SCAN ME

Slides & Papers

https://github.com/amarkos/opaworkshop

Distance-Based Methods for Mixed-Type Data: Advances & Applications*

Angelos Markos

Democritus University of Thrace

**Joint work with M. van de Velden, A. Iodice D' Enza, C. Cavicchia, E. Costa, , I. Papatsouma*

OUTLINE

- Introduction/Motivation
- Analytical Approaches for Mixed Data
- Foundations of Distance-Based Methods
- Distances by Data Type
  - * Numerical
  - * Ordinal
  - * Categorical
  - * Mixed
- Distance-Based Algorithms
- Discussion/Open Problems

Distance-Based Methods for Mixed-Type Data: Advances & Applications*

Angelos Markos
Democritus University of Thrace

**Joint work with M. van de Velden, A. Iodice D' Enza, C. Cavicchia, E. Costa, , I. Papatsouma*

Motivating Example: InsideAirbnb.com

Listing ID	Price (€)	Rating	Bedrooms	Property Type	Neighborhood	Amenities
42781	89.5	4.87	2	Apartment	Plaka	["Wifi", "Kitchen", "AC"]
37159	145	4.92	3	Villa	Glyfada	["Pool", "Wifi", "Parking"]
18472	65.2	3.75	1	Studio	Exarcheia	["Wifi"]
65120	112.8	4.55	2	Apartment	Monastiraki	["Wifi", "Kitchen"]
29384	205	4.96	4	House	Kolonaki	["Pool", "Wifi", "Kitchen", "Parking"]

Mixed-Type Variables in this Dataset:

Numerical: Price (€), Rating
Ordinal: Bedrooms
Nominal: Property Type, Neighborhood
Array/Categorical Set: Amenities

Goals: identify market segments, predict/explain guest ratings, optimize pricing strategies etc

Challenges: incompatible measurement scales, different distributional properties, complex semantic relationships between variable types

Analytical Approaches for Mixed-Type Data

Model-Based Approaches
- Based on probabilistic assumptions
- Parameterize variable distributions

Examples:
→ Latent class models
→ Model-based clustering
→ (Bayesian) Mixture models

- Strengths: Statistical inference, uncertainty quantification, formal model selection
- Challenges: Distributional assumptions, computational complexity, interpretability issues

Distance-Based Methods ← Our Focus
- Based on (dis)similarity between objects or between objects and representative objects

Examples:
→ Hierarchical/Partitional clustering
→ Multidimensional Scaling
→ K-Nearest Neighbors

- Strengths: No strict distributional assumptions, flexibility, interpretability
- Challenges: Choice of dissimilarity measure, dimensionality challenges, nouncertainty quantification, fragmented literature with ad hoc implementations

For an overview, see van de Velden et al. (2019).

van de Velden, M., Iodice D' Enza, A., & Markos, A. (2019). Distance-based clustering of mixed data. *Wiley Interdisciplinary Reviews: Computational Statistics*, *11*(3), e1456.

Foundations of Distance-Based Methods

A ***dissimilarity*** is a function $d : \mathcal{X}^2 \mapsto \mathbb{R}_0^+, \mathcal{X}$ *being the object space, so that* $d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x}) \geq 0$ *and* $d(\mathbf{x}, \mathbf{x}) = 0$ *for* $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.

A dissimilarity fulfiling the triangle inequality

$d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z}) \geq d(\mathbf{x}, \mathbf{z}), \mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{X},$

is called a **distance** or **metric**.

Dissimilarity Based on Variables

Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ be a set of N objects where each $\mathbf{x}_i = \{x_{i1}, x_{i2}, \dots, x_{ip}\}$ is a vector of p variables of mixed-type, $p = p_n + p_c + p_o$, where p_n, p_c, p_o, are the number of **numerical**, **categorical** and **ordinal** variables, respectively.

A general multivariate mixed-variable dissimilarity between two objects i and l:

$$d(\mathbf{x}_i, \mathbf{x}_l) = \sum_{j=1}^{p_n} w_j d_{j_n}(x_{ij}, x_{lj}) + \sum_{j=1}^{p_c} w_j d_{j_c}(x_{ij}, x_{lj}) + \sum_{j=1}^{o} w_j d_{j_o}(x_{ij}, x_{lj})$$

and w_j is a weight corresponding to each of these functions.

