Unsupervised Learning: Clustering

Clustering

Objective:

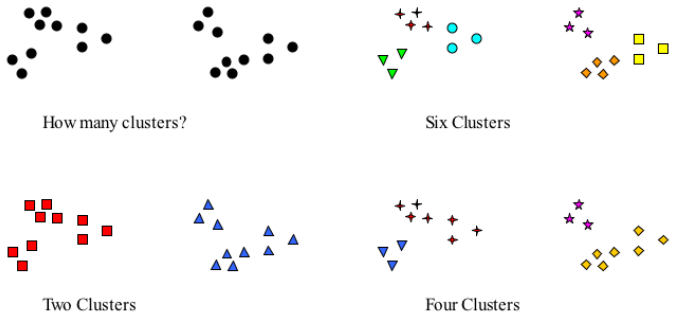
- Definition: Group or segment the data set (a collection of objects) into subsets so that those within each subset are more closely related to others than those objects assigned to other subsets.
- Each group (subset) is called a cluster.

Challenging:

- What is a meaningful cluster?
- How do we validate clustering results?

Clustering Challenges

What are meaningful clusters?



Clustering Concepts:

- Hard vs. Soft Clustering.
- Model-Based vs. Algorithmic.
- Flat vs. Nested.
- Clustering observations (most common) vs. Clustering features vs. Clustering both (Biclustering).

Proximity and Dissimilarity Matrices

Clustering results are crucially dependent on the measure of dissimilarity (or distance) between the "points" to be clustered.

- Proximity Matrix: $n \times n$ with the ij-th element d_{ij} measuring the proximity between the i-th and the jth objects (or observations). D is typically symmetric.
- One can use a dissimilarity matrix instead.
- Dissimilarity between $\mathbf{x}_i$ and $\mathbf{x}_{i'}$:

$$D(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p} d_j(x_{ij}, x_{i'j}).$$

- A weighted version:

$$D(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p} w_j d_j(x_{ij}, x_{i'j}); \sum_{j=1}^{p} w_j = 1.$$

Types of Distances:

- Squared distance: $d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2$.
- A more general distance: $d(\mathbf{x}_i, \mathbf{x}_{i'}) = l(|\mathbf{x}_i - \mathbf{x}_{i'}|)$.
- Correlation:

$$\rho(\mathbf{x}_i, \mathbf{x}_{i'}) = \frac{\sum_j (x_{ij} - \bar{x}_i)(x_{i'j} - \bar{x}_{i'})}{\sqrt{\sum_j (x_{ij} - \bar{x}_i)^2 \sum_j (x_{i'j} - \bar{x}_{i'})^2}}$$

- If inputs are standardized, $\sum_j (x_{ij} - x_{i'j})^2 \propto 2(1 - \rho(\mathbf{x}_i, \mathbf{x}_{i'}))$: clustering based on correlation (similarity) is equivalent to that based on squared distance (dissimilarity).
- Others? For Discrete Data? (`dist()` in R).

Clustering Algorithms:

- K-means.
  - Combinatorial algorithms.
  - NMF for soft-clustering.
  - Model-based soft-clustering.
- Hierarchical Clustering.
  - Biclustering - Cluster-Heatmap.
  - Convex Clustering & Convex Biclustering.

Combinatorial Algorithms

- Each observation is uniquely labeled by an integer $i \in \{1, 2, \ldots, n\}$.
- k clusters: $k \in \{1, \ldots, K\}$.
- Let $k = C(i)$ denote the ith observation get assigned to the k-th cluster.
- $d(\mathbf{x}_i, \mathbf{x}_{i'})$: dissimilarity between $\mathbf{x}_i$ and $\mathbf{x}_{i'}$.
- Goal: search C^* such that $W(C)$ (within cluster dissimilarity) is minimized:

$$W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} d(\mathbf{x}_i, \mathbf{x}_{i'}).$$

Combinatorial Algorithms

- Total Dissimilarity:

$$T = \frac{1}{2}\sum_{i=1}^{n}\sum_{i'=1}^{n} d_{ii'} = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\Big(\sum_{C(i')=k} d_{ii'} + \sum_{C(i')\neq k} d_{ii'}\Big).$$

- $$B(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(i')\neq k} d_{ii'}.$$

- $W(C) = T - B(C)$.
- Minimizing $W(C)$ is equivalent to maximizing $B(C)$.

Combinatorial Algorithms

- One needs to minimize W over all possible assignments of n points to K clusters.
- The number of distinct assignments is

$$S(n, K) = \frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\binom{K}{k}k^n.$$

- It is not feasible for large n and K.
- It calls for more efficient algorithms: may not be optimal but a reasonably good suboptimal partition.

