Recent Advances in Partitioning Clustering Algorithms for Interval-Valued Data

Francisco de A. T. de Carvalho
Universidade Federal de Pernambuco – UFPE, Brazil

Outline

- Introduction
- Some Partitioning Clustering Algorithms for Interval-Valued Data
  - Adequacy criterion
  - Distance functions between vectors of intervals
  - Algorithm
- Cluster and Partition Interpretation
  - Partition Interpretation Indices
  - Cluster Interpretation Indices
- Example
- Final Remarks
- References

Symbolic Data

- Symbolic Data Analysis (Bock and Diday (2000), Billard and Diday (2006), Diday and Noirhome (2008))
  - Aims to develop data analysis methods (clustering, factorial analysis, etc) to manage symbolic data
- Symbolic data generalizes usual categorical or quantitative data
  - A symbolic variable can take several values
- New types of variables
  - Set-valued, ordered list-valued, interval-valued, histogram-valued variables

Interval-Value Data

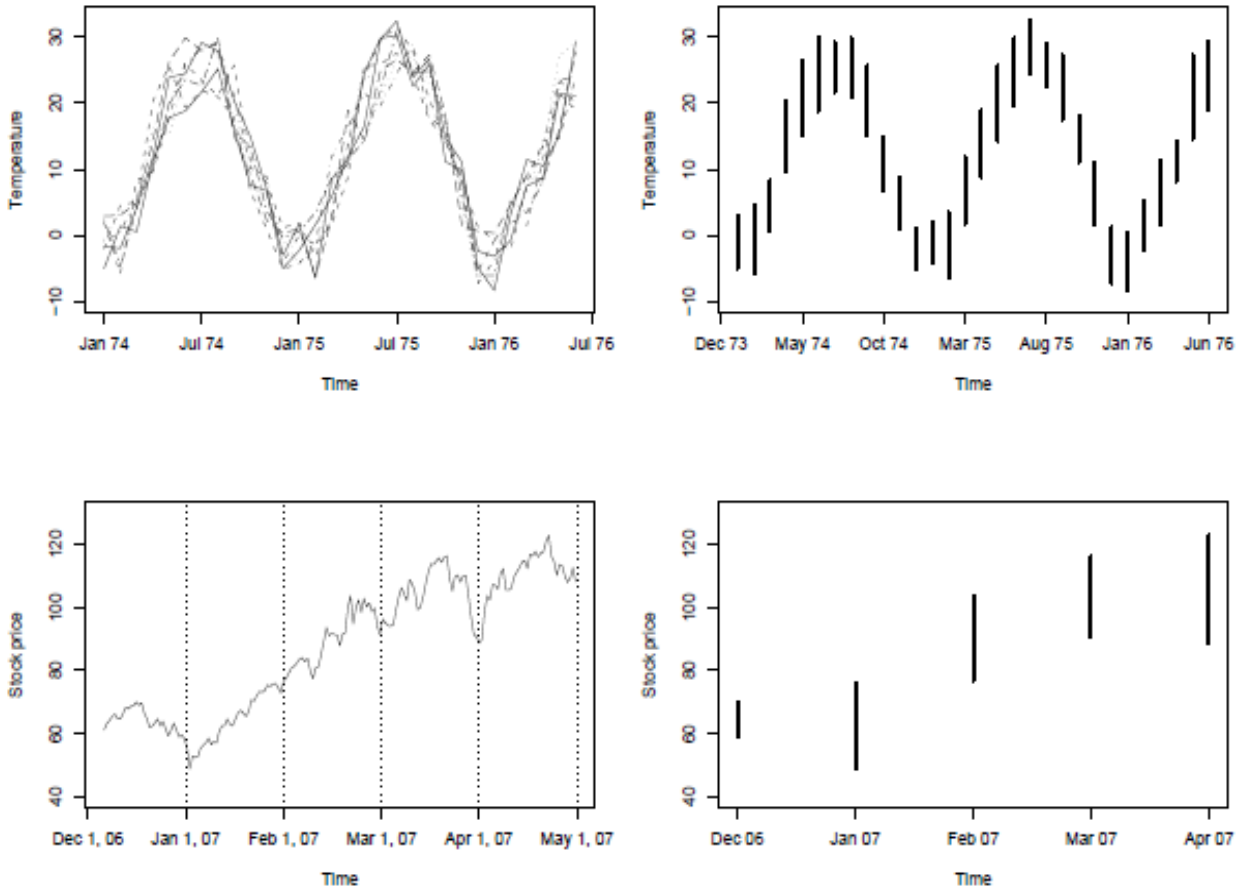
	Pulse Rate	Systolic pressure	Diastolic pressure
1	[60, 72]	[90,130]	[70,90]
2	[70,112]	[110,142]	[80,108]
3	[54,72]	[90,100]	[50,70]
4	[70,100]	[130,160]	[80,110]
5	[63,75]	[60,100]	[140,150]
6	[44,68]	[90,100]	[50,70]

Each object i is described by a vector of intervals

Interval-Valued Data Analysis Tools are required

Introduction

Figure 1: Two interval-valued time series (rigth side) obtained from a set of usual time series (left side).



