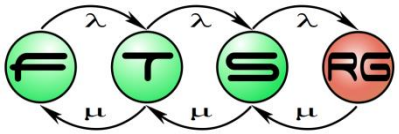
Clustering and visual evaluation

Visual Analysis of Measurement Data

László Gönczy

2019.12.05.

Budapest University of Technology and Economics
Fault Tolerant Systems Research Group

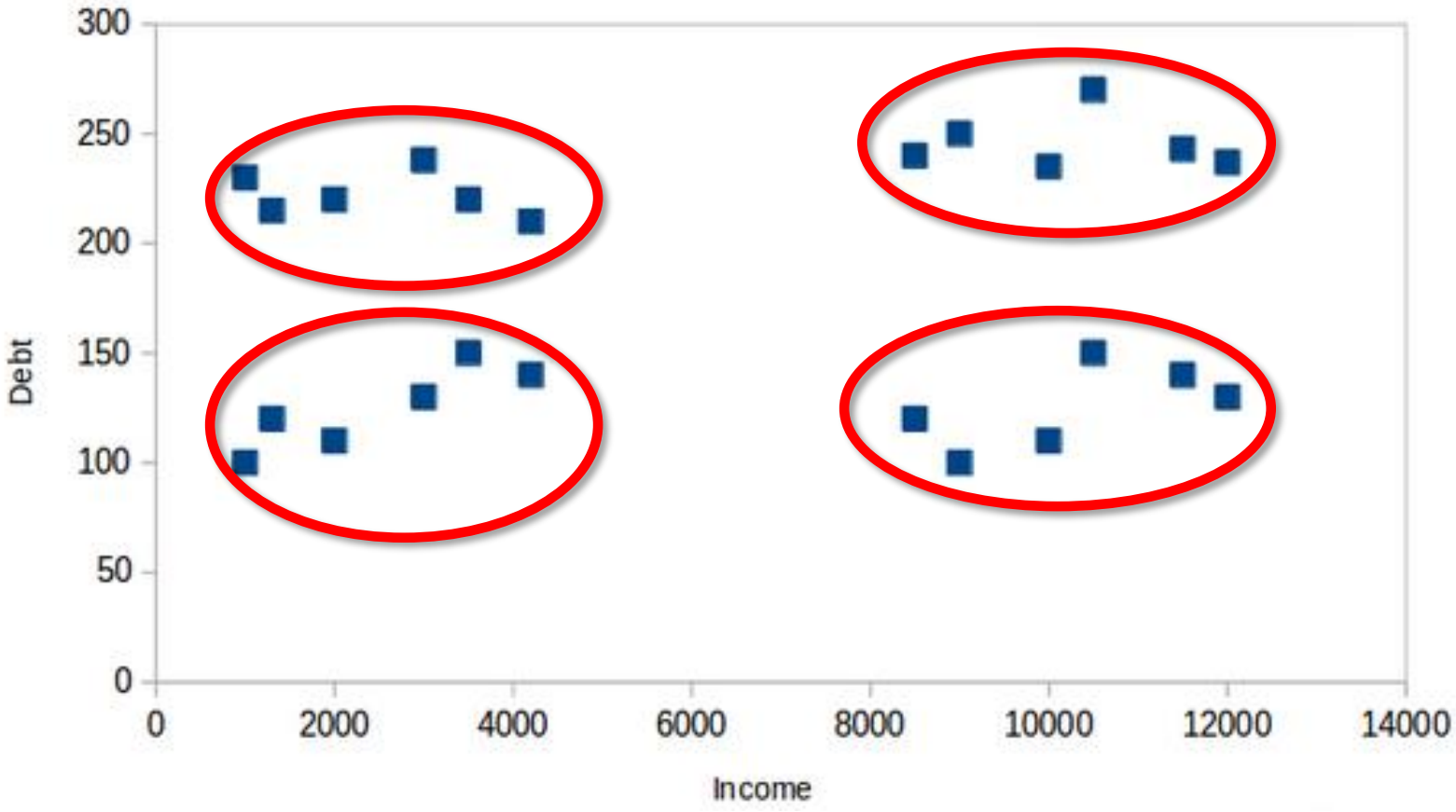


Goal

- Find patterns in data
- Emphasize similarities
  - E.g. operational modes, similar data sources, etc.
- Handle typical cases
- How many groups?
- How to measure similarity?