K-Means Clustering

K-means

- One of the most popular iterative descent clustering methods.
- Tries to find a fast, local solution to the combinatorial clustering problem.
- For the case that all variables are quantitative.
- Dissimilarity measure: Squared Euclidean distance

$$d(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 = ||\,\mathbf{x}_i - \mathbf{x}_{i'}\,||^2.$$

K-means

Minimizing $W(C)$:

$$W(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{i\neq i', C(i')=k} ||\mathbf{x}_i - \mathbf{x}_{i'}||_2^2$$

$$= \sum_{k=1}^{K} n_k \sum_{C(i)=k} ||\mathbf{x}_i - \bar{\mathbf{x}}_k||_2^2$$

where $\bar{\mathbf{x}}_k = \frac{1}{n_k}\sum_{C(i)=k}\mathbf{x}_i$.

Idea - Split problem into two parts:

1. Cluster means. ($\mathbf{m}_k = \bar{\mathbf{x}}_k$)
2. Cluster assignments. ($C(i)$)

K-means Algorithm

- Initialize each observation i to a cluster assignment k.
- Repeat until cluster assignments are unchanged:

1. Find cluster means. (cluster assignments fixed)

$$\hat{\mathbf{m}}_k = \text{argmin}_m \sum_{C(i)=k} ||\mathbf{x}_i - \mathbf{m}_k||_2^2$$

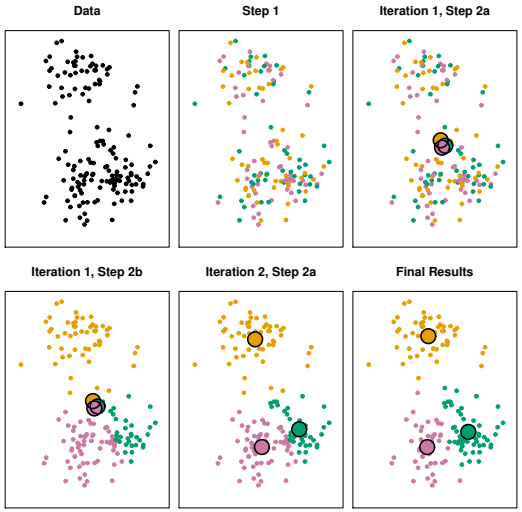
$\Longrightarrow$ Take mean of points in cluster.

2. Find cluster assignments. (cluster means fixed)

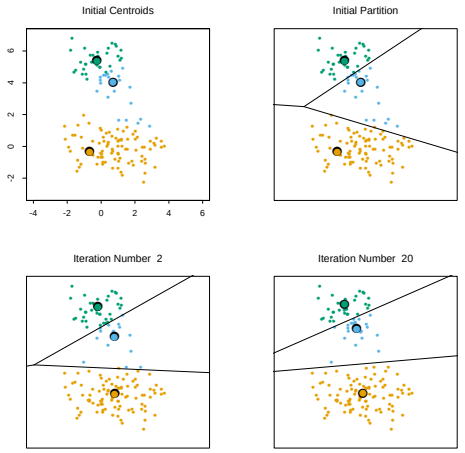
$$\hat{C}(i) = \text{argmin}_k \ ||\mathbf{x}_i - \mathbf{m}_k||_2^2$$

$\Longrightarrow$ Assign each observation to closest cluster mean.

K-means Algorithm



K-means Algorithm

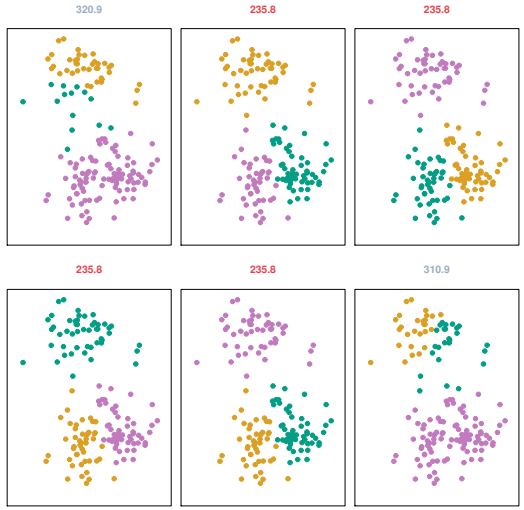


K-means Properties

- Steps 1 and 2 decrease $W(C)$.
- Local solution - not necessarily global solution.
- Depends on starting values (initialization).
- K needs to be set before.
- Best for compact, spherical clusters.
- Does not work well when cluster sizes are different.