Foundations of Distance-Based Methods

Dissimilarity Matrix

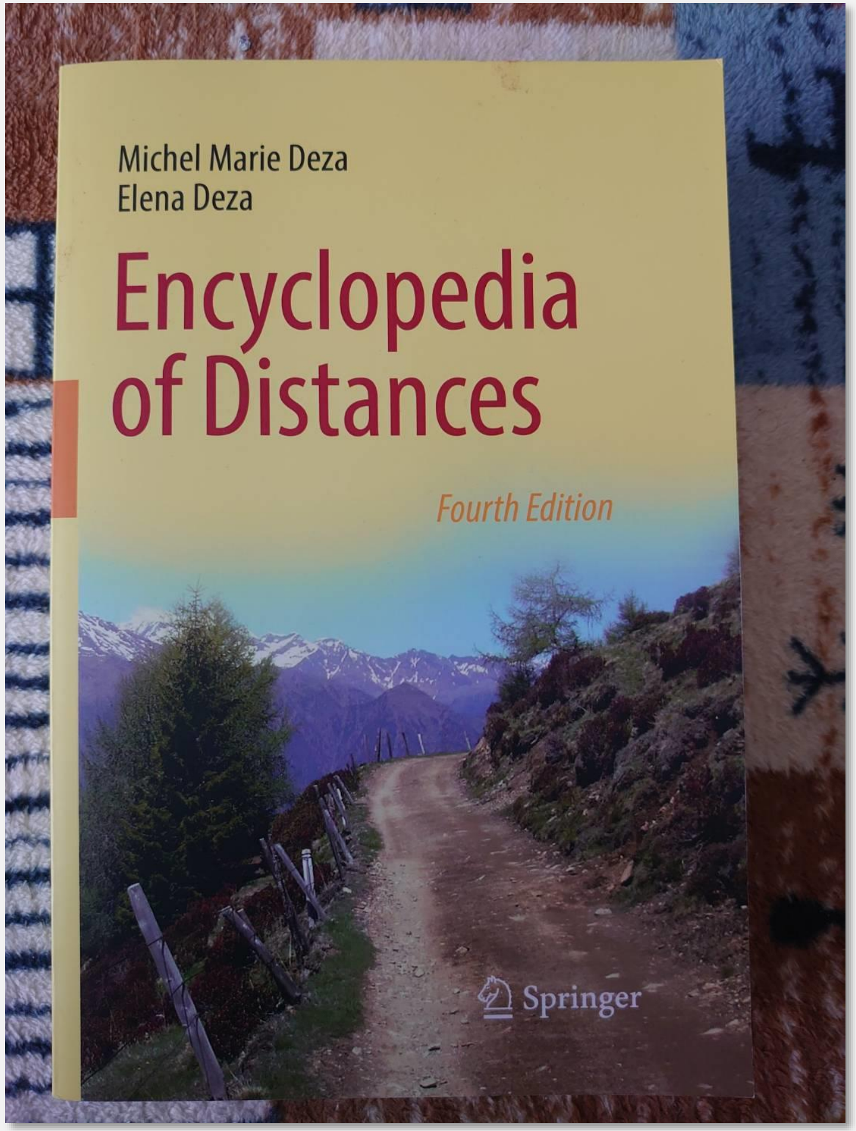
An $N \times N$ matrix $\mathbf{D}$ representing dissimilarities between pairs of objects.

Many clustering/classifications/visualization algorithms are directly applied to **D.**

Most algorithms require:

- Non-negativity, $d(\mathbf{x}_i, \mathbf{x}_l) \geq 0$
- Zero diagonal elements, $d(\mathbf{x}_i, \mathbf{x}_i) = 0$
- Symmetry, $d(\mathbf{x}_i, \mathbf{x}_l) = d(\mathbf{x}_l, \mathbf{x}_i)$

	42781	***37159***	***18472***	***65120***	...
42781	**0**	0.84	0.42	0.39	...
37159	0.84	**0**	0.58	0.66	...
18472	0.42	0.58	**0**	0.54	...
65120	0.39	0.66	0.54	**0**	...
...	...	...	...	**...**	...



Earth Mover's Distance (Wasserstein distance)

Suppose you have two distributions: **one is a bunch of piles of dirt, and the other is a set of holes**.

The Earth Mover's Distance is *the minimum effort required to move the dirt into the holes, considering both the amount moved and the distance travelled*.

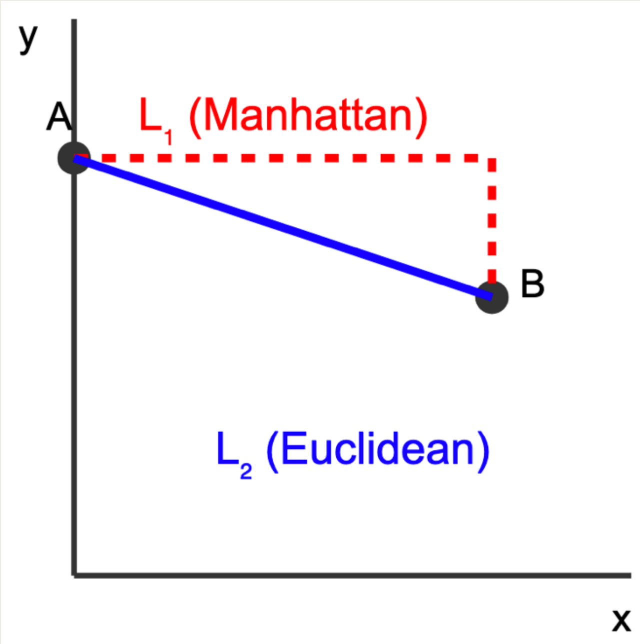
It's often used in image recognition, where one distribution (e.g., colors in an image) is transformed into another.