Example

- Visual clustering

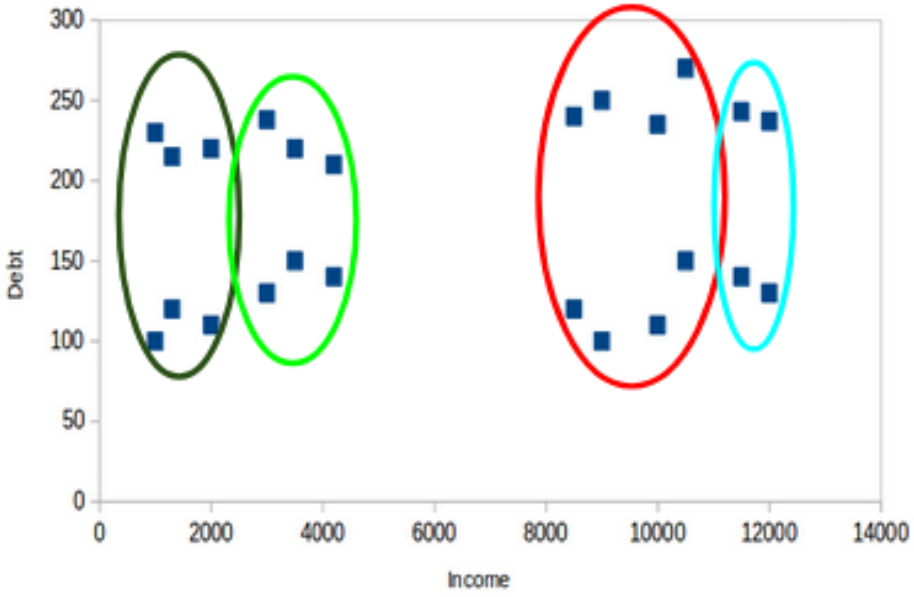


https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/

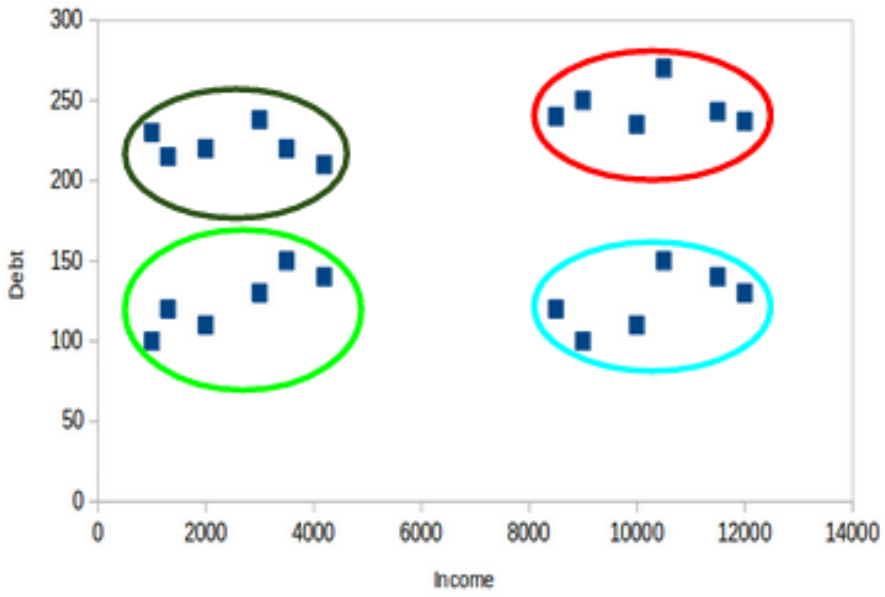
Assumptions

- Similarity: data points in a given cluster are similar to each other (→ can be represented by a typical data point)
- Distance: the distance between clusters should be larger than within clusters