Introduction

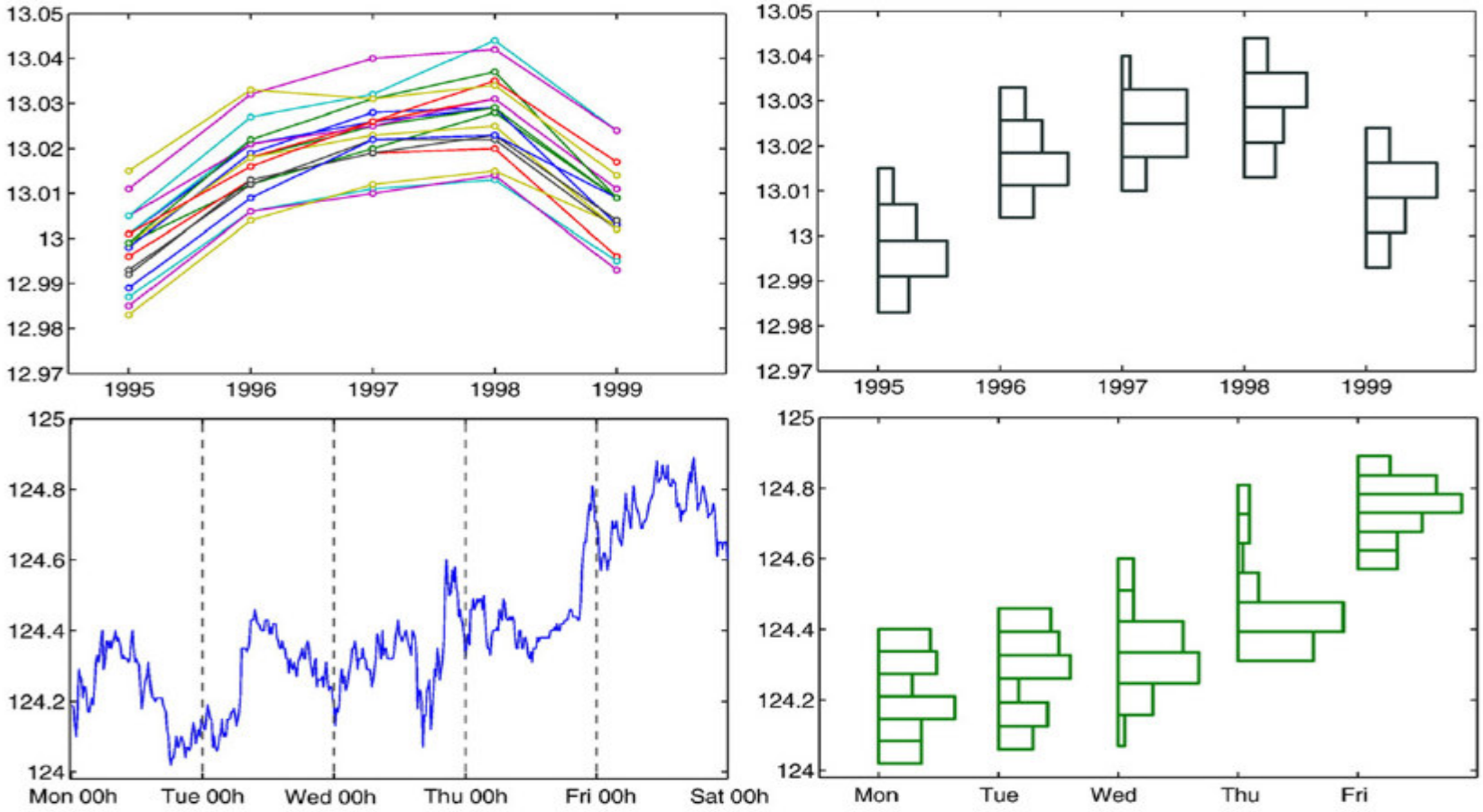


Fig. 1. Top: HTS (right) obtained from a set of distributions observed through time (left). Bottom: HTS (right) obtained from a time series of higher frequency (left).

Some Data Analysis Methods for Interval-Valued Data

- Central Tendency, Dispersion, Histograms: De Carvalho (1995), Billard and Diday (2003)
- Hierarchical and Pyramidal Clustering: Gowda and Diday (1991), Ichino and Yaguchi (2004), Guru and Kinaragi (2005), Brito and De Carvalho (2008)
- Fatorial Analysis: Chouakria et al (2007), Lauro and Palumbo (2000), Palumbo and Verde (2000)
- Time Series Analysis: Maia, De Carvalho and Ludermir (2008), Arroyo and Maté (2009), Maia and De Carvalho (2010)
- Multidimensional scaling: Groenen et al (2006)
- MLP: Munõz San Roque et al (2007)
- Regression: Billard and Diday (2000), Lima Neto and De Carvalho (2008), Lima Neto and De Carvalho (2010)

Dynamic Clustering Algorithm

- Diday (1971), Diday and Simon (1976)
- Dynamic clustering are relocation algorithms
- They optimizes (locally) an adequacy criterion
- The adequacy criterion express the best fitting between a partition and the set of prototypes which represent the clusters
- Prototypes can be a set of individuals, a mean vector, a regression model, a factorial plan, etc
- k-means like algorithm: If the criterion is the variance and the prototypes are mean vectors of the clusters

Dynamic Clustering Algorithm with Adaptive Distances

- Diday and Govaert (1974), Diday and Govaert (1977)
- There is a different distance for each cluster which changes at each iteration
- Main Steps
  - Initialization: Starts from a initial partition and alternates 3 steps
  - Step 1: Determination of the best prototypes
  - Step 2: Determination of the best distances
  - Step 3: Determination of the best partition
  - Repeat steps 1 to 3 until the convergence of the adequacy criterion

Partitioning Dinamic Clustering Algorithm for Interval-Valued Data

- Chavent and Lechevallier (2002), Souza and De Carvalho (2004), Chavent et al (2006), De Carvalho et al (2006-a, 2006b), Irpino and Verde (2008), De Carvalho and Lechevallier (2009-a, 2009-b)
- *E:* set of *n* examples described by *p* interval-valued variables
- Each example *i* is represented by a vector of intervals
  - $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})$, where $x_{ij}=[a_{ij},b_{ij}]$ (j=1,...,p)
- The prototype of cluster C_k is also represented as a vector of intervals
  - $\boldsymbol{y}_k = (y_{k1},\ldots,y_{kp})$, where $y_{kj} =[\alpha_{kj}, \beta_{kj}]$ (k=1,...,K)

Partitioning Clustering Algorithms

- These algorithms look for
  - a partition of E in K clusters $(C_1,\ldots,C_K)$ and
  - their corresponding prototypes $(\mathbf{y}_1,\ldots,\mathbf{y}_K)$
- such that an adequacy criterion W is (locally) minimized
- Adequacy criterion:

$$W = \sum_{k=1}^{K} \sum_{i \in C_k} d(\mathbf{x}_i, \mathbf{y}_k)$$

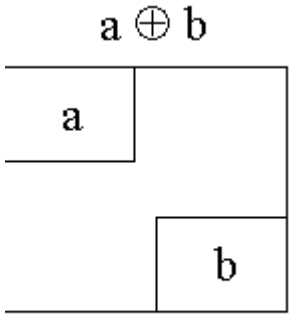
  - $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})$, where $x_{ij}=[a_{ij},b_{ij}]$ (i=1,...,n) (j=1,...,p)
  - $\mathbf{y}_k = (y_{k1},\ldots,y_{kp})$, where $y_{kj} =[\alpha_{kj}, \beta_{kj}]$ (k=1,...,K) (j=1,...,p)

Non-Adaptive Dissimilarity Functions Between Vectors of Intervals

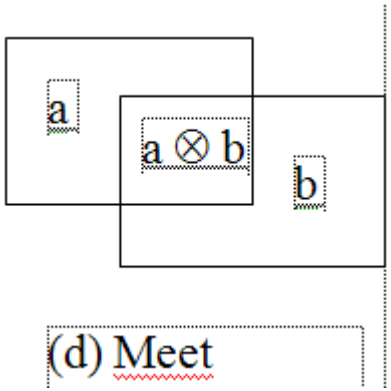
- Ichino and yaguchi (1994): $a = (A_1, \ldots, A_p)$; $b = (B_1, \ldots, B_p)$

$$d_q(a,b) = \left(\sum_{j=1}^{p} \left(\phi(A_j, B_j) \right)^q \right)^{\frac{1}{q}}, q \geq 1$$

$$\phi(A_j, B_j) = |A_j \oplus B_j| - |A_j \otimes B_j| + \gamma\left(2|A_j \otimes B_j| - |A_j| - |B_j|\right)$$



(a) Join



(d) Meet

Non-Adaptive Dissimilarity Functions Between Vectors of Intervals

- Non Adaptive Dissimilarity Functions
  - Euclidean, city-block, Hausdorff distances , Wasserstein distances
  - They are the same for all clusters
  - They do not change at each algorithm's iteration

$$d(\mathbf{x}_i, \mathbf{y}_k) = \sum_{j=1}^{p} d_j(x_{ij}, y_{kj})$$

Dissimilarity Functions Between Intervals - I

- city-block distances

$$d(x_{ij}, y_{kj}) = \left|a_{ij} - \alpha_{kj}\right| + \left|b_{ij} - \beta_{kj}\right|$$

- Hausdorff distances

$$d(x_{ij}, y_{kj}) = \max\left\{\left|a_{ij} - \alpha_{kj}\right|, \left|b_{ij} - \beta_{kj}\right|\right\}$$