Distances for Numerical Data

The dissimilarity measure directly reflects the magnitude of difference between values.

The Minkowski (L_q)-distance $$d_{L_q}(\mathbf{x}_i, \mathbf{x}_l) = \sqrt[q]{\sum_{j=1}^{p} d_j(x_{ij}, x_{lj})^q},$$

where $d_j(\mathbf{x}, \boldsymbol{y}) = |\mathbf{x} - \mathbf{y}|$.



Special Cases

- Manhattan distance (L_1): $d_{L_1} = \sum |x_{ij} - x_{lj}|$
- Euclidean distance (L_2): $d_{L_2} = \sqrt{\sum (x_{ij} - x_{lj})^2}$

Properties

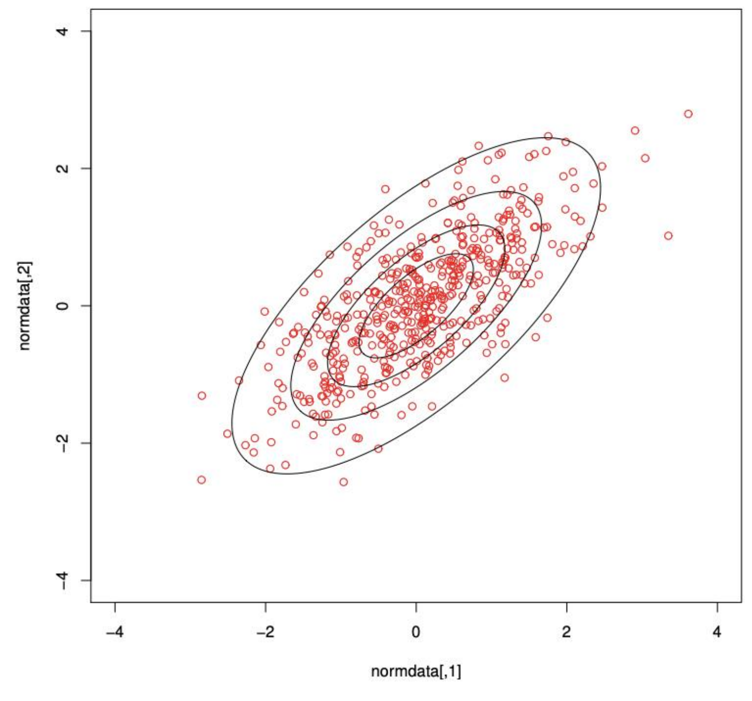
- Larger q gives more weight to larger differences in single variables
- Not scale equivariant: dominated by variables with larger variation → *standardization (z-score, Min-max etc)*
- Only L_2 (Euclidean) is rotation invariant.

Distances for Numerical Data

The **(squared) Mahalanobis distance** $d_M(\mathbf{x}_i, \mathbf{x}_l)^2 = (\boldsymbol{x}_i - \boldsymbol{x}_l)^T \mathbf{S}^{-1}(\mathbf{x}_i - \mathbf{x}_l)$

where $\boldsymbol{S}$ is a scatter matrix such as the sample covariance matrix.

Properties: Both scale and rotation invariant.



Other strategies to account for the correlations between numerical data:

- PCA to transform to uncorrelated variables before applying Euclidean distance (see, e.g. Markos et al. 2018).
- Variable selection to remove redundant variables

Distances for Ordinal Data

The dissimilarity measure must respect the meaningful order between categories while providing numerical values suitable for distance calculation.

Categories have a natural order (e.g., ”1 = Low", ”2 = Medium", ”3 = High") but the distance between adjacent categories is not inherently defined.

Transform and treat as numerical (Hastie, Tibshirani & Friedman, 2009)

Convert ordinal positions to evenly spaced values in [0,1] using: $\left|(i-\frac{1}{2})/M\right|$ where *i* is the position in ordering (1,2,3…), and M = total number of categories. This transformation places each ordinal category at the midpoint of its corresponding interval on a continuous [0,1] scale, representing the expected value for that category. They are then treated as numerical variables on this scale.

Example (5-point Likert scale):

"Strongly Disagree" (i=1) → 0.1 "Disagree" (i=2) → 0.3 "Neutral" (i=3) → 0.5
"Agree" (i=4) → 0.7 "Strongly Agree" (i=5) → 0.9

Distances for Categorical Data

For a categorical variable, it is not obvious how to quantify differences between different categories.