Example



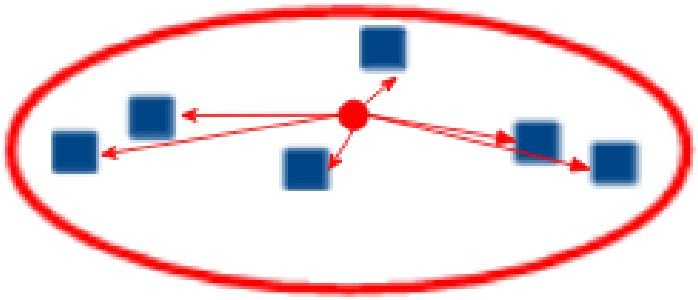
Case - I



Case - II

https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/

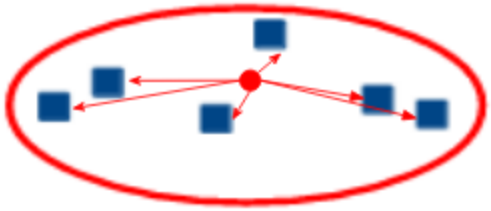
Basic properties: inertia



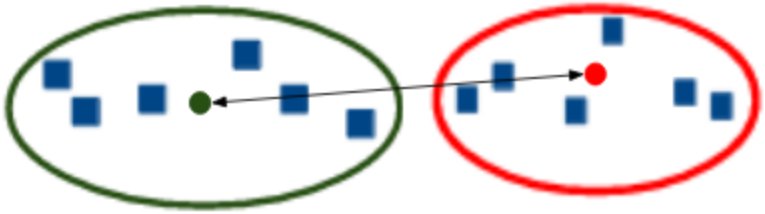
Intra cluster distance

https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/

Basic properties: Dunn index



Intra cluster distance



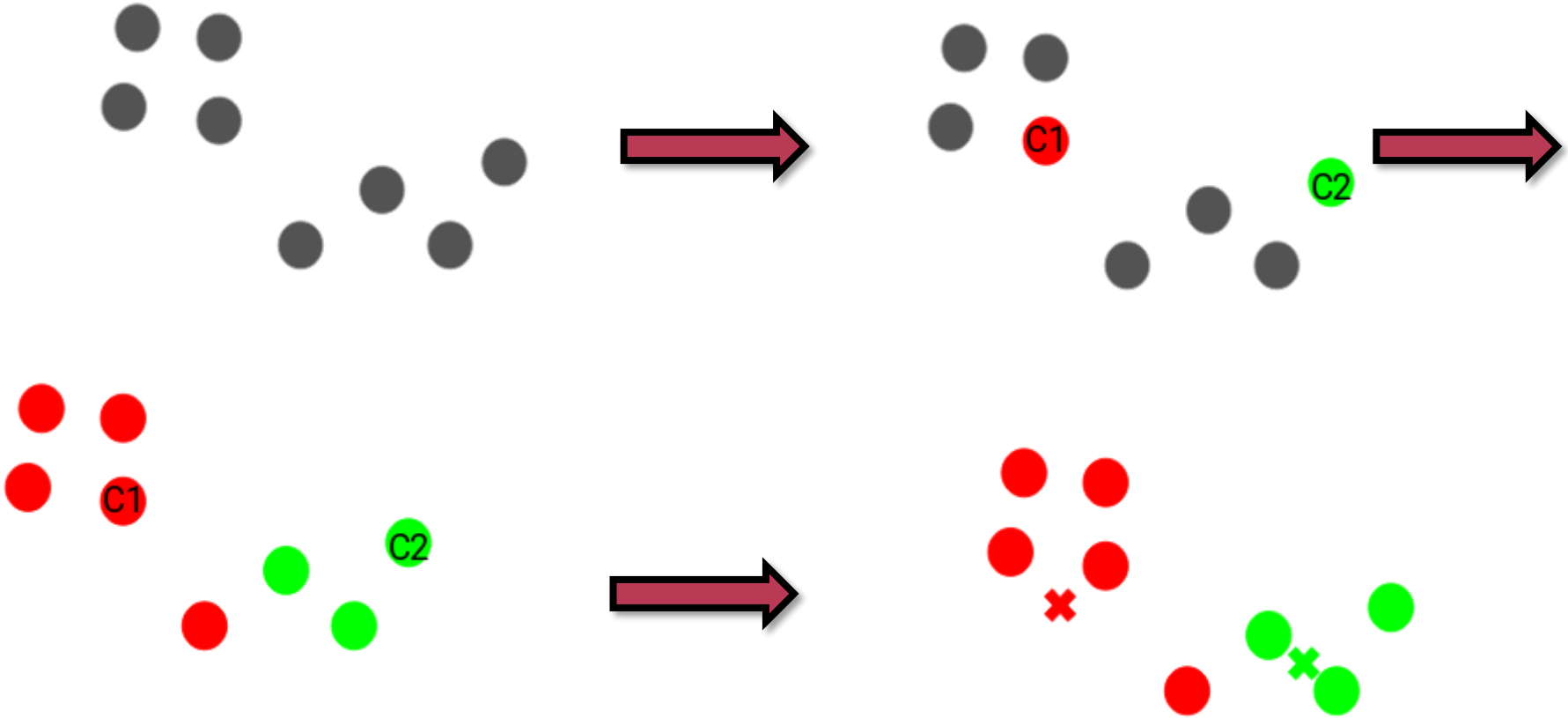
Inter cluster distance

$$\text{Dunn Index} = \frac{\min(\text{Inter cluster distance})}{\max(\text{Intra cluster distance})}$$

https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/

How to determine clusters?

- Simplest solution: k-means
- k = number of clusters (input!)



https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/

When to stop?

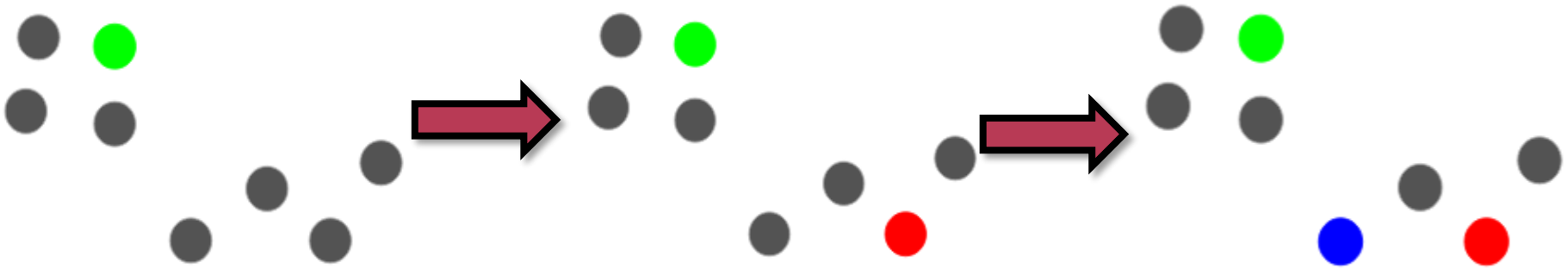
- Centroids will remain the same
- Point will remain in same cluster
- Max number of iterations reached
- How to start in an effective way?
- How to determine the number of clusters?