- Euclidean distances

$$d(x_{ij}, y_{kj}) = (a_{ij} - \alpha_{kj})^2 + (b_{ij} - \beta_{kj})^2$$

Dissimilarity Functions Between Intervals - II

- Wasserstein distance

$$d(x_{ij}, y_{kj}) = (m_{ij} - m_{kj})^2 + \frac{1}{3}(r_{ij} - r_{kj})^2$$

$$m_{ij} = \frac{(a_{ij} + b_{ij})}{2} \qquad m_{kj} = \frac{(\alpha_{ij} + \beta_{ij})}{2}$$

$$r_{ij} = \frac{(b_{ij} - a_{ij})}{2} \qquad r_{kj} = \frac{(\beta_{ij} - \alpha_{ij})}{2}$$

Single Adaptive Dissimilarity Functions Between Vectors of Intervals - I

- Single Adaptive Dissimilarity Functions
  - Euclidean, city-block, Hausdorff distances, Wasserstein distances
  - They are parameterized by a weight vector

$$\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_p)$$

  - The weight vector is the same for all clusters
  - The weight vector changes at each algorithm's iteration

$$d_{\boldsymbol{\lambda}}(\mathbf{x}_i, \mathbf{y}_k) = \sum_{j=1}^{p} \lambda_j \, d_j(x_{ij}, y_{kj})$$

Single Adaptive Dissimilarity Functions Between Vectors of Intervals - II

- Single Adaptive Quadratic Distances
  - Mahalanobis distances
  - They are parameterized by a weight matrix **M**
  - The weight matrix is the same for all clusters
  - The weight matrix changes at each algorithm's iteration

$$d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{y}_k) = (\mathbf{x}_{iL} - \mathbf{y}_{kL})^T \mathbf{M} (\mathbf{x}_{iL} - \mathbf{y}_{kL}) + (\mathbf{x}_{iU} - \mathbf{y}_{kU})^T \mathbf{M} (\mathbf{x}_{iU} - \mathbf{y}_{kU})$$

$$\mathbf{x}_{iL} = (a_{i1}, \ldots, a_{ip}) \qquad \mathbf{x}_{iU} = (b_{i1}, \ldots, b_{ip})$$

$$\mathbf{y}_{kL} = (\alpha_{k1}, \ldots, \alpha_{kp}) \qquad \mathbf{y}_{kL} = (\beta_{k1}, \ldots, \beta_{kp})$$

Cluster Adaptive Dissimilarity Functions Between Vectors of Intervals

- Cluster Adaptive Dissimilarity Functions
  - Euclidean, city-block, Hausdorff, Wasserstein distances
  - They are parameterized by weight vectors

$$\boldsymbol{\lambda}_k = (\lambda_{k1}, \ldots, \lambda_{kp}) \; (k = 1, \ldots, K)$$

  - The weight vectors are different from one cluster to another
  - The weight vectors change at each algorithm's iteration

$$d_{\boldsymbol{\lambda}_k}(\mathbf{x}_i, \mathbf{y}_k) = \sum_{j=1}^{p} \lambda_{kj} \; d_j(x_{ij}, y_{kj})$$

Cluster Adaptive Dissimilarity Functions Between Vectors of Intervals - II

- Cluster Adaptive Quadratic Distances
  - Mahalanobis distances
  - They are parameterized by weight matrices $\mathbf{M}_k$ (k=1,…,K)
  - The weight matrices are different from one cluster to another
  - The weight matrices change at each algorithm's iteration

$$d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{y}_k) = (\mathbf{x}_{iL} - \mathbf{y}_{kL})^T \mathbf{M}_k (\mathbf{x}_{iL} - \mathbf{y}_{kL}) + (\mathbf{x}_{iU} - \mathbf{y}_{kU})^T \mathbf{M}_k (\mathbf{x}_{iU} - \mathbf{y}_{kU})$$

$$\mathbf{x}_{iL} = (a_{i1}, \ldots, a_{ip}) \qquad \mathbf{x}_{iU} = (b_{i1}, \ldots, b_{ip})$$

$$\mathbf{y}_{kL} = (\alpha_{k1}, \ldots, \alpha_{kp}) \qquad \mathbf{y}_{kL} = (\beta_{k1}, \ldots, \beta_{kp})$$

Step 1: Definition of the best prototypes - I

- The partition of E in K clusters and the distances are fixed
- The best prototype $\boldsymbol{y}_k = (y_k^1,\ldots,y_k^p)$ has the boundaries of the interval $y_k^j = [\alpha_k^j, \beta_k^j]$ calculated according to
  - City-block distances:

    $\alpha_{kj} = Median\ \{a_{ij} : i \in C_k\}$ and $\beta_{kj} = Median\ \{b_{ij} : i \in C_k\}$

  - Hausdorff distances (Chavent and Lechevallier 2002):
    $\alpha_{kj} = \mu_{kj} - \rho_{kj}$ and $\beta_{kj} = \mu_{kj} + \rho_{kj}$ where

    $\mu_{kj} = Median\ \{m_{ij} : i \in C_k\}$ and
    $\rho_{kj} = Median\ \{r_{ij} : i \in C_k\}$

Step 1: Definition of the best prototypes - II

- Euclidean and Mahalanobis distances:

  α_{kj}= *Average* $\{a_{ij} : i \in C_k\}$ and β_{kj} = *Average* $\{b_{ij} : i \in C_k\}$

- Wasserstein distances (Irpino and Verde (2008):
  $\alpha_{kj} = m_{kj} - r_{kj}$ and $\beta_{kj} = m_{kj} + r_{kj}$ where

  m_{kj} = *Average* $\{m_{ij}: i \in C_k\}$ and
  r_{kj} = *Average* $\{r_{ij} : i \in C_k\}$

Step 2: Definition of the Single best distances - I

- Euclidean, city-block, Haudorff, Wasserstein distances
- The partition and the prototypes are fixed
- The best vector of weights $\lambda = (\lambda^1, \ldots, \lambda^p)$, which minimizes the adequacy criterion W under,

$$\lambda_j > 0 \quad \text{and} \quad \prod_{j=1}^{p} \lambda_j = 1$$

has its components computed according to

$$\lambda_j = \frac{\left\{ \prod_{h=1}^{p} \left(\sum_{k=1}^{K} \left[\sum_{i \in C_k} d_h(x_{ih}, y_{ih}) \right] \right) \right\}^{\frac{1}{p}}}{\sum_{k=1}^{K} \left[\sum_{i \in C_k} d_j(x_{ij}, y_{ij}) \right]}$$

Step 2: Definition of the Single best distances - II

- Mahalanobis distances
- The best matrix of weights ***M***, which minimizes the adequacy criterion *J* under,

$$\det(\mathbf{M}) = 1$$

is computed according to

$$\mathbf{M} = [\det(\mathbf{Q})]^{\frac{1}{p}}\, \mathbf{Q}^{-1} \qquad \mathbf{Q} = \sum_{k=1}^{K} \mathbf{Q}_k$$

$$\mathbf{Q}_k = \sum_{i \in C_k} \left[(\mathbf{x}_{iL} - \mathbf{y}_{kL})(\mathbf{x}_{iL} - \mathbf{y}_{kL})^T + (\mathbf{x}_{iU} - \mathbf{y}_{kU})(\mathbf{x}_{iU} - \mathbf{y}_{kU})^T \right]$$