$d_{SM}(1,3) = {}^{2}/_{3}$

Listing ID	Property Type	Neighborhood	Pool
42781	Apartment	Plaka	No
37159	Apartment	Glyfada	Yes
65120	House	Monastiraki	No

$d_{SM}(1,2) = {}^{2}/_{3}$

Simple Matching Distance: $d_{SM}(\mathbf{x}_i, \mathbf{x}_l) = \frac{1}{p}\sum_{j=1}^{p} 1\big(x_{ij} \neq x_{lj}\big)$, where $1(\bullet)$ denotes the indicator function.

Counts the number of categorical variables on which two objects *i* and *l* do not coincide, divided by *p*.

- What if presence (e.g. of a Pool) is more important than absence?
- What if the number of categories should be taken into account (for instance, it is “easier” to differ when the variable has more categories)
- What if there are highly associated variables?

Distances for Categorical Data

For a categorical variable, it is not obvious how to quantify differences between different categories.

Listing ID	**Property Type**	**Neighborhood**	**Pool**
42781	Apartment	Plaka	No
37159	Apartment	Glyfada	Yes
65120	House	Monastiraki	No

Simple Matching Distance: $d_{SM}(\mathbf{x}_i, \mathbf{x}_l) = \frac{1}{p}\sum_{j=1}^{p} 1\left(x_{ij} \neq x_{lj}\right)$, where $1(\bullet)$ denotes the indicator function.

Counts the number of categorical variables on which two objects *i* and *l* do not coincide, divided by *p*.

Independent Measures

Lin, OF, IOF, Goodall: higher (or lower) weights to rare matches
Eskin: higher weights for larger number of categories

Association-Based Measures

Total Variation Distance: ${}^{1}/{}_{2}\, L_1$ norm between conditional probability distributions
Chi-square distance
Kullback-Leibler divergence (symmetric version)

Distances for Categorical Data

A general framework for distances between categorical variables (van de Velden et al., 2024)

Define category dissimilarity matrices $\mathbf{\Delta}_j$ for each variable j. The elements of this matrix, δ_{ab} quantify the dissimilarities between the categories a and b of the jth variable.

Example: If *Property Type* had just two categories {Apartment and House}, then for Simple Matching Distance:

$$\mathbf{\Delta}_{Property\_type} = \begin{array}{cc} & \begin{array}{cc} a & b \end{array} \\ \begin{array}{c} a \\ b \end{array} & \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{array}$$

The dissimilarities between the observations for the categorical variable j are $\boldsymbol{D}_j = \mathbf{Z}_j\mathbf{\Delta}_j\mathbf{Z}_j'$, where matrix $\mathbf{Z}_j$ is the indicator matrix corresponding to the jth categorical variable.

The dissimilarity matrix can be calculated as $\mathbf{D} = \mathbf{Z}\mathbf{\Delta}\mathbf{Z}' = \sum_{j=1}^{p} \mathbf{Z}_j\mathbf{\Delta}_j\mathbf{Z}_j' = \sum_{j=1}^{p} \mathbf{D}_j$

van De Velden, M., Iodice D'Enza, A., Markos, A., & Cavicchia, C. (2024). A general framework for implementing distances for categorical variables. *Pattern Recognition*, *153*, 110547.

Distances for Categorical Data

A general framework for distances between categorical variables (van de Velden et al., 2024)

Distance	Category disimilarity matrix $\mathbf{\Delta}_j$ (or its typical element δ_{ab}, for $a \neq b$)
Matching	$\mathbf{\Delta}_{m_j} = \mathbf{1}\mathbf{1}^\top - \mathbf{I}$
Eskin	$\mathbf{\Delta}_{e_j} = 2/q_j^2 \mathbf{\Delta}_{m_j}$
Occurence frequency (OF)	$\mathbf{\Delta}_{OF_j} = \log(\mathbf{p}_j)\log(\mathbf{p}_j)^\top \odot \mathbf{\Delta}_{m_j}$
Inverse OF	$\mathbf{\Delta}_{IOF_j} = \log(n\mathbf{p}_j)\log(n\mathbf{p}_j)^\top \odot \mathbf{\Delta}_{m_j}$
Indicator: No scaling	$\mathbf{\Delta}_{d_j} = 2\mathbf{\Delta}_{m_j}$
Indicator: Hennig-Liao scaling	$\mathbf{\Delta}_{HL_j} = 2\eta_j \mathbf{\Delta}_{m_j}$
Indicator: Standard deviation scaling	$\delta^s_{ab_j} = \sqrt{\frac{1}{q_j}}\left(s_a^{-1/2} + s_b^{-1/2}\right)$
Indicator: Cat. dissimilarity scaling	$\mathbf{\Delta}_{cds_j} = \frac{1}{q_j}\mathbf{S}_{j_d}^{-1/2}\mathbf{\Delta}_{m_j}\mathbf{S}_{j_d}^{-1/2}$