Kmeans++

- One centroid (instead of k)

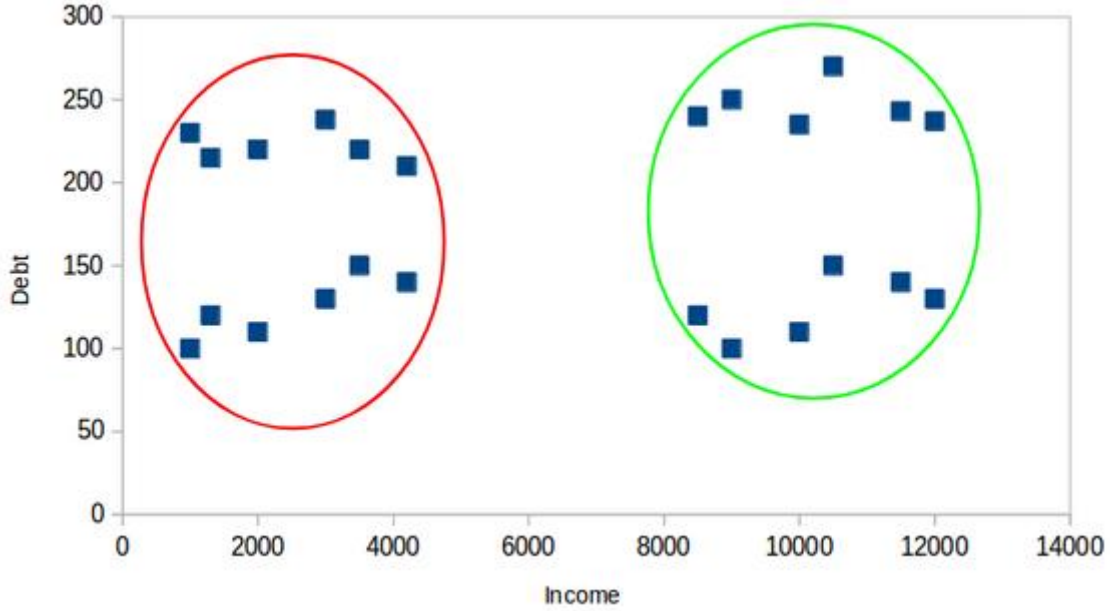
→ take the farthers one

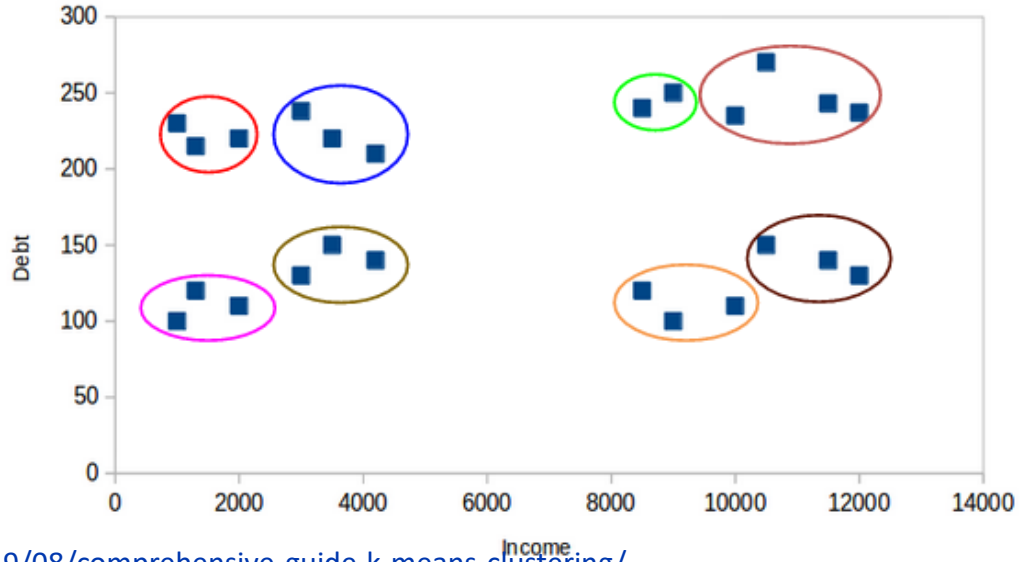
→repeat this until k centroids are selected