Step 2: Definition of the Cluster best distances - I

- Euclidean, city-block Haudorff, Wasserstein distances
- The partition and the prototypes are fixed
- The best vector of weights $\lambda_k = (\lambda_{k1}, \ldots, \lambda_{kp})$, which minimizes the adequacy criterion J under,

$$\lambda_{kj} > 0 \quad \text{and} \quad \prod_{j=1}^{p} \lambda_{kj} = 1$$

has its components calculated according to

$$\lambda_{kj} = \frac{\left\{ \prod_{h=1}^{p} \left(\sum_{i \in C_k} d_h(x_{ih}, y_{ih}) \right) \right\}^{\frac{1}{p}}}{\left[\sum_{i \in C_k} d_j(x_{ij}, y_{ij}) \right]}$$

Step 2: Definition of the Cluster best distances - II

- Mahalanobis distances
- The best matrices of weights $\boldsymbol{M}_k$ (k=1,...,K), which minimizes the adequacy criterion *J* under,

$$\det(\mathbf{M}_k) = 1$$

is computed according to

$$\mathbf{M}_k = [\det(\mathbf{Q}_k)]^{\frac{1}{p}} \mathbf{Q}_k^{-1}$$

$$\mathbf{Q}_k = \sum_{i \in C_k} \left[(\mathbf{x}_{iL} - \mathbf{y}_{kL})(\mathbf{x}_{iL} - \mathbf{y}_{kL})^T + (\mathbf{x}_{iU} - \mathbf{y}_{kU})(\mathbf{x}_{iU} - \mathbf{y}_{kU})^T \right]$$

Step 3: Definition of the best partition

- The prototypes and the distances are fixed
- The best partition $(C_1,\ldots,C_K)$, which minimizes the adequacy criterion J, has its clusters updated according to

$$C_k = \{i \in E : d(\mathbf{x}_i, \mathbf{y}_k) \leq d(\mathbf{x}_i, \mathbf{y}_h), \forall h \neq k\}$$

Cluster and partition interpretation

- Important step in clustering analysis
- For usual quantitative data, Celeux et al (1989) introduced a family of indices for cluster and partition interpretation
- For this case, the dispersions decompose into the dispersions within clusters plus the dispersions between clusters.
- Chavent et al (2006) presented an approach to measure the partition (or cluster) quality which holds even if the dispersions does not decomposes as before

Cluster and partition interpretation

- Let us consider
  - A partition $C = (C_1,\ldots,C_K)$ of E in K clusters of cardinality n_k
  - Each cluster has a prototype $\mathbf{y}_k = (y_k^1,\ldots,y_k^p)$
  - Let us consider a overall prototype of E as $\mathbf{y} = (y^1,\ldots,y^p)$
- Overall Dispersion

$$T = \sum_{i=1}^{n} d(\mathbf{x}_i, \mathbf{y}) = \sum_{k=1}^{K} \sum_{i \in C_k} d(\mathbf{x}_i, \mathbf{y})$$

Overall Prototype - I

- The overall prototype $\boldsymbol{y} = (y_1,\ldots,y_p)$ has the boundaries of the interval $y_j = [\alpha_j, \beta_j]$ calculated according to
  - City-block distances:

    $\alpha_j = Median\ \{a_{ij}\ for\ all\ i \in E\}$ and $\beta_j = Median\ \{b_{ij}\ for\ all\ i \in E\}$

  - Hausdorff distances: $\alpha_j = \mu_j - \rho_j$ and $\beta_j = \mu_j + \rho_j$ where

    $\mu_j = median\ \{m_{ij}\ for\ all\ i \in E\}$ and

    $\rho_j = median\ \{r_{ij}\ for\ all\ i \in E\}$

Overall Prototype - II

- The overall prototype $\boldsymbol{y} = (y_1,\ldots,y_p)$ has the boundaries of the interval $y_j = [\alpha_j, \beta_j]$ calculated according to

  - Euclidean and Mahalanobis distances:

    α_j = *Average* $\{a_{ij}$ *for all* $i \in E\}$ and β_j = *Average* $\{b_{ij}$ *for all* $i \in E\}$

  - Wasserstein distances:
    $\alpha_{kj} = m_{kj} - r_{kj}$ and $\beta_{kj} = m_{kj} + r_{kj}$ where

    m_{kj} = *Average* $\{m_{ij}: i \in C_k$ *for all* $i \in E\}$ and
    r_{kj} = *Average* $\{r_{ij} : i \in C_k$ *for all* $i \in E\}$

Overall dispersion - I

$$T = \sum_{i=1}^{n} d(\mathbf{x}_i, \mathbf{y}) = \sum_{k=1}^{K} \sum_{i \in C_k} d(\mathbf{x}_i, \mathbf{y})$$

- It decomposes
  - into the sum of the cluster-specific overall dispersion

$$T = \sum_{k=1}^{K} T_k \qquad T_k = \sum_{i \in C_k} d(\mathbf{x}_i, \mathbf{y})$$

Overall dispersion - II

- into the sum of the variable-specific overall dispersion (except for the Mahalanobis distance)

$$T = \sum_{j=1}^{p} T_j \qquad T_j = \sum_{k=1}^{K} \sum_{i \in C_k} d_j(x_{ij}, y_j)$$

$$T_j = \sum_{k=1}^{K} \sum_{i \in C_k} \lambda_j d_j(x_{ij}, y_j) \qquad T_j = \sum_{k=1}^{K} \sum_{i \in C_k} \lambda_{kj} d_j(x_{ij}, y_j)$$

- into the sum of the variable-cluster-specific overall dispersion (except for the Mahalanobis distance)

$$T = \sum_{k=1}^{K} \sum_{j=1}^{p} T_{kj} \qquad T_{kj} = \sum_{i \in C_k} d_j(x_{ij}, y_j)$$

$$T_{kj} = \sum_{i \in C_k} \lambda_j d_j(x_{ij}, y_j) \qquad T_{kj} = \sum_{i \in C_k} \lambda_{kj} d_j(x_{ij}, y_j)$$

Within-cluster dispersion - I

$$W = \sum_{k=1}^{K} \sum_{i \in C_k} d(\mathbf{x}_i, \mathbf{y}_k)$$

- It decomposes
  - into the sum of the cluster-specific within-cluster dispersion

$$W = \sum_{k=1}^{K} W_k \qquad W_k = \sum_{i \in C_k} d(\mathbf{x}_i, \mathbf{y}_k)$$

Within-cluster dispersion - II

- into the sum of the variable-specific within-cluster dispersion (except for the Mahalanobis distance)

$$W = \sum_{j=1}^{p} W_j \qquad W_j = \sum_{k=1}^{K} \sum_{i \in C_k} d_j(x_{ij}, y_{kj})$$

$$W_j = \sum_{k=1}^{K} \sum_{i \in C_k} \lambda_j d_j(x_{ij}, y_{kj}) \qquad W_j = \sum_{k=1}^{K} \sum_{i \in C_k} \lambda_{kj} d_j(x_{ij}, y_{kj})$$

- into the sum of the variable-cluster-specific within-cluster dispersion (except for the Mahalanobis distance)

$$W = \sum_{k=1}^{K} \sum_{j=1}^{p} W_{kj} \qquad W_{kj} = \sum_{i \in C_k} d_j(x_{ij}, y_{kj})$$

$$W_{kj} = \sum_{i \in C_k} \lambda_j d_j(x_{ij}, y_{kj}) \qquad W_{kj} = \sum_{i \in C_k} \lambda_{kj} d_j(x_{ij}, y_{kj})$$