Distances for Mixed-Type Data

Recall the general multivariate mixed-variable dissimilarity:

$$d(\mathbf{x}_i, \mathbf{x}_l) = \sum_{j=1}^{p_n} d_{j_n}(x_{ij}, x_{lj}) + \sum_{j=1}^{p_c} d_{j_c}(x_{ij}, x_{lj})$$

Gower's Dissimilarity (1971):

$$d(\mathbf{x}_i, \mathbf{x}_l) = \underbrace{\sum(\text{Numerical Distances})}_{\text{Range Normalized Manhattan}} + \underbrace{\sum(\text{Categorical Distances})}_{\text{Simple Matching}}$$

Heterogeneous Euclidean Overlap Metric (Wilson & Martinez, 1997):

$$d(\mathbf{x}_i, \mathbf{x}_l) = \underbrace{\sum(\text{Numerical Distances})}_{\text{Normalized Euclidean}} + \underbrace{\sum(\text{Categorical Distances})}_{\text{Overlap}}$$

GUDMM (Mousavi & Sehhati, 2023):

$$d(\mathbf{x}_i, \mathbf{x}_l) = \underbrace{\sum(\text{Numerical Distances})}_{\substack{\text{Modified Mahalanobis with} \\ \text{Mutual Information-based relevance}}} + \underbrace{\sum(\text{Categorical Distances})}_{\substack{\text{Entropy-based distances} \\ \text{(Normalized joint entropy for nominal,} \\ \text{Jensen-Shannon for numerical-categorical)}}}$$

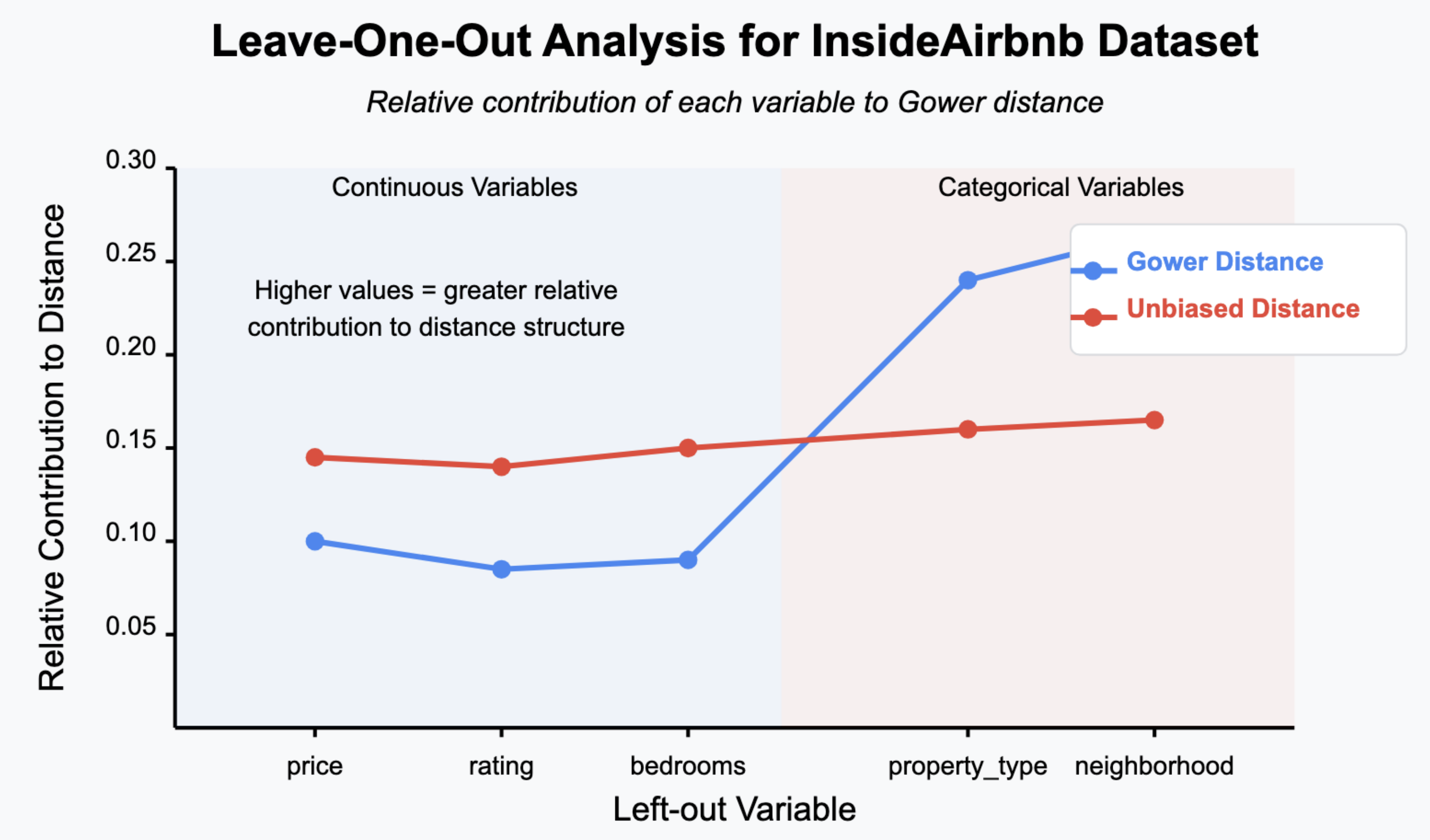
(Un)Biased Distances

The **influence** of the jth variable **on object dissimilarity** *depends upon its relative contribution to the average object dissimilarity measure* over all pairs of objects in the data set.

