Dendron: a simple library to assess stability of clustering patterns on HCA

Eugenio J. Llanos B. Wilmer Leal J.

ellanos@sciocorp.org

Universität Leipzig
Max Planck Institut für Mathematik in den Naturwissenschaften
Corporación SCIO

October 11, 2019

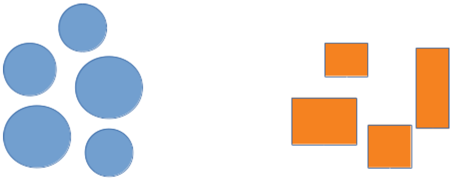
Table of Contents

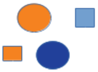
Table of Contents

Clustering

What is clustering?

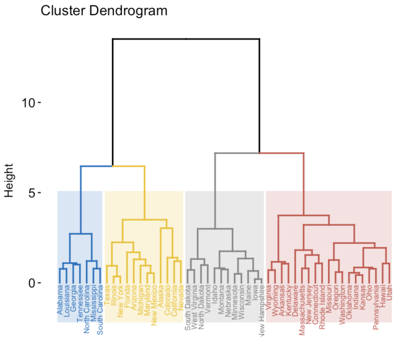
Clustering is a frequently used technique of multivariate analysis that divides a dataset into groups that contain similar objects, while dissimilar objects fall into different groups. Generally, the aim is to relate similarity on attributes to similarity on behavior.





HCA
Definition

In hierarchical clustering the input is a set X and the output is a nested collection of partitions $D_X(\delta)$, $\delta \geq 0$ (resolution parameter). For a given D_X, $x \sim x' \iff x, x' \in D_X$. If $x \sim x'$ at a given value δ_0, then $x \sim x'$ for all $\delta \geq \delta_0$.



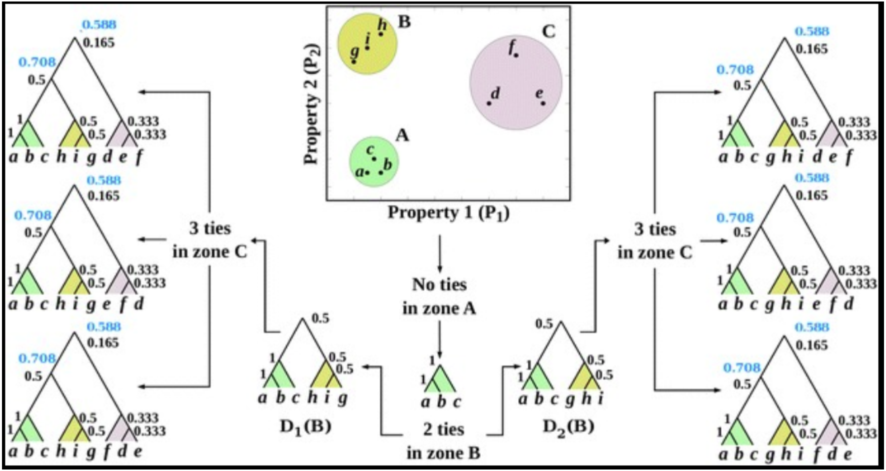
HCA in Python
scipy.cluster.hierarchy

Package *scipy.cluster* offers the module *hierarchy*, which performs hierarchical clustering form a data set returning a *linkage matrix*. Such matrix can be transformed into a tree like structure (to_tree). The algorithm uses a data structure called a *heap* to store distances.
Dendrogram might be plotted using *hierarchy.dendrogram()*.

Table of Contents

What are ties?

We don't have just one dendrogram



Ties in proximity

Our implementation

```
def _hcluster_ (vec,dmatrix):
    dist = find_min_dist(vec,dmatrix)
    res=[]
    if len(vec) == 2: return dist
    else:
        if len(dist) == 1:
            return hcluster(make_couple(vec,dist[0]),dmatrix)
        else:
            for i in range(len(dist)):
                nt = hcluster(make_couple(vec,dist[i]),dmatrix)
                res += nt
            return res
```