Between-cluster dispersion - I

$$B = \sum_{k=1}^{K} n_k d(\mathbf{y}_k, \mathbf{y})$$

- It decomposes
  - into the sum of the cluster-specific between-cluster dispersion

$$B = \sum_{k=1}^{K} B_k \qquad B_k = n_k d(\mathbf{y}_k, \mathbf{y})$$

Between-cluster dispersion - II

- into the sum of the variable-specific between-cluster dispersion (except for the Mahalanobis distance)

$$B = \sum_{j=1}^{p} B_j \qquad B_j = \sum_{k=1}^{K} n_k d_j(y_{kj}, y_k)$$

$$B_j = \sum_{k=1}^{K} n_k \lambda_j d_j(y_{kj}, y_k) \qquad B_j = \sum_{k=1}^{K} n_k \lambda_{kj} d_j(y_{kj}, y_k)$$

- into the sum of the variable-cluster-specific between-cluster dispersion (except for the Mahalanobis distance)

$$B = \sum_{k=1}^{K} \sum_{j=1}^{p} B_{kj} \qquad B_{kj} = n_k d_j(y_{kj}, y_j)$$

$$B_{kj} = n_k \lambda_j d_j(y_{kj}, y_j) \qquad B_{kj} = n_k \lambda_{kj} d_j(y_{kj}, y_j)$$

Relations Between Overall, Within and Between Dispersion - I

- $T = B + W$ (Euclidean, Mahalanobis and Wasserstein distances)
- $T_k = B_k + W_k$ for k=1,…,K (Euclidean, Mahalanobis and Wasserstein distances)
- $T_j = B_j + W_j$ for j=1,…,p (Euclidean and Wasserstein distances)
- $T_{kj} = B_{kj} + W_{kj}$ for k=1,…,K and j=1,…,p (Euclidean and Wasserstein distances)

Relations Between Overall, Within and Between Dispersion - II

- For all distances, the following relations hold:
  - $T > W$
  - $T_k > W_k$ for k=1,…,K
  - $T_j > W_j$ for j=1,…,p
  - $T_{kj} > W_{kj}$ for k=1,…,K and j=1,…,p

Some Partition Interpretation Indices

- ***Overall heterogeneity index***: it measures the quality of a partition $C = (C_1,\ldots,C_K)$ of Ω in K clusters

$$Q(C) = \frac{T - W}{T} = 1 - \frac{W}{T}$$

$$0 \leq Q(C) \leq 1$$

- Rule: a partition C in K clusters is better than a partition C' in K clusters if $Q(C) > Q(C')$

Some Partition Interpretation Indices

- ***Overall heterogeneity index with respect to single variables***: it measures the quality of a partition $C = (C_1,\ldots,C_K)$ of Ω in K clusters concerning the *j-th* variables

$$Q_j(C) = \frac{T_j - W_j}{T_j} = 1 - \frac{W_j}{T_j}$$

- This index measures the discriminant power of the *j-th* variable in the partition $C = (C_1,\ldots,C_K)$
- The comparison between Q_j and Q evaluates if the discriminant power of the *j-th* variable is above or below the average

Some Cluster Interpretation Indices

- ***Cluster heterogeneity indices***
  - The proportion of the overall dispersion in cluster C_k

$$T(k) = \frac{T_k}{T} \qquad \sum_{k=1}^{K} T(k) = 1$$

  - The relative contribution of cluster C_k to the overall within-cluster dispersion

$$W(k) = \frac{W_k}{W} \qquad \sum_{k=1}^{K} W(k) = 1$$

- A large value of *W(k)* indicates that cluster C_k is relatively heterogeneous in comparison with the other clusters

Some Cluster Interpretation Indices

- ***Cluster heterogeneity indices***
  - The quality of a cluster C_k

$$Q(C_k) = \frac{T_k - W_k}{T_k} = 1 - \frac{W_k}{T_k}$$

- This indice measures the gain of homogeneity of the cluster C_k obtained when replacing the overall prototype $\boldsymbol{y}$ by the prototype $\boldsymbol{y}_k$ in the calculation of the homogeneity

Some Cluster Interpretation Indices

- ***Cluster heterogeneity indices with respect to sigle variables***
  - The quality of a cluster C_k concerning the *j-th* variable

$$Q_j(C_k) = \frac{T_{kj} - W_{kj}}{T_{kj}} = 1 - \frac{W_{kj}}{T_{kj}}$$

- Rule: the *j-th* variable characterizes the cluster C_k if $Q_j(C_k) > Q(C_k)$

City Temperature Interval-Valued Data Set

Available at http://www.bbc.co.uk/weather/world/city_guides/.

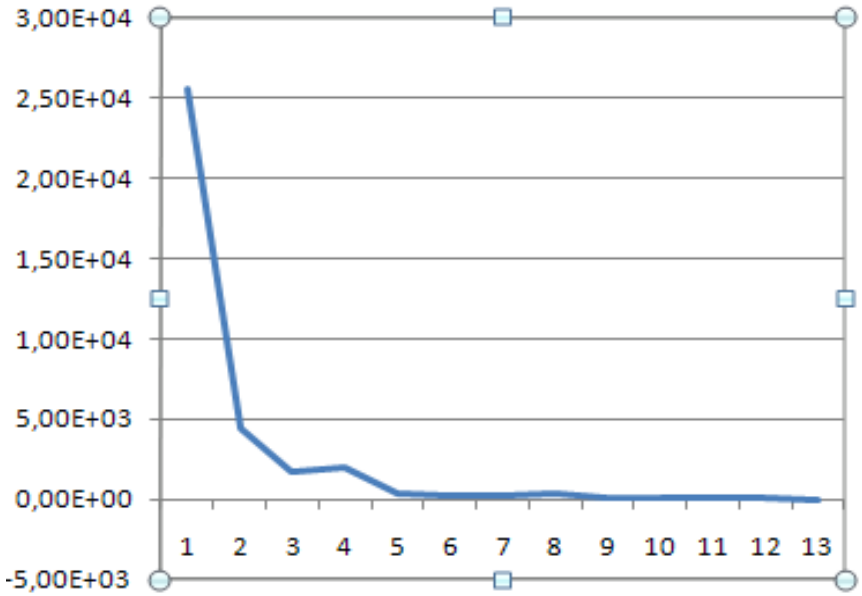
- **gives the average minimal and average maximal monthly temperatures of cities in degrees centigrade**
- **the data set consists of 503 cities described by 12 interval-valued variables.**
- **In this example, the algorithm uses single adaptive city-block distances**
- **For a fixed number K = {1,…,10}, the algorithm is run 100 times and the best result is selected**

City Temperature Interval-Valued Data Set

	January	February	...	November	December
Amsterdam	$[-4, 4]$	$[-5, 3]$	...	$[1, 10]$	[-1, 4]
Athens	$[6, 12]$	$[6, 12]$	...	$[11, 18]$	$[8, 14]$
...	...	...	...	...	...
Mauritius	$[22, 28]$	$[22, 29]$	...	$[19, 27]$	$[21, 28]$
...	...	...	...	...	...
Vienna	$[-2, 1]$	$[-1, 3]$	...	$[2, 7]$	$[1, 3]$
Zurich	$[-11, 9]$	$[-8, 15]$	...	$[0, 19]$	$[-11, 8]$

Determination of the number of clusters

SPAD Software, Gomes Da Silva (2009): peaks on the graph of the ``second order differences'' of the clustering criterion: $W^{(K-1)} + W^{(K+1)} - 2\,W^{(K)}$ (K=2,…,9)



Partition in 5 clusters

Cluster 1: the cities have very cold temperatures in winter similar to that of northern and eastern Europe

Cluster 2: the cities have temperatures similar to that of southern Europe

Cluster 3: the cities have temperatures similar to that of western and central Europe.