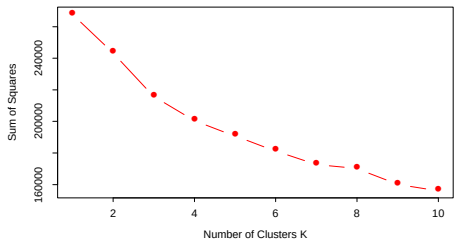
K-means in R: `kmeans`

K-means - Initializations

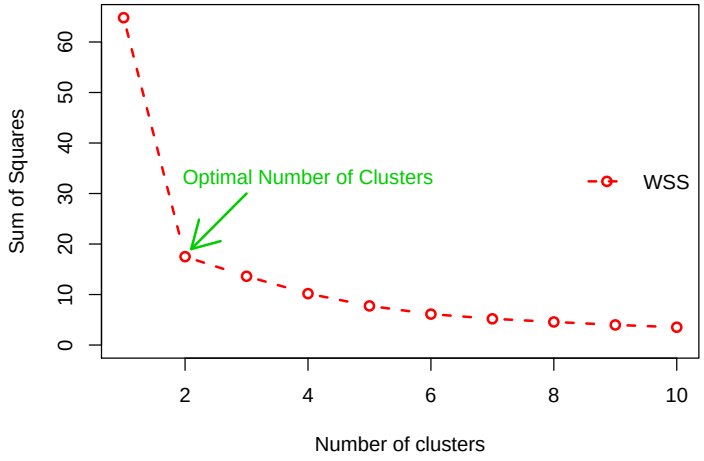


How to Choose K?

- Can we choose K that minimizes $W(C)$?
- Can we choose K by a Validation set? Cross-Validation?



How to Choose K?



How to Choose K?

- Gap Statistic.
- Silhouette Statistic.
- Cluster Prediction Strength.
- Cluster Stability.

How to Choose K?

Gap Statistic:

- Idea: Choose K that yields most grouped data compared to random data.
- Random uniform points over data domain.
- Cluster random points with K-means.
- Choose K that gives biggest "Gap" (difference) between random $W(C)^*$ and observed $W(C)$.
- Issues with this?
- R: `clusGap` in `cluster` package.

How to Choose K?

Silhouette Statistic:

- a_i - mean within-cluster dissimilarity with observation i.
- b_i - mean between-cluster dissimilarity with observation i.
- Silhouette Statistic:

$$S_i = \frac{b_i - a_i}{\max(b_i, a_i)}.$$

- S_i close to 1 = good clustering.
- S_i close to -1 = bad clustering.
- Choose K that maximizes average S_i.
- R: `silhouette` in `cluster` package.

How to Choose K?

Prediction Strength:

- Choose K where clusters have most overlap between many training and test set splits.

Cluster Stability:

- Perturb data (bootstrap; sub-sampling; etc.).
- Choose K where cluster assignments are most stable over perturbations.

Metrics to measure overlap between cluster assignments:

- Rand Index.
- R: `rand_indep` in `clusteval` package.
- Jaccard Index.
- R: `jaccard_indep` in `clusteval` package.

Applications - K-means

- Vector Quantization - Signal Processing & Image compression.

- Community Detection in Networks.
- Many others!!

Summary - K-means

Strengths:

- Fast.
- Simple.
- Others?

Weaknesses:

- Local solution - highly depends on initialization.
- High-dimensional settings? ($p >> n$ - more features than observations)
- Others?

K-means - Related Algorithms

Soft Clustering: Mixture Models

- Mixture of k distributions.
- Assign each observation a probability of arising from distribution k.
- Most Common: Gaussian Mixture model.
- Algorithm: EM (Expectation-Maximization).
  - E-step: Cluster probabilities for each observation.
  - M-Step: Given soft-cluster assignments, maximize likelihood for each distribution.

`mclust` package in `R`.

Soft-Clustering: NMF

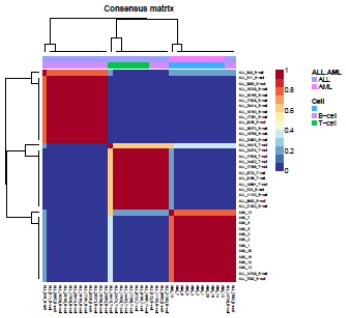
$$\mathbf{X}_{n\times p} \approx \mathbf{W}_{n\times K}\,\mathbf{H}_{K\times p}$$

- Clusters: Each column of $\mathbf{W}$.
- Soft-Cluster Assignments: $\mathbf{W}_k^T = (.4, 1, 0, 0, 2.1, 0)$.
- Observations can be assigned non-zero weights to more than one cluster.
- Hard-Cluster Assignment: Cluster of i defined as argmax of $\mathbf{W}_i$.
- Features contributing to cluster k: Rows of $\mathbf{H}_k$.

`NMF` package in `R`.

