set $\mathcal{T}^b$, $b = 1, \ldots, B$.
  - Apply the clustering algorithm to observations in the l
$\mathcal{L}^b$.
  - Build a classifier using the class labels from the cluster
  - Apply the classifier to the test set $\mathcal{T}^b$.
  - Compute a similarity score $s_{k,b}$ comparing the test set
labels from prediction and clustering.

Clest

- The similarity score for k clusters is the median of the similarity scores: $t_k = \text{median}(s_{k,1}, \cdots, s_{k,B})$.
- The number of clusters K is estimated by comparing observed similarity score t_k for each k to its expected under a suitable reference distribution with $K = 1$.

Applies to any partitioning algorithm and any classifier.

Better suited for clustering samples than clustering genes.

Inference

van der Laan & Bryan (2001).

General framework for statistical inference in cluster anal

View clustering as a deterministic rule that can be applie parameters (or estimates thereof) of the distribution of ge expression measures.

Parameters of interest include covariances between the ex measures of different genes.

The parametric bootstrap can be used to study distributi properties (bias, variance) of the clustering results.

Outliers

In classification it has often been found useful to define a
outliers.

This does not seem to have been extended to clustering. I
outlier detection is an important issue since outliers can g
affect the between–cluster distances.

Simple tests for outliers, such as identifying observations t
responsible for a disproportionate amount of the within–c
sum–of–squares seems prudent.

Hybrid method – HOPACH

Hierarchical Ordered Partitioning And Collapsing – HOPACH (van der Laan & Pollard, 2001)

- Apply a partitioning algorithm iteratively to produce hierarchical tree of clusters.
- At each node, a cluster is partitioned into two or mor clusters. Splits are not restricted to be binary. E.g. cl based on average silhouette.

Hybrid method – HOPACH

- **Hierarchical.** Can look at clusters at increasing levels of

- **Ordered.** Ordering of the clusters and elements within cl data–adaptive and unique, performing better than other h algorithms. Clustering and ordering are based on the sam function. The ordering of elements in any level can be use reorder the data or distance matrices, and visualize the cl structure.

- **Partitioning.** At each node, a cluster is split into two or smaller clusters.

- **Collapsing.** Clusters can be collapsed at any level of the similar clusters and correct for errors made in the partitio

- **Hybrid.** Combines the strengths of both partitioning and hierarchical clustering methods.

Bagged clustering

Leisch (1999). Hybrid method combining partitioning a
hierarchical methods. A partitioning method is applied to
bootstrap learning sets and the resulting partitions are co
by performing hierarchical clustering of the cluster center

Dudoit & Fridlyand (2001). Apply a partitioning clus
method to bootstrap samples of the learning set. Combin
resulting partitions by (i) voting or (ii) the creation of a n
distance matrix. Assess confidence in the clustering result
cluster votes.

R clustering software

- `class` package: Self Organizing Maps (`SOM`).
- `cluster` package:
  - AGglomerative NESting (`agnes`),
  - Clustering LARe Applications (`clara`),
  - DIvisive ANAlysis (`diana`),
  - Fuzzy Analysis (`fanny`),
  - MONothetic Analysis (`mona`),
  - Partitioning Around Medoids (`pam`).
- `e1071` package:
  - Fuzzy C–means clustering (`cmeans`),
  - Bagged clustering (`bclust`).
- `mva` package:
  - Hierarchical clustering (`hclust`),
  - k–means (`kmeans`).

Specialized summary, plot, and print methods for clustering results.

Acknowledgments

- **Brown Lab**, Biochemistry, Stanford.
- **Sabina Chiaretti**, Dana Farber Cancer Institute.
- **Jane Fridlyand**, UCSF Cancer Center.
- **Mark van der Laan**, Biostatistics, UC Berkeley.
- **Katie Pollard**, Biostatistics,UC Berkeley.
- **Yee Hwa (Jean) Yang**, Statistics, UC Berkeley.