https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/

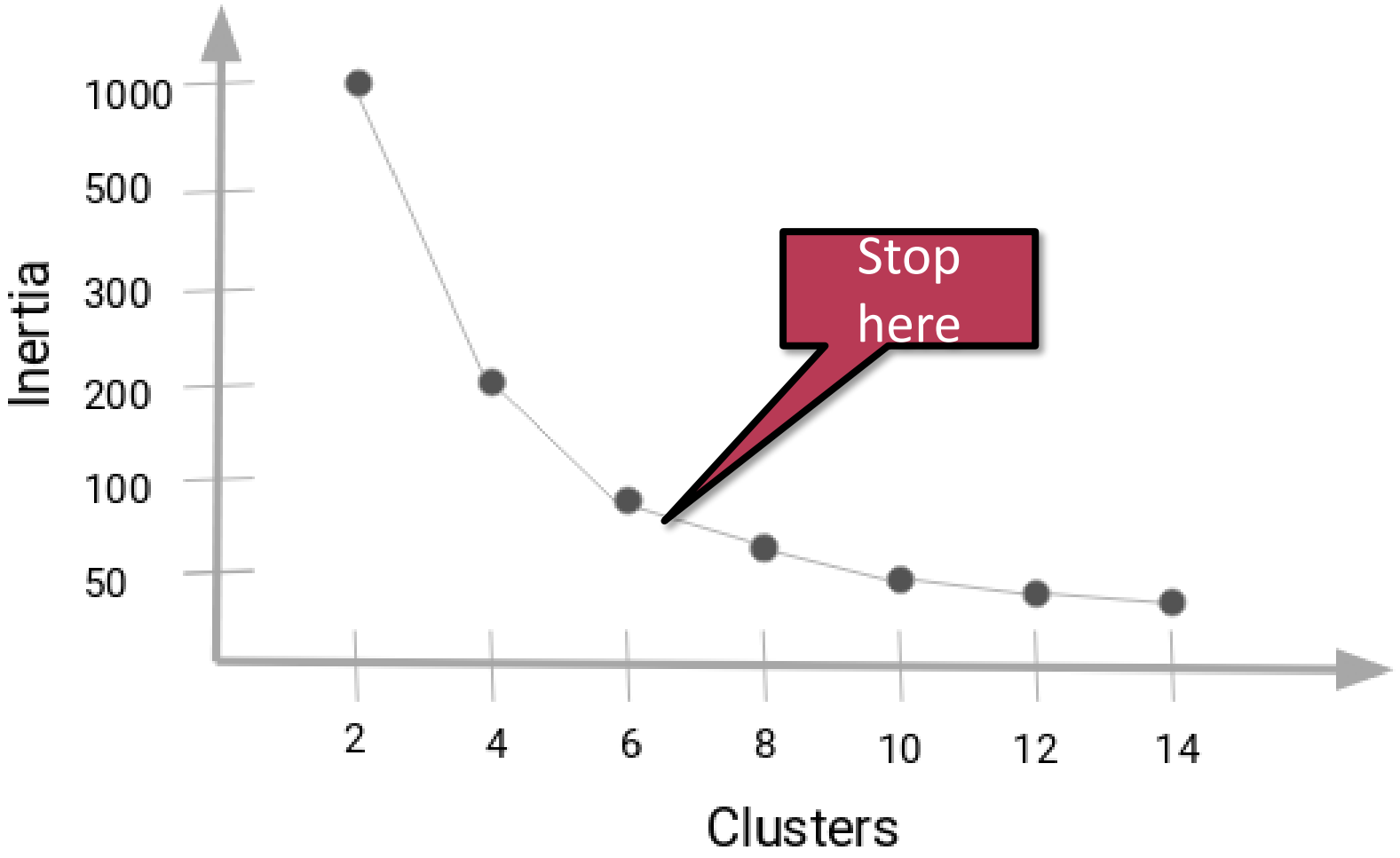
How many clusters?





https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/

Evaluate inertia vs k



VISUAL EVALUATION OF SCATTERPLOTS

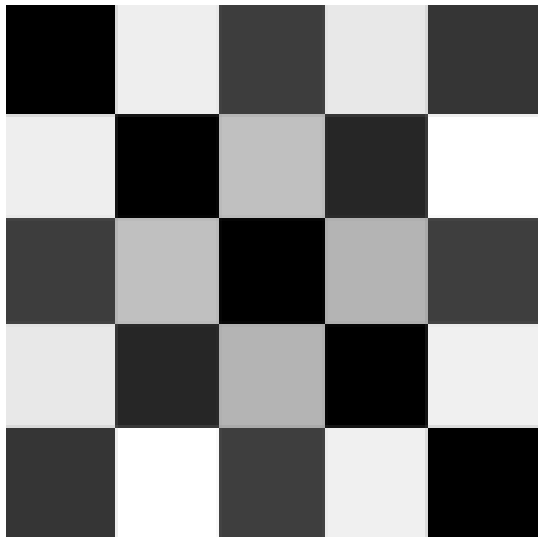
VAT algorithm

- Visual Assessment of Cluster Tendency
- Goal: support clustering by evaluating pairwise similarity → dissimilarity matrix, dissimilarity image
- Organize plot to create similare groups → Ordered Dissimilarity Image (ODI)