Soft Clustering - NMF

Consensus NMF Clustering:



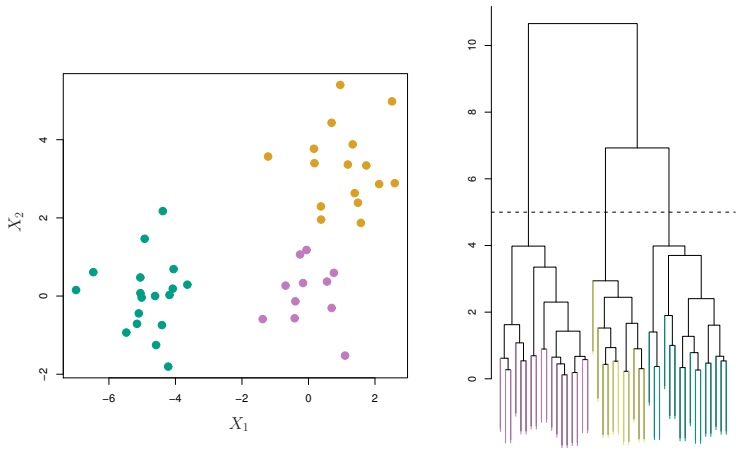
- Consensus = % time observation i and j in same cluster.

Hierarchical Clustering

Hierarchical Clustering

- Nested Clusters: Produce hierarchical representations in which the clusters at each level of the hierarchy are created by merging clusters at the next lower level.
- At the lowest level, each cluster contains a single observation.
- At the highest level there is only one cluster containing all observations.
- Two paradigms: agglomerative (bottom-up; most popular) and divisive (top-down; less popular).
- Use dendrogram to display the clustering result.

Interpreting a Dendogram



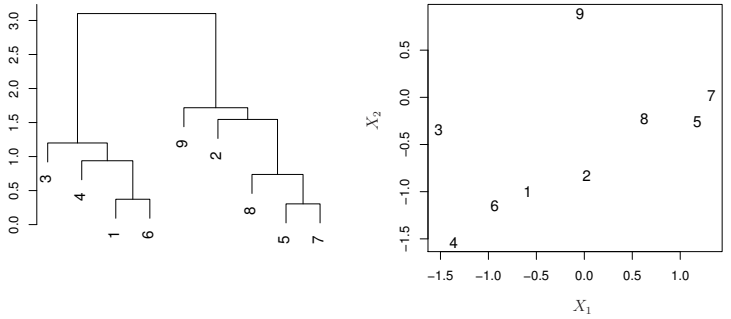
Interpreting a Dendogram

- Bottom of the tree - leaf for each observation.
- As we move up the tree, some leaves begin to fuse into branches: these are observations that are similar to each other.
- The lower in the tree fusions occur, the more similar the groups of observations are to each other.
- Observations that fuse near the top of the tree, can be quite different.
- Height of fusions indicate how similar objects are.
- Horizontal axis does not indicate how similar objects are - just the vertical.

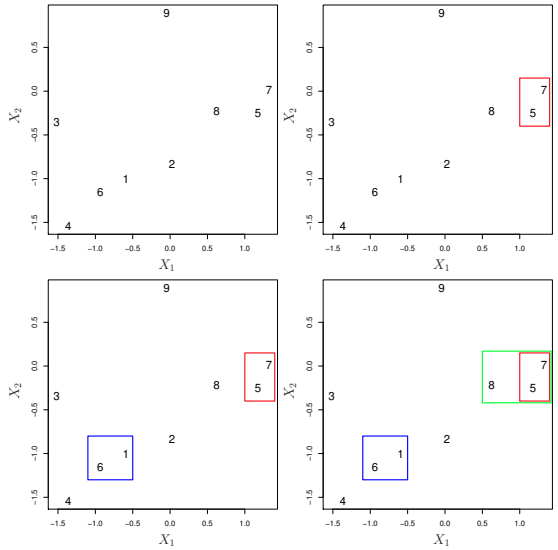
Agglomerative Clustering

- Begin with every observation representing a singleton cluster.
- At each step, merge two "closest" clusters into one cluster and reduce the number of clusters by one.
- Need a measure of dissimilarity between two clusters - called linkages.
- Dissimilarity between G and H: $d(G, H)$, function of the set of pairwise dissimilarities $d_{ii'}$, point i is in G and point i' is in H.

Agglomerative Clustering



Agglomerative Clustering



Linkages

Linkages - Measure of dissimilarity between two sets of objects that determine how two set of objects are merged.

Major Types:

- Single linkage.
- Complete linkage.
- Average Linkage.
- Ward's Linkage.

Single Linkage

$$d_{SL}(G, H) = \min_{i \in G, i' \in H} d_{ii'}$$

.