Numerical: $w_{j_n} = 1/\bar{d}_j$

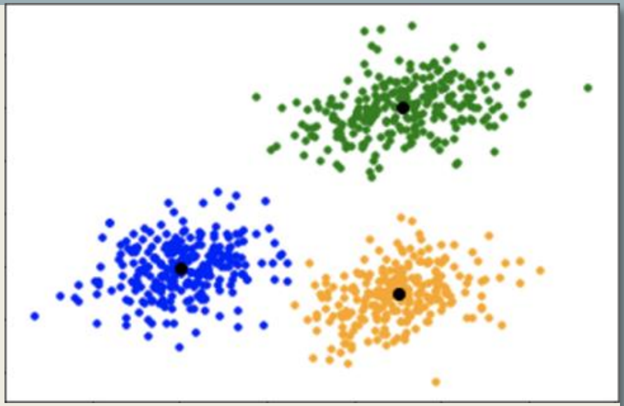
Categorical: $w_{j_c} = 1/\left(\mathbf{p}_j^{\mathrm{T}} \boldsymbol{\Delta}_j \mathbf{p}_j\right)$

$\mathbf{p}_j$: category probabilities of j



van De Velden, M., Iodice D'Enza, A., Markos, A., & Cavicchia, C. (2025). Unbiased mixed variables distance. *arXiv preprint arXiv:2411.00429*.

Distance-Based Algorithms: K-means Clustering



Algorithm

1. Initialize K cluster centers $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_k$ randomly
2. Repeat until convergence:

   a. Assignment step: Assign each observation $\boldsymbol{x}_i$ to the closest cluster center,
   $argmin\{l \in 1 \ldots K\} d(\mathbf{x}_i, \boldsymbol{\mu}_l)$, where $d(\mathbf{x}_i, \boldsymbol{\mu}_l) = \sum_{j=1}^{p_n} \left(x_{ij}, \mu_{lj}\right)^2$

   b. Update step: Recalculate cluster centers as the mean of all objects assigned to that cluster,
   $\boldsymbol{\mu}_l = (1/|S_l|) \sum \langle \mathbf{x}_i \in S_l \rangle \mathbf{x}_i$, where S_l is the set of observations in cluster l.

Model-Based Equivalent (Gaussian Mixture Model)

- K-means cluster centers are Maximum Likelihood estimators for mean vectors in a mixture of *K* Gaussian distributions, where all distributions have identical spherical covariance matrices $(\boldsymbol{\Sigma} = \mathrm{b}\mathbf{I})$

Extensions of K-means to Mixed Data

K-Prototypes (Huang, 1997)

$$d(\mathbf{x}_i, \boldsymbol{q}_l) = \underbrace{\sum(\text{Numerical Distances})}_{\text{Euclidean}} + \gamma \underbrace{\sum(\text{Categorical Distances})}_{\text{Simple Matching}}$$

Modha-Spangler K-means (2003)

$$d(\mathbf{x}_i, \boldsymbol{q}_l) = \underbrace{\sum(\text{Numerical Distances})}_{\text{Sq. Euclidean}} + \underbrace{\sum(\text{Categorical Distances})}_{\text{Cosine Dissimilarity on 0-1}}$$

Ahmad & Dey (2007)

$$d(\mathbf{x}_i, \boldsymbol{q}_l) = \underbrace{\sum(\text{Numerical Distances})}_{\text{Weighted Euclidean}} + \underbrace{\sum(\text{Categorical Distances})}_{\text{Total Variation Distance}}$$