Cluster 4: the cities have temperatures similar to that of cities located in the southern hemisphere.

Cluster 5: the cities have a tropical climate and warm to hot temperatures

Partition in 5 clusters

Partition quality

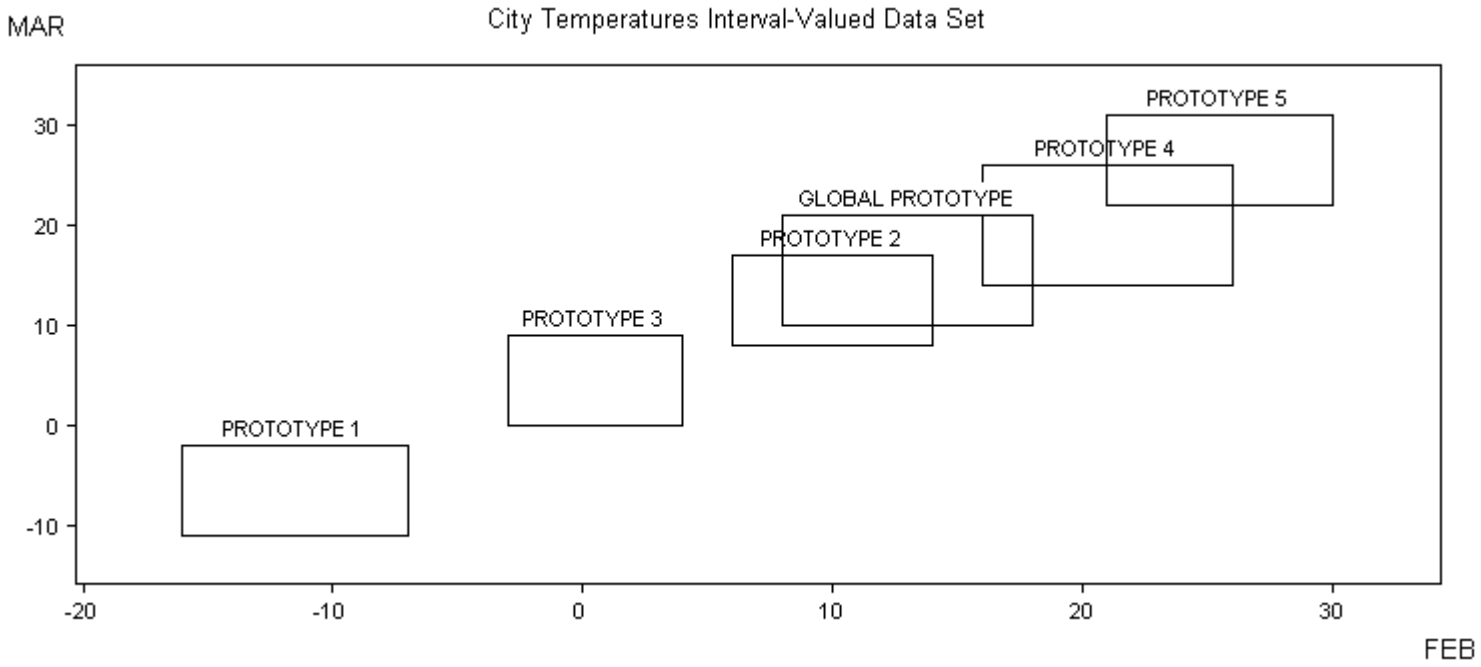
$Q(C) = 62.82$

Partition quality / Variable j

Variable	Qj(C)
January	69.49
February	70.68
March	71.20
April	66.54
May	56.32
June	46.81
July	41.09
August	42.65
September	52.66
October	64.16
November	69.75
December	69.78

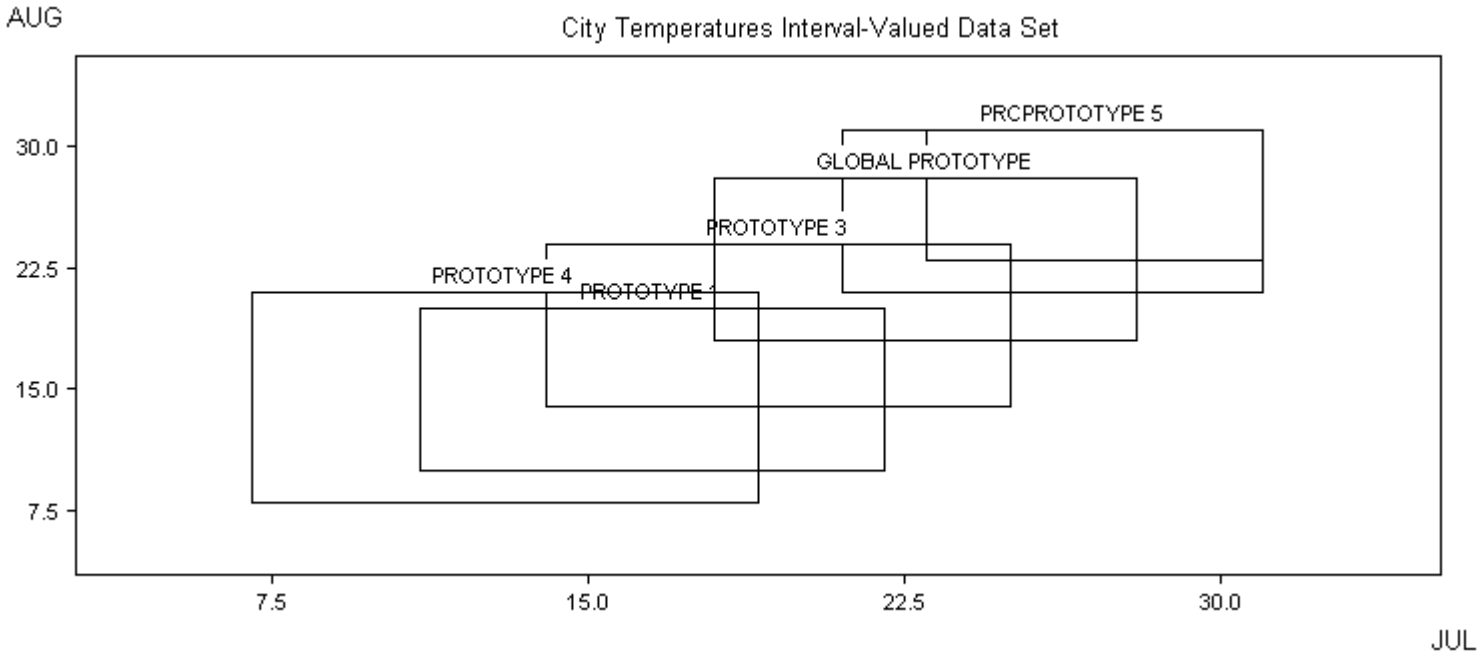
⇨ **Discriminant power of the months**

Partition in 5 clusters



Most discriminant months

Partition in 5 clusters



Least discriminant months

Partition in 5 clusters

Cluster quality $Q(C_1)$=71.65 $Q(C_2)$=16.14 $Q(C_3)$=68.67

Cluster quality/ Variable j

Variable	$Q_j(C_1)$	$Q_j(C_2)$	$Q_j(C_3)$
January	76.15	23.10	76.59
February	78.28	20.51	77.49
March	79.22	24.80	79.22
April	70.83	1.46	73.40
May	62.94	6.17	52.02
June	56.27	15.53	34.50
July	53.51	22.33	25.32
August	57.73	22.39	30.93
September	66.86	16.84	52.84
October	71.52	1.47	72.27
November	74.74	10.52	80.76
December	74.72	21.42	78.86