Bezdek J.C., Hathaway, R.J., 2002. VAT: a tool for visual assessment of (cluster) tendency.
Proceedings of the IEEE International Joint Conference on Neural Networks, , pp. 2225-2230.
Hathaway R.J., Bezdek J.C., 2003. Visual cluster validity for prototype generator clustering models.
Pattern Recognition Letters, 24, 1563?1569.
Huband J.M., Bezdek J.C., 2008. VCV2 ? Visual Cluster Validity.
In Zurada J.M., Yen G.G., Wang J. (Eds.): Lecture Notes in Computer Science, 5050, pp. 293?308. Springer-Verlag, Berlin Heidelberg.

Dissimilarity matrix/image

$$\mathbf{R} = \begin{pmatrix} 0 & 0.73 & 0.19 & 0.71 & 0.16 \\ 0.73 & 0 & 0.59 & 0.12 & 0.78 \\ 0.19 & 0.59 & 0 & 0.55 & 0.19 \\ 0.71 & 0.12 & 0.55 & 0 & 0.74 \\ 0.16 & 0.78 & 0.19 & 0.74 & 0 \end{pmatrix} = \mathbf{I}$$



Bezdek J.C., Hathaway, R.J., 2002. VAT: a tool for visual assessment of (cluster) tendency. Proceedings of the IEEE International Joint Conference on Neural Networks, , pp. 2225-2230.

Algorithm

Step 1 Set $K = \{1,2,\ldots,n\}$; $I = J = \varnothing$; $P[0] = (0,\ldots,0)$.

Step 2 Select $(i,j) \in \underset{p\in K, q\in K}{\arg\max} \{R_{pq}\}$.

Set $P(1) = i$; $I = \{i\}$; and $J = K - \{i\}$.

Step 3 For $r = 2,\ldots, n$:

Select $(i,j) \in \underset{p\in I, q\in J}{\arg\min} \{R_{pq}\}$.

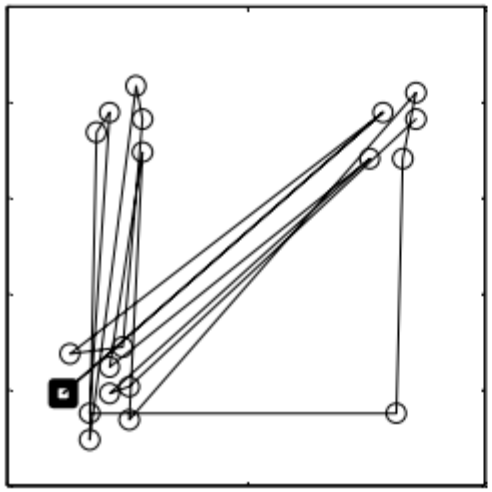
Set $P(r) = j$; Replace $I \leftarrow I \cup \{j\}$ and $J \leftarrow J - \{j\}$.

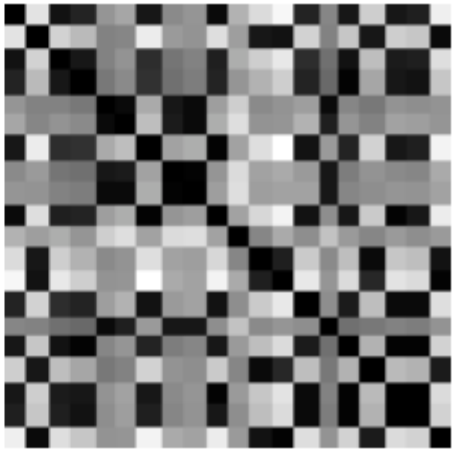
Next r.

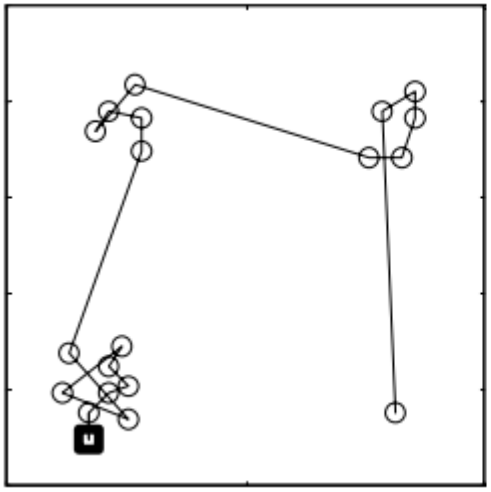
Step 4 Obtain the ordered dissimilarity matrix $\tilde{R}$ using the ordering array P as: $\tilde{R}_{ij} = R_{P(i)P(j)}$, for $1 \le i,j \le n$.