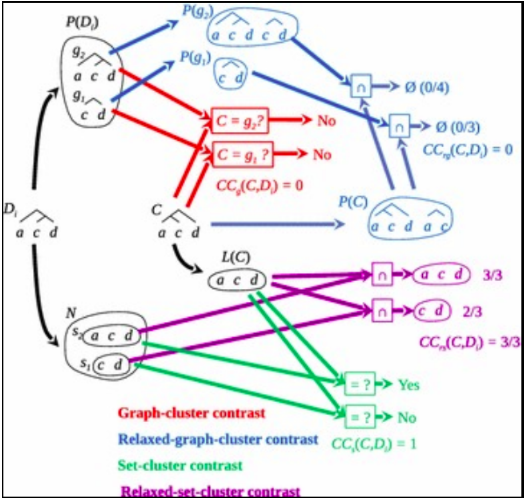
Equidistance

Felsenstein number

The extreme case when $d(x, y) = 0$ when $x = y$ and $d(x, y) = 1$ whenever $x \neq y$ the number of dendrogram is given by the number of Felsenstein:

$$F(n) = \frac{(2n-3)!}{2^{n-2}(n-2)!} \tag{1}$$

Frequency of clusters
Contrast functions



Frequency of clusters
Real data

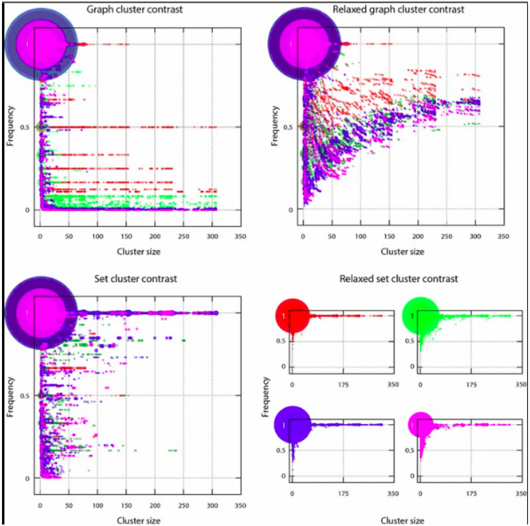
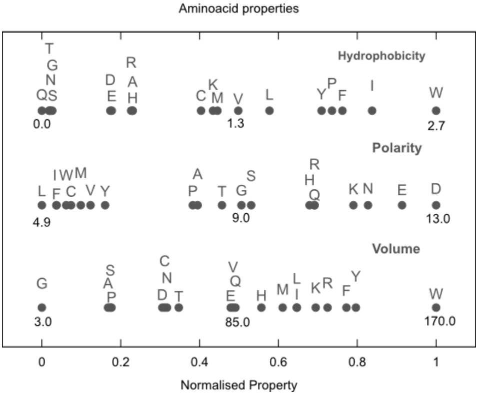


Table of Contents

Random Noise

We don't have just one dendrogram



Adding random noise
Testing groups

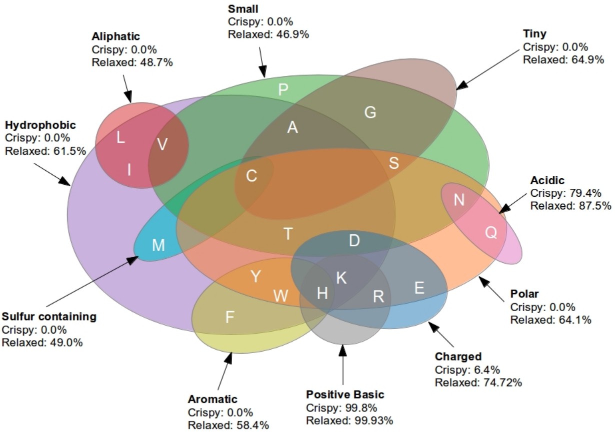


Table of Contents

Our implementation
Main ideas

- Object Oriented approach.
- Recursive functions (clarity).
- Sets are represented as bit vectors.
- Graphs are represented as tuples.

Dendron

What can be done using dendron

- Calculate all dendrograms taking into account ties in proximity.
- Get one random dendrogram from ties in proximity.
- Add random noise to objects and perform an HCA.
- Calculate frequency of patterns in a given set of replicas using four contrast functions.
- Calculate frequency of all patterns on a set of replicas.
- Convert dendrograms into linkage matrix format to plot them with scipy.