- Minimum dissimilarity between points in two sets used to determine which two sets should be merged.
- Can handle diverse shapes.
- Very sensitive to outliers or noise.
- Often results in unbalanced clusters.
- Extended, trailing clusters in which observations fused one at a time - chaining.

Complete Linkage

$$d_{CL}(G, H) = \max_{i \in G, i' \in H} d_{ii'}$$

- Maximum dissimilarity between points in two sets used to determine which two sets should be merged.
- Often gives comparable cluster sizes.
- Less sensitive to outliers.
- Works better with spherical distributions.

Average Linkage

$$d_{GA}(G, H) = \frac{1}{n_G n_H} \sum_{i \in G} \sum_{i' \in A} d_{ii'}$$

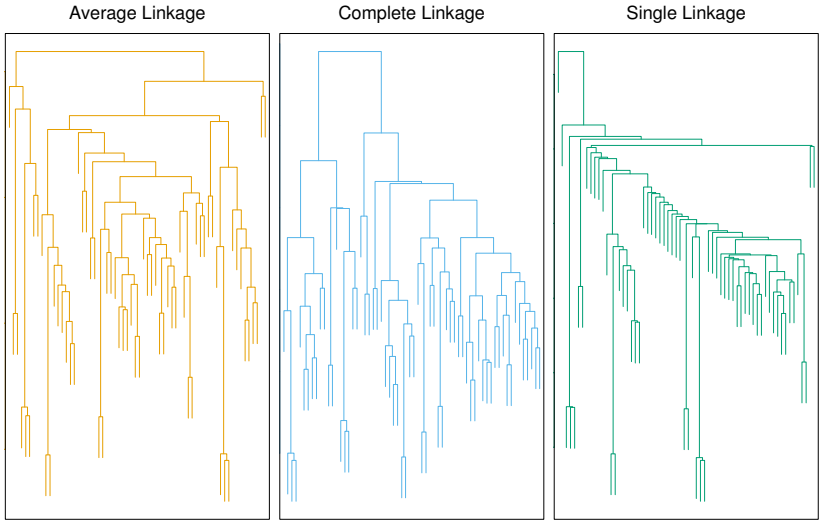
- Average dissimilarity between points in two sets used to determine which two sets should be merged.
- A compromise between single and complete linkage.
- Less sensitive to outliers.
- Works better with spherical distributions.

Similar linkage: Ward's linkage.

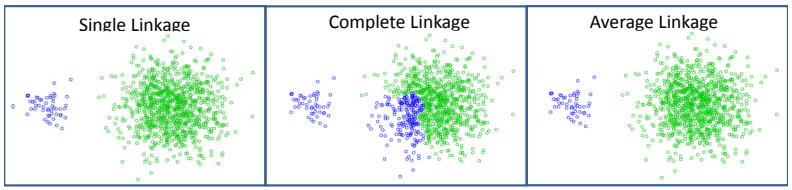
$$d_{i,j} = || \mathbf{x}_i - \mathbf{x}_j ||_2^2$$

- Join objects that minimize Euclidean distance / average Euclidean distance.

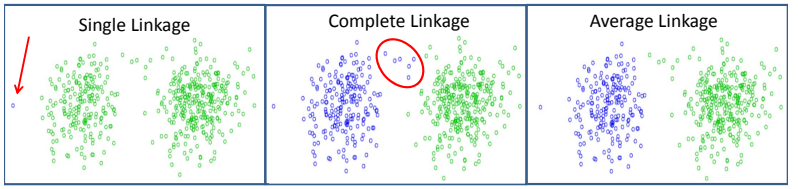
Linkage Examples



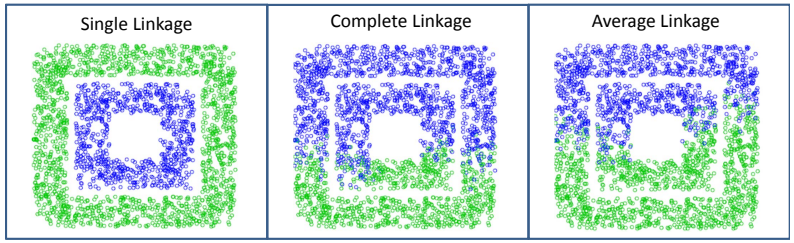
Linkage Examples



Linkage Examples



Linkage Examples



Linkages

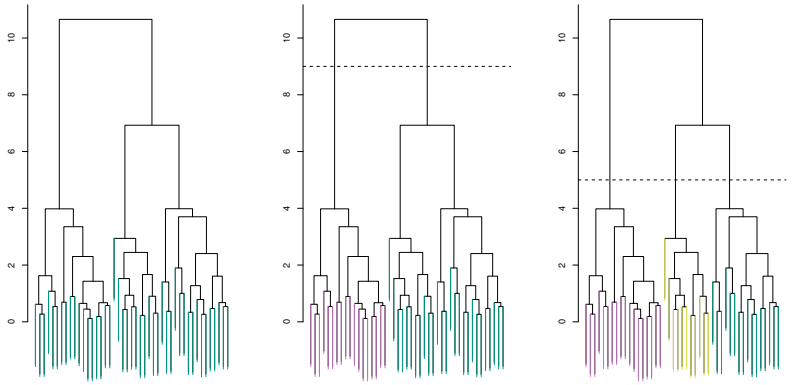
Discussion:

- When are different linkages appropriate?
- Average Linkage has a statistical consistency property violated by single and complete linkage.
- Most Robust?