Distance-Based Algorithms: PAM

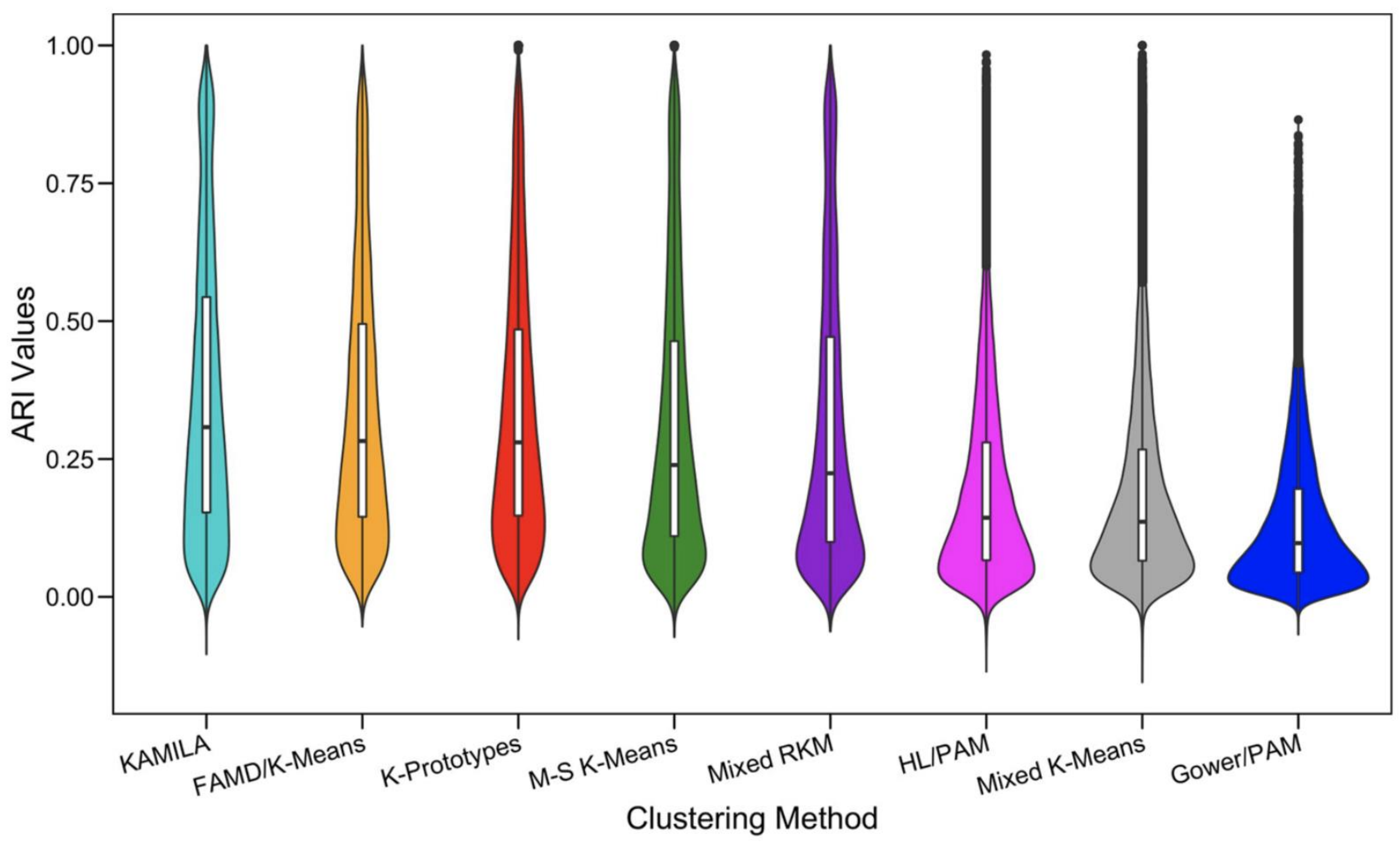
Partitioning Around Medoids (Kaufman & Rousseeuw, 1987)
Uses actual data points (medoids) as cluster representatives

1. Initialize: Select *k* objects as initial medoids
2. Repeat until convergence:
   a. Assignment step: Assign each object $\boldsymbol{x}_i$ to the nearest medoid based on $d(\mathbf{x}_i, \mathbf{x}_l)$,

   b. Update step: For each cluster evaluate if replacing the medoid with another object reduces the sum of dissimilarities

Advantages
- Works directly with dissimilarity matrices, $\mathbf{D}$
- Compatible with *any* dissimilarity measure
- Robustness with outliers

Which partitioning method is "best"?



Costa, E., Papatsouma, I., & Markos, A. (2023). Benchmarking distance-based partitioning methods for mixed-type data. *Advances in Data Analysis and Classification*, *17*(3), 701-724.

Hierarchical Clustering

Initialisation. $k = 1$. Every object is a cluster on its own.