Partition in 5 clusters

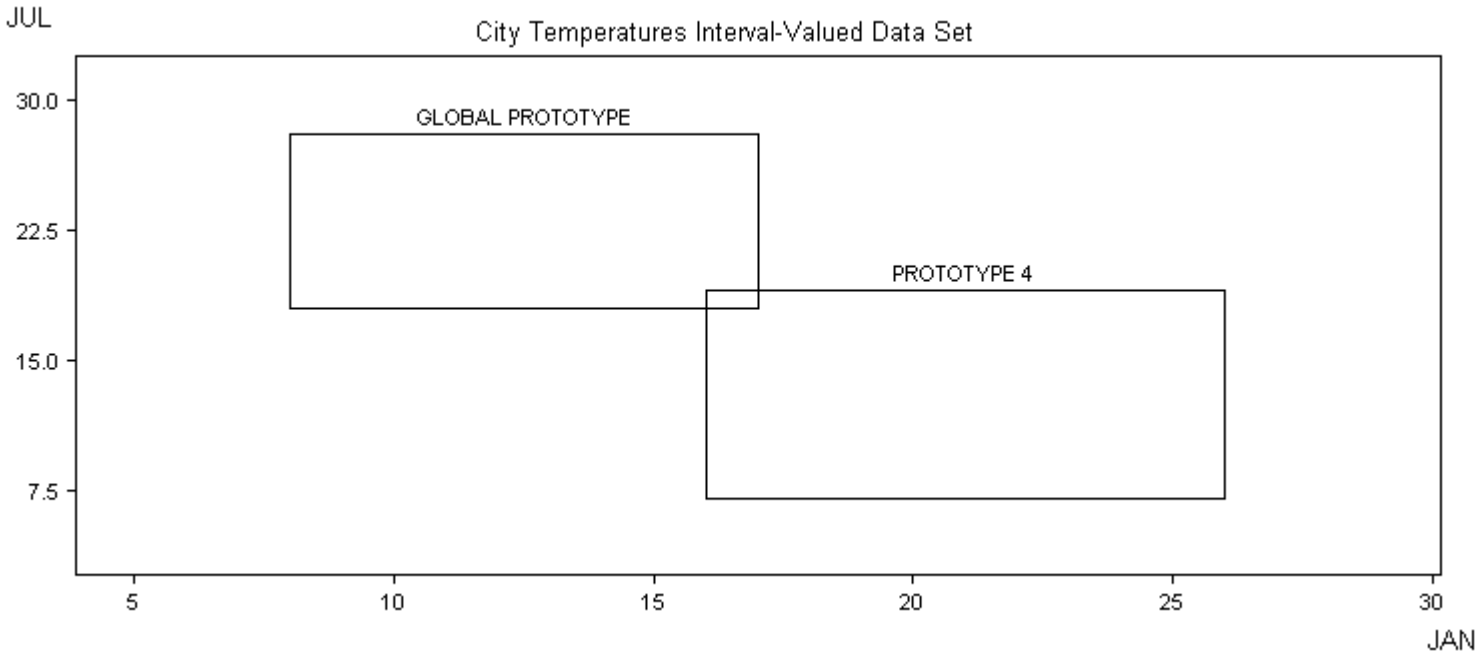
$Q(C_4)$=46.01 $Q(C_5)$=70.78

Cluster quality/

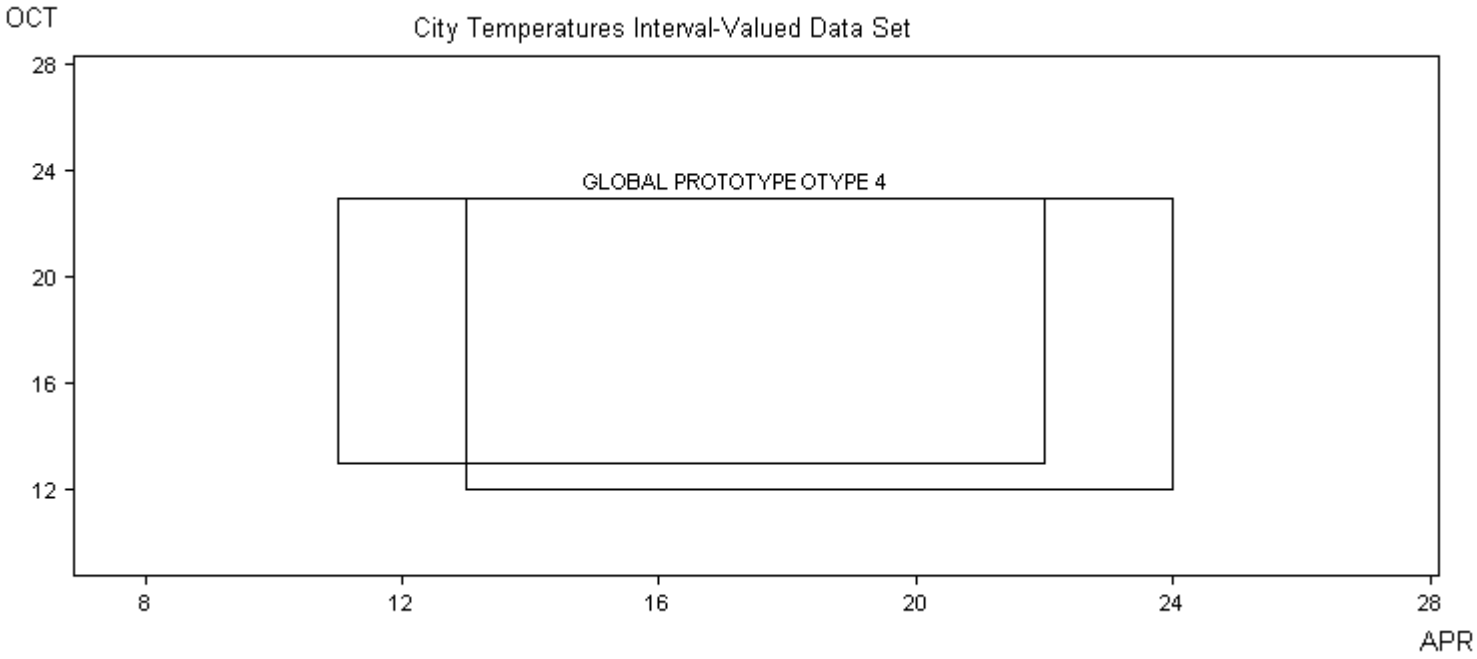
Variable j

Variable	$Q_j(C_4)$	Qj(C5)
January	63.04	72.77
February	62.12	73.93
March	44.01	75.79
April	10.09	78.64
May	19.15	73.18
June	50.84	57.84
July	62.45	36.85
August	60.29	39.19
September	37.40	63.14
October	1.07	77.99
November	22.18	78.50
December	55.00	73.85

Partition in 5 clusters



Partition in 5 clusters



Some Remarks

- Interval modelling
- Others distance functions
- Set-valued, list-valued, Histogram-valued data
- Mixed-feature type symbolic data

ARTICLE IN PRESS

Pattern Recognition Letters xxx (2009) xxx–xxx

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

ELSEVIER

Unsupervised pattern recognition models for mixed feature-type symbolic data

Francisco de A.T. de Carvalho *, Renata M.C.R. de Souza

Some Remarks

- Others classification structures: overlapping clusters
- Clustering mixtures

Bibliography

[1] H. H. Bock and E. Diday, Analysis of Symbolic Data, Springer-Verlag, Heidelberg, 2000

[2] L. Billard and E. Diday, Symbolic Data Analysis. Conceptual Statistics and Data Mining. Wiley, Chichester, 2006.