R: `hclust` function with `method=''single''`, `method=''complete''`, `method=''average''`, `method=''ward.D''`

Number of Clusters

Tree-Cuts:



R: `cutree` function.

Hierarchical Clustering - Summary

Strengths:

- Simple / intuitive.
- Visualization.
- Family of possible clusterings (nested).

Extremely popular!!

Weaknesses:

- Local Solution.
- Unstable Solution.
- Depends heavily on type of linkage.
- No optimization criterion - purely algorithmic.

Biclustering

Biclustering

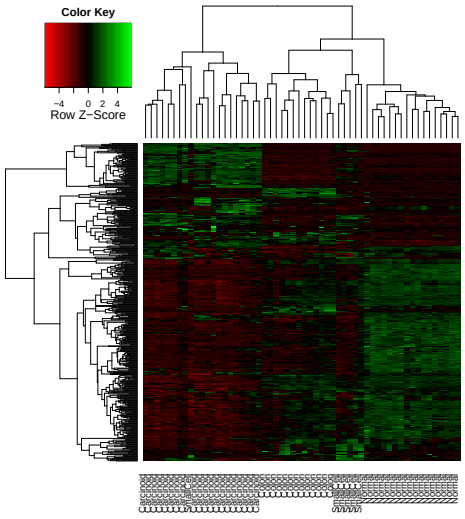
- Idea: Find groups of BOTH observations & features.
- Like clustering both rows and columns of data matrix.

Two main types:

1. Overlapping Biclusters.
   - Plaid models & Sparse SVD models.
2. Non-overlapping Biclusters (Checkerboard mean).
   - Cluster heatmap. (`heatmap` in `R`)

Cluster Heatmap

Hierarchical Clustering Separately on Rows & Columns:



Biclustering - Applications

- Biomedicine - "omics" data.
  - Cancer genomics: Finding subtypes. Find groups of patients (subtypes) and groups of genes (genomic signatures) that separate subtypes.
  - Famous Example: Breast Cancer.
- Text mining.
  - Word-Document associations.
- Collaborative Filtering.
  - Find users who highly rate particular products.
  - Famous Examples: Netflix, Amazon.

Convex Clustering & Biclustering

Convex Clustering

Motivation:

- Clustering algorithms yield local solutions, dependent on initializations, unstable results.
- Can we formulate a *convex* method for clustering that will yield a unique & global solution?

Idea: Fuse mean column vectors together to form clusters.

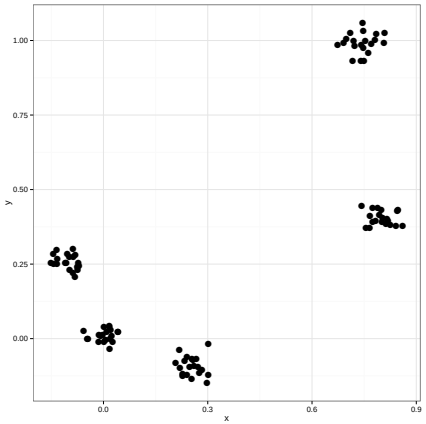
Convex Clustering

Optimization Problem:

$$\underset{\mathbf{U}}{\text{minimize}} \quad \frac{1}{2}\sum_{i=1}^{n} || \mathbf{x}_i - \mathbf{u}_i ||_2^2 + \lambda \sum_{i<i'} w_{ii'} || \mathbf{u}_i - \mathbf{u}_{i'} ||_1$$

- λ controls BOTH cluster assignments & number of clusters. (Can be selected by cross-validation)
- $\lambda = 0$ - each observation is its own cluster.
- λ larger - column means begin to coalesce together into clusters.
- λ very large - all observations fused into one cluster.

Convex Clustering



$\lambda = 0$

Convex Biclustering

Motivation:

- Cluster heatmap highly unstable.
- Can we formulate a *convex* method for biclustering that will yield a unique & global solution?

Idea:

- Simultaneously fuse row & column means to form bicluster means.
- Finds a checkerboard mean matrix.