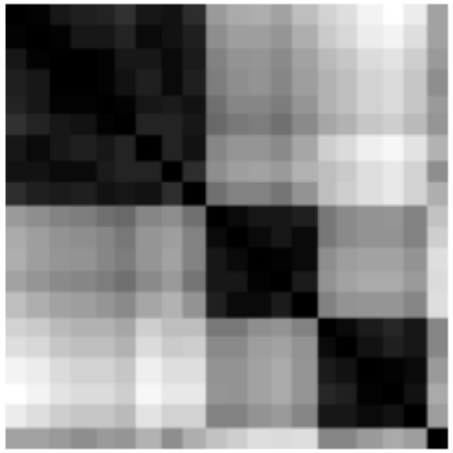
Step 5 Display the reordered matrix $\tilde{R}$ as the ODI $\tilde{I}$ using the conventions given above.

Example

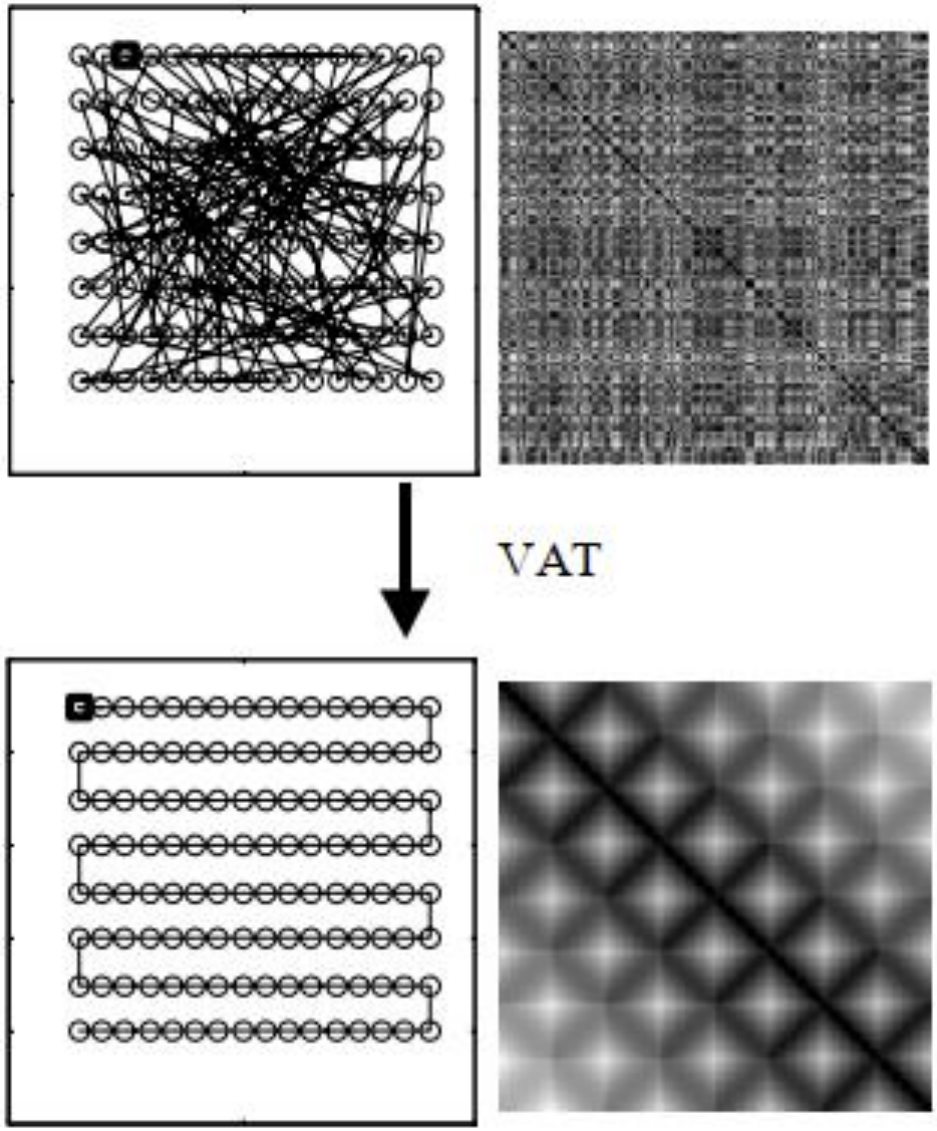




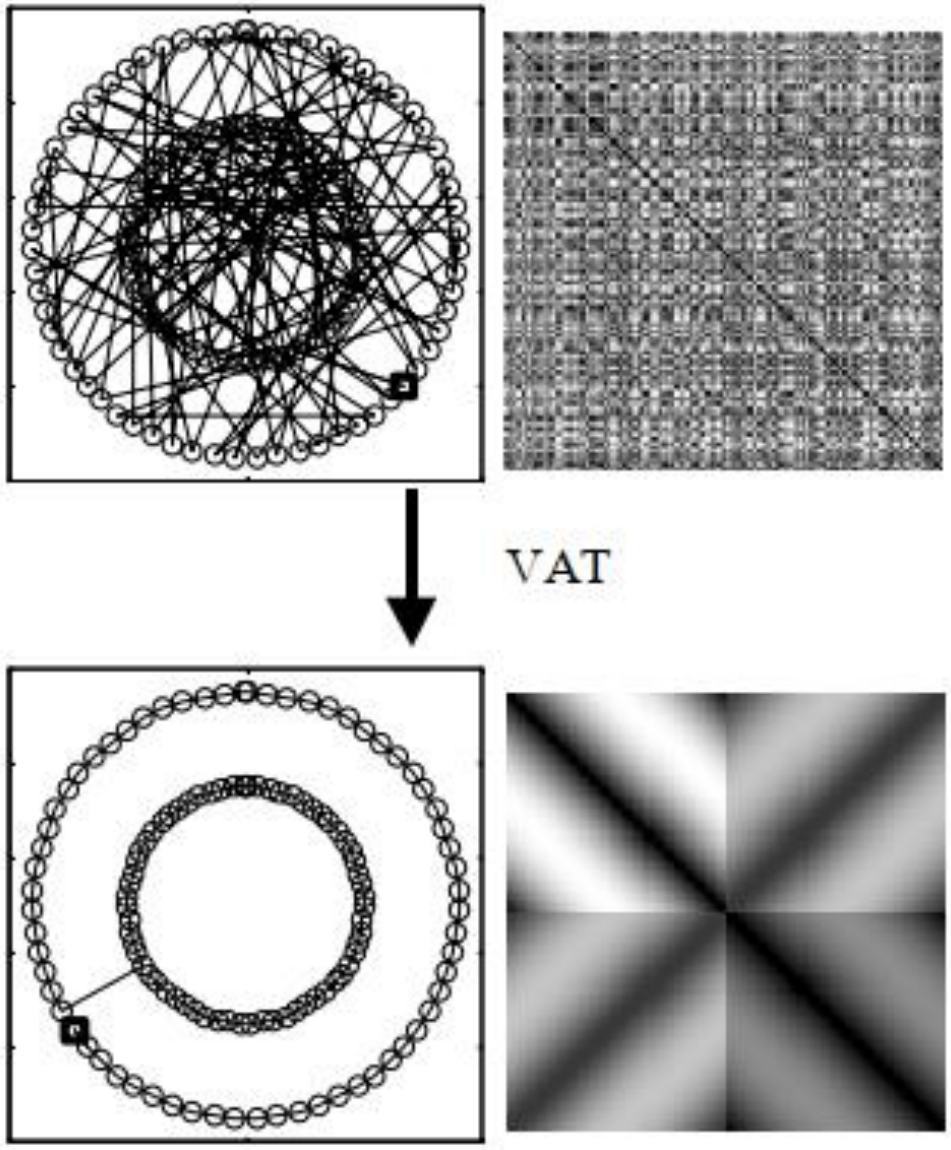




Transforming linear data



Transforming circular data



Example usage

```
library(fclust)

## McDonald's data
data(Mc)
names(Mc)
## data normalization by dividing the nutrition facts by the Serving Size (column 1)
for (j in 2:(ncol(Mc)-1))
Mc[,j]=Mc[,j]/Mc[,1]
## data standardization (after removing the column Serving Size)
Mc=scale(Mc[,1:(ncol(Mc)-1)],center=TRUE,scale=TRUE)[,]
## plot of VAT
VAT(Mc)
```

Source: https://rdrr.io/cran/fclust/man/VAT.html

What if the clusters are complex?

- iVAT: improved VAT
- Instead of simple distance:
- Use the distance of from „close dense points”

→ path-based distance

$$d'_{ij} = \min_{p \in P_{ij}} \left\{ \max_{1 \le h < |p|} d_{p[h]p[h+1]} \right\}$$

- Distort distances: smaller within cluster, larger between clusters

$$f(t_{xy}) = 1 - \exp(-t_{xy}^2 / \sigma^2)$$

Intensity of a given pixel

Mean intensity

aVAT

- Automated detection of number of clusters

→Chamfer matching $d_{cham}(\mathcal{U},\mathcal{V}) = \frac{1}{N}\sum_{\mathbf{u}_i\in\mathcal{U}} \min_{\mathbf{v}_j\in\mathcal{V}} \|\mathbf{u}_i - \mathbf{v}_j\|.$

→Average distance of cluster boundaries
(in image processing: distance between edges)