[3] E. Diday and M. Noirhome, Symbolic Data Analysis and the SODAS Software, Wiley, 2008

[4] E. Diday, 1971. La méthode des Nueés dynamiques. Rev. Statist. Appl. 19 (2), 19–34

[5] E. Diday, G. Govaert, 1977. Classification automatique avec distances adaptatives. RAIRO Inform. Computer Sci. 11 (4), 329–349.

[6] E. Diday, J.J. Simon, 1976. Clustering analysis. In: Fu, K.S. (Ed.), Digital Pattern Recognition. Springer-Verlag, Heidelberg, pp. 47–94.

[7] G. Celeux, E. Diday, G. Govaert, Y. Lechevallier, H. Ralambondrainy, 1989. Classification Automatique des Donne´es. Bordas, Paris.

Bibliography

Computational Statistics (2004)

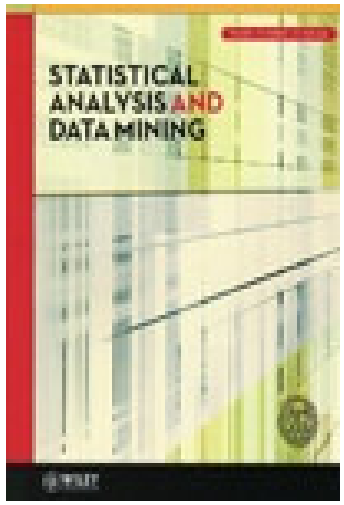
Special Issue on Interval Data

Edited by Francesco Palumbo

Statistical Analysis and Data Mining (2010)

Special Issue on Symbolic Data Analysis

Edited by Lynne Billard



SODAS and ASSO projects: http://www.info.fundp.ac.be/asso/objective.htm

Bibliography

[8] M. Chavent, Y. Lechevallier, 2002. Dynamical clustering algorithm of interval data: optimization of an adequacy criterion based on Hausdorff distance, in: H.H. Sokolowsky, K. Bock, A. Jaguja (Eds.), Classification, Clustering and Data Analysis (IFCS2002), Springer, Berlin, pp. 53–59.

[9] [13] R.M.C.R. Souza, F.A.T. De Carvalho, 2004. Clustering of interval data based on city-block distances. Pattern Recognition Letters, 25 (3), 353–365.

[10] M. Chavent, F.A.T. De Carvalho, Y. Lechevallier, R. Verde 2006. New clustering methods for interval data. Computational Statistics, 21 (2), 211-230

[11] F.A.T. De Carvalho, P. Brito, H.H. Bock, 2006. Dynamic clustering for interval data based on L2 distance. Computational Statistics, 21(2), 231-250.

Bibliography

[12] F.A.T. De Carvalho, R.M.C. R. Souza, M. Chavent and Y. Lechevallier, 2006. Adaptive Hausdorff distances and dynamic clustering of symbolic data. Pattern Recognition Letters, 27 (3), 167–179.

[13] A. Irpino and R. Verde, 2008. Dynamic clustering of interval data using a Wasserstein-based distance. Pattern Recognition Letters, 29, 1648–1658

[14] A. Gomes Da Silva, 2009. Analyse des données évolutives : application aux données d'usage du Web. These de Doctorat. Université Paris-IX Dauphine

[15]] F.A.T. De Carvalho and Y. Lechevallier, 2009. Partitional Clustering algorithms for symbolic interval data based on single adaptive distances. Pattern Recognition, 42, 1223–1236

[16] F.A.T. De Carvalho and Y. Lechevallier, 2009. Dynamic clustering of interval-valued data based on adaptive quadratic distances. IEEE Transactions on Systems, Man and Cybernetics – Part A: Systems and Humans, 39, 1295–1306

Bibliography

[17] J. Arroy, C. Maté. Forecasting histogram time series with k-nearest neighbours methods, International Journal of Forecasting, 25,192–207, 2009

[18] A.L.S. Maia, F.A.T. De Carvalho, T.B. Ludermir. Forecasting models for interval-valued time series. Neurocomputing, 71, 3344-3352, 2008.

[19] A.L.S. Maia, F.A.T. De Carvalho. Holt's Exponential Smoothing and Neural Network Models for Forecasting Interval-Valued Time Series. International Journal of Forecasting (Accepted)

[20] Billard, L. and Diday, E. 2000. Regression Analysis for Interval-Valued Data. In: Data Analysis, Classification and Related Methods: Proceedings of IFCS 2000, Springer, 369-374.

[21] Billard, L. and Diday, E. 2003. From the statistics of data to the statistics of knowledge: Symbolic Data Analysis. Journal of American Statistical Association, 98, (462), 470-487.

Bibliography

[22] Cazes, P., Chouakria, A., Diday, E. and Schektman, S. 1997. Extension de l'analyse en composantes principales des donnes de type intervalle. Revue de Statistique Aplique, XLV (3), 5–24.

[23] Chavent, M. 1998. A monothetic clustering method. Pattern Recognition Letters, 19, 989–996.

[24] De Carvalho, F. A. T. 1995. Histograms In Symbolic Data Analysis. Annals of Operations Research, 55, 229–322.

[25] Gowda, K. C. and Diday, E. 1991. Symbolic clustering using a new dissimilarity measure. Pattern Recognition, 24, (6), 567–578.

[26] Guru, D.S. and Kiranagi, B.B. 2005. Multivalued type dissimilarity measure and concept of mutual dissimilarity value for clustering symbolic patterns. Pattern Recognition, 38, 151–256

[27] Groenen, P.J.F., Winsberg, S., Rodrigues, O. and Diday, E. 2006. I-Scal: Multidimensional scaling of interval dissimilarities. Computational Statistics and Data Analysis, 51 (1), 360–378.

Bibliography

[28] Cazes, P., Chouakria, A., Diday, E. and Schektman, S. 1997. Extension de l'analyse en composantes principales des donnes de type intervalle. Revue de Statistique Aplique, XLV (3), 5–24.

[29] Lauro, N.C. and Palumbo, F. 2000. Principal component analysis of interval data: a symbolic data analysis approach. Computational Statistics, 15 (1), 73–87.

[30] Palumbo, F. and Verde, R. 2000. Non-symmetrical factorial discriminant analysis for symbolic objects. Applied Stochastic Models in Business and Industry, 15 (4), 419–427.

[31] Maia, A.L.S., De Carvalho, F.A.T., Ludermir, T. Forecasting models for interval-valued time series. Neurocomputing 71 (16-18), 3344-3352

[32] Munõz San Roque et al, 2007. iMLP: applying multilayer perceptron to interval-valued data. Neural Processing Letters 25, 157–169.

[33] Lima Neto, E. A. ; ., De Carvalho, F.A.T. Centre and Range Method for Fitting a Linear Regression Model to Symbolic Interval Data. Computational Statistics & Data Analysis, v. 52, p. 1500-1515, 2008.

Thank you