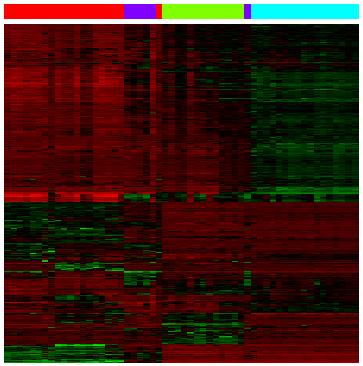
Convex Biclustering

Optimization Problem:

$$\underset{\mathbf{U}}{\text{minimize}} \quad \frac{1}{2}||\mathbf{X}-\mathbf{U}||_F^2 + \lambda \left[\sum_{i<i'} w_{ii'}||\mathbf{U}_{i,}-\mathbf{U}_{i',}||_1 + \sum_{j<j'} \tilde{w}_{jj'}||\mathbf{U}_{,j}-\mathbf{U}_{,j'}||_1\right]$$

- λ controls BOTH bicluster assignments & number of biclusters.
- $\lambda = 0$ - each matrix element its own bicluster.
- λ larger - row and column means begin to coalesce together into biclusters.
- λ very large - whole matrix is one bicluster (matrix mean).

Convex Biclustering



$\lambda = 0$

Convex Clustering & Biclustering

Strengths:

- Unique, global solution.
- Stable solution.
- Fast algorithm.

Weaknesses:

- Performance can depend on weights.

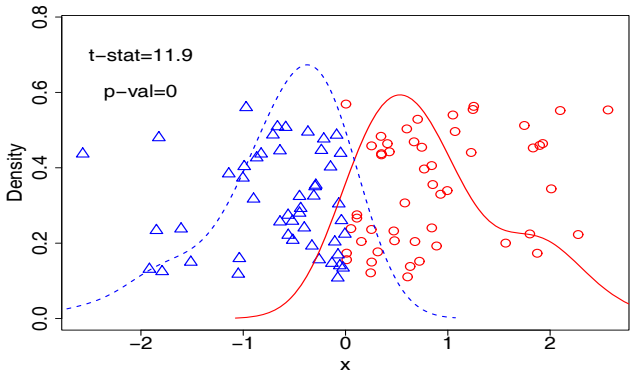
`cvxclustr` and `cvxbiclustr` in `R`.

Significance of Clustering (SigClust)

Question of Cluster Significance

- When clusters seem to appear, e.g. found by clustering method, how do we know they are really there?
- How to assess significance?
- High dimensional data in modern data analysis make it more challenging to assess significance
- Need a more formal definition of clusters

Simple Gaussian Example



Lessons of Simple Gaussian Example

- Clearly only 1 Cluster in this Example.
- Random relabelling T-statitic is not significant, but extreme T-statistic is strongly significant.
- This comes from clustering operation
- Conclude sub-populations are different
- Message from the toy example: sub-populations are not the same as clusters really there.

Definition of Cluster

- No universal definition
- SigClust definition of a cluster: data coming from a single Gaussian distribution (may not be standard spherical normal).
- Gaussian mixtures have similar definitions (Titterington et al. (1985), McLachlan and Basford (1988), etc.)
- Main difference: SigClust does not attempt to estimate full parameters of Gaussian components of mixture models.

Statistical Significance of Clustering (SigClust)

- Formulate a test procedure
  H_0 : data from a single Gaussian distribution
  H_1 : data came from d-dimensional non Gaussian distribution.
- Test statistic: 2-means cluster index
- Estimate null distribution from Monte Carlo simulation
- Significance judged by p-value
- Key points: Rotation Invariance of the test statistic (the cluster index)& Factor Model (simplify parameter estimation for HDLSS data)

2-means Cluster Index

- $CI_2 = \frac{\sum_{k=1}^{2} \sum_{j \in C_k} ||\mathbf{x}_j - \bar{\mathbf{x}}^{(k)}||^2}{\sum_{j=1}^{n} ||\mathbf{x}_j - \bar{x}||^2}$
- Within Class Sum of Square Distances to Class Means divide (normalize) by Overall Sum of Suqared Distance to Mean
- Puts on scale of proportions
- Small CI_2 gives tight clustering (within SS contains little variation)
- Large CI_2 gives poor clustering (within SS contains most of variation)