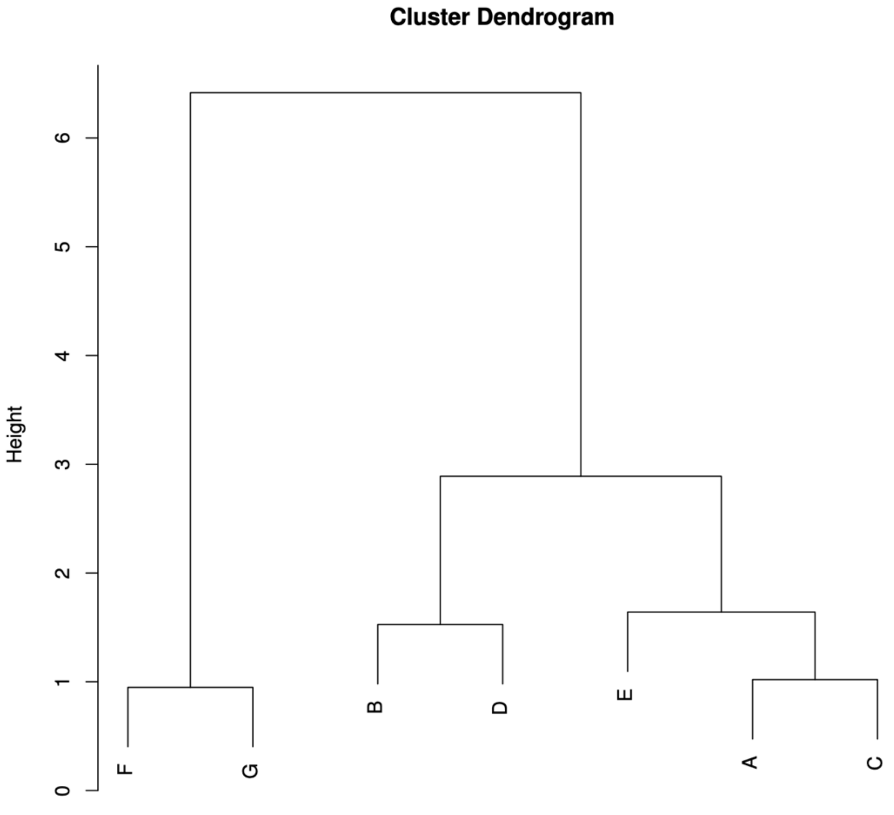
$$\mathcal{C}_1 = \{C_1, \ldots, C_n\} = \{\{\mathbf{x}_1\}, \ldots, \{\mathbf{x}_n\}\}\,,\ K_1 = n.$$

Step k.1. Find $C_i, C_j \in \mathcal{C}_k$ so that $D(C_i, C_j) = \min_{(C_l, C_m)} D(C_l, C_m)$.

Step k.2. Merge C_i, C_j:

$$\mathcal{C}_{k+1} = \mathcal{C}_k \cup \{C_i \cup C_j\} \setminus \{C_i, C_j\}$$

so that $K_{k+1} = K_k - 1$. Set $H_k = D(C_i, C_j)$, the level (dendrogram height) at which C_i and C_j are merged[1].



Distance-Based Algorithms: K-Nearest Neighbors

KNN Algorithm

1. Calculate dissimilarities $d(\mathbf{x}_i, \mathbf{x}_l)$ between all observations
2. For each query point $\mathbf{x}_p$:
   a. Identify the K observations with smallest dissimilarity to $\mathbf{x}_p$
   b. *For classification*: assign the majority class among neighbors
   c. *For regression:* compute weighted average of K neighbors' values

KNN with Mixed-Type data

Define appropriate dissimilarity measure for mixed-type data

Discussion/Research Directions

Unification

Fragmented literature with ad hoc implementations (the "reinvention issue").

Need for unified frameworks. **No** need for **more methods**.

Uncertainty Quantification

Distance-based methods lack probabilistic uncertainty estimates (e.g., confidence intervals for cluster assignments).

Potential solution: Integrate bootstrapping or Bayesian frameworks.

Theoretical Foundations: Establishing stronger theoretical connections between distance-based and model-based approaches for mixed data.

Key References

Costa, E., Papatsouma, I., & Markos, A. (2023). Benchmarking distance-based partitioning methods for mixed-type data. *Advances in Data Analysis and Classification*, *17*(3), 701-724.

Deza, M. M., & Deza, E. (2016). *Encyclopedia of Distances* (4th ed.). Springer. https://doi.org/10.1007/978-3-662-52844-0

Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Tibshirani, R., & Friedman, J. (2009). Unsupervised learning. *The Elements of Statistical Learning: Data mining, Inference, and Prediction*, 485-585.

Markos, A., D'Enza, A. I., & van de Velden, M. (2019). Beyond tandem analysis: Joint dimension reduction and clustering in R. *Journal of Statistical Software*, *91*, 1-24.

van de Velden, M., Iodice D' Enza, A., & Markos, A. (2019). Distance-based clustering of mixed data. *Wiley Interdisciplinary Reviews: Computational Statistics*, *11*(3), e1456.

van De Velden, M., Iodice D'Enza, A., Markos, A., & Cavicchia, C. (2024). A general framework for implementing distances for categorical variables. *Pattern Recognition*, *153*, 110547.

van De Velden, M., Iodice D'Enza, A., Markos, A., & Cavicchia, C. (2024). *Unbiased mixed variables distance*. arXiv preprint arXiv:2411.00429.

Contact me at:
Angelos Markos
email: amarkos@eled.duth.gr
website: amarkos.gr
Blog: Celebrating Uncertainty

SCAN ME

https://github.com/amarkos/opaworkshop