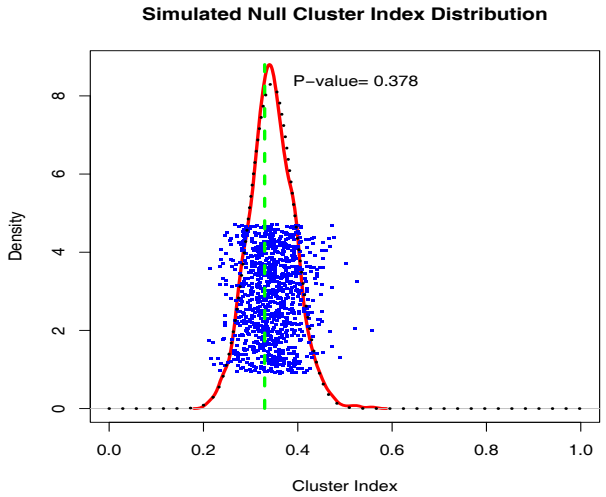
SigClust Gaussian null distribution

- Key step in SigClust is to estimate null distribution $N(\mu, \Sigma)$
- CI_2 is location-invariant $\rightarrow \mu = \mathbf{0}$
- Estimation of $\Sigma = M \Lambda M^T$ is crucial in HDLSS data
  CI_2 is orthogonal rotation-invariant: $\rightarrow$ only need to estimate diagonal matrix Λ
  (If $\mathbf{z}_i = M\mathbf{x}_i$ and $M^{-1} = M^T$, then $||\mathbf{z}_i - \mathbf{z}_I||^2 = ||\mathbf{x}_i - \mathbf{x}_I||^2$.)
- Model covariance as: $\Lambda = \Lambda_B + \sigma_N^2 \times I_d$ (Biology + Noise)
  - Λ_B is "fairly low dimensional"
  - σ_N^2 is estimated from background noise $\hat{\sigma}_N = \frac{\text{MAD}_{d \times n \text{ data set}}}{\text{MAD}_{N(0,1)}}$
- Estimated Null distribution using univariate Gaussian distributions.

Procedures for the SigClust

- Step 1: Calculate 2-means cluster index for original dataset.
- Step 2: Get $\hat{\sigma}_N^2$ using all entries from original data matrix.
- Step 3: Simulate data from null distribution: $(x_1, \ldots, x_d)$ are independent with $x_j \sim N(0, \tilde{\lambda}_j)$ (Liu et al., 08, Huang et al., 14).
- Step 4: Perform 2-means clustering on simulated data from Step 3 and calculate corresponding 2-means cluster index.
- Step 5: Repeat Steps 3 and 4 N_{Sim} times to obtain an empirical distribution of cluster index based on H_0.
- Step 6: Using cluster indices of the simulated data, calculate a p-value for the cluster index of the original dataset.

SigClust Results on the Toy Example



Two real cancer data sets from The Cancer Genome Atlas Network (TCGA)

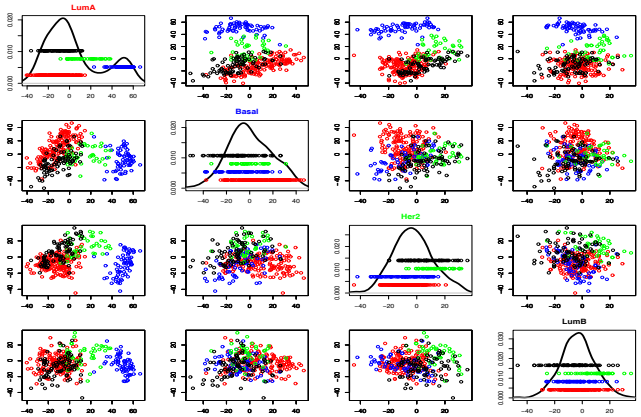
Glioblastoma Multiforme (GBM):

- $n = 383$, $d = 2727$
- 117 Mesenchymal, 69 Neural, 96 Proneural, and 101 Classical

Breast Cancer data (BRCA):

- $n = 348$, $d = 4000$
- 154 LumA, 81 LumB, 42 Her2, and 66 Basal

PCA projection scatter plot view of the BRCA data



GBM	MES.CL	MES.PN	MES.NL	CL.PN	CL.NL	PN.NL
Sample	0.9	0	0	0	0	0
Hard	$< 10^{-4}$	0	0	0	0	0
combined	10^{-3}	0	0	0	0	0
BRCA	LA.B	LA.H	LA.LB	B.H	B.LB	H.LB
Sample	$< 10^{-4}$	0.8	1	0.01	10^{-3}	0.9
Hard	0	$< 10^{-4}$	0.9	0	0	0.06
combined	0	0.03	1	$< 10^{-4}$	0	0.9

`sigclust and hsigclust in R.`

References

Textbooks:

- Elements of Statistical Learning by Hastie, Tibshirani & Friedman.
  `http://statweb.stanford.edu/~tibs/ElemStatLearn/`
- Introduction to Statistical Learning by James, Witten, Tibshirani & Hastie.
  `http://www-bcf.usc.edu/~gareth/ISL/`

Some of the figures in this presentation are taken from these two textbooks with permission from the authors.