

Statistical Jump Model for Mixed-Type Data with Missing Data Imputation - Supplementary Material

Additional simulation results

This supplementary material includes additional results on the accuracy and convergence of JM-mix, providing extended evaluations to complement the findings presented in the main text.

Accuracy

To evaluate clustering performance, one can consider several metrics commonly used in the literature. The Adjusted Rand Index (ARI, [Hubert & Arabie, 1985](#)) is widely recognized as a robust measure for assessing clustering quality, as it accounts for both cluster agreement and chance alignment. ARI directly compares the estimated clustering with the ground truth and provides a reliable performance assessment, particularly in simulated environments where the true labels are known. Another commonly used metric is the Silhouette Index, which measures the cohesion and separation of clusters. However, it does not take into account the ground truth, making it less suitable for evaluating accuracy when true labels are available. The Normalized Mutual Information (NMI) is defined as a normalized measure of the mutual dependence between clustering outcomes and true labels. While NMI is effective for comparing cluster assignments, it is computationally intensive, particularly for large-scale datasets¹.

Cluster Purity (CP) is a measure used to evaluate the quality of clustering solutions when ground truth labels are available. It is based on the fraction of correctly classified data points in each cluster by comparing the clustering results to the ground truth.

The formula for its calculation is the following:

$$CP = \frac{1}{N} \sum_{k=1}^K \max_c |C_k \cap T_c|$$

where:

- N is the total number of data points.
- K is the number of clusters.

¹For a detailed discussion on these and other cluster quality indices, please refer to [Aghabozorgi, Shirkhorshidi, and Wah \(2015\)](#).

- C_k is the set of data points assigned to cluster k .
- T_c is the set of data points belonging to the true class c .
- $|C_k \cap T_c|$ is the number of data points in the intersection of cluster k and true class c .
- $\max_c |C_k \cap T_c|$ identifies the size of the majority class in each cluster k .

The purity score ranges from 0 to 1: a value of 1 indicates that all clusters are perfectly pure (i.e., contain points from only one class), whereas a lower value indicates a higher degree of impurity within the clusters. Unlike ARI, CP does not penalize over-segmentation, where true clusters are unnecessarily divided into smaller ones. While these fragmented clusters may appear pure—containing a majority of points from the same true class—this does not reflect the true data structure. CP evaluates only the majority class within each cluster, making it less discriminative for assessing clustering performance.

Tables 1 and 2 summarize the results of our clustering analysis. Specifically, Table 1 presents the outcomes across the three simulation setups described in Section 3 of the paper, while Table 2 focuses on setup 1 under different missing data mechanisms (MCAR, MAR, MNAR). The results highlight that, for a lower number of features ($P = 4$), the JM-mix consistently outperforms the alternatives in terms of cluster purity, showing its robustness in scenarios with few information. However, cluster purity appears less discriminative than ARI.

In conclusion, while ARI remains our preferred metric for assessing cluster quality due to its robustness, CP provides a complementary perspective and further supports the strong performance of the JM-MIX model.

Table 1 Cluster Purity (CP) for k -prototypes (k -prot), spectral clustering (spClust), and jump model for mixed data (JM-mix) across three setups.

Setup 1				
T	Method	$P = 4$	$P = 20$	$P = 50$
50	k -prot	0.91	0.99	1.00
	spClust	0.91	0.99	1.00
	JM-mix	0.91	0.99	1.00
100	k -prot	0.89	0.99	0.99
	spClust	0.89	0.99	0.99
	JM-mix	0.89	0.99	0.99
500	k -prot	0.89	1.00	1.00
	spClust	0.89	1.00	1.00
	JM-mix	0.89	0.99	0.99
Setup 2				
T	Method	$P = 4$	$P = 20$	$P = 50$
50	k -prot	0.91	0.98	0.99
	spClust	0.91	0.98	0.99
	JM-mix	0.95	0.98	0.99
100	k -prot	0.89	0.98	0.99
	spClust	0.89	0.98	0.99
	JM-mix	0.95	0.98	0.99
500	k -prot	0.90	0.98	0.99
	spClust	0.90	0.98	0.99
	JM-mix	0.98	0.99	0.99
Setup 3				
T	Method	$P = 4$	$P = 20$	$P = 50$
50	k -prot	0.85	0.99	1.00
	spClust	0.84	0.99	1.00
	JM-mix	0.94	0.99	1.00
100	k -prot	0.86	0.99	1.00
	spClust	0.85	0.99	1.00
	JM-mix	0.95	0.99	0.99
500	k -prot	0.85	0.99	1.00
	spClust	0.84	0.99	1.00
	JM-mix	0.98	0.99	0.99

Table 2 Cluster Purity (CP) for k -prototypes (k -prot) and jump model for mixed data (JM-mix) under MCAR, MAR, and MNAR mechanisms with 10% and 20% missing data (setup 1).

	T	Method	$P = 4$	$P = 20$	$P = 50$
10% Missing Data					
MCAR	50	k -prot	0.90	0.99	0.99
		JM-mix	0.94	0.99	0.99
	100	k -prot	0.86	0.99	1.00
		JM-mix	0.94	0.99	1.00
	500	k -prot	0.86	0.99	1.00
		JM-mix	0.98	0.99	0.99
MAR	50	k -prot	0.88	0.99	0.99
		JM-mix	0.93	0.99	1.00
	100	k -prot	0.85	0.98	1.00
		JM-mix	0.95	0.99	0.99
	500	k -prot	0.86	0.99	1.00
		JM-mix	0.97	0.99	0.99
MNAR	50	k -prot	0.82	0.89	0.91
		JM-mix	0.85	0.91	0.91
	100	k -prot	0.77	0.85	0.87
		JM-mix	0.83	0.85	0.87
	500	k -prot	0.71	0.81	0.81
		JM-mix	0.80	0.82	0.82
20% Missing Data					
MCAR	50	k -prot	0.88	0.98	0.99
		JM-mix	0.94	0.98	0.99
	100	k -prot	0.83	0.97	1.00
		JM-mix	0.92	0.98	0.99
	500	k -prot	0.83	0.97	0.99
		JM-mix	0.97	0.98	0.99
MAR	50	k -prot	0.88	0.98	0.99
		JM-mix	0.91	0.98	0.99
	100	k -prot	0.83	0.97	1.00
		JM-mix	0.92	0.98	0.99
	500	k -prot	0.81	0.97	1.00
		JM-mix	0.96	0.98	0.99
MNAR	50	k -prot	0.82	0.89	0.91
		JM-mix	0.84	0.91	0.91
	100	k -prot	0.77	0.85	0.87
		JM-mix	0.83	0.85	0.87
	500	k -prot	0.71	0.81	0.81
		JM-mix	0.80	0.82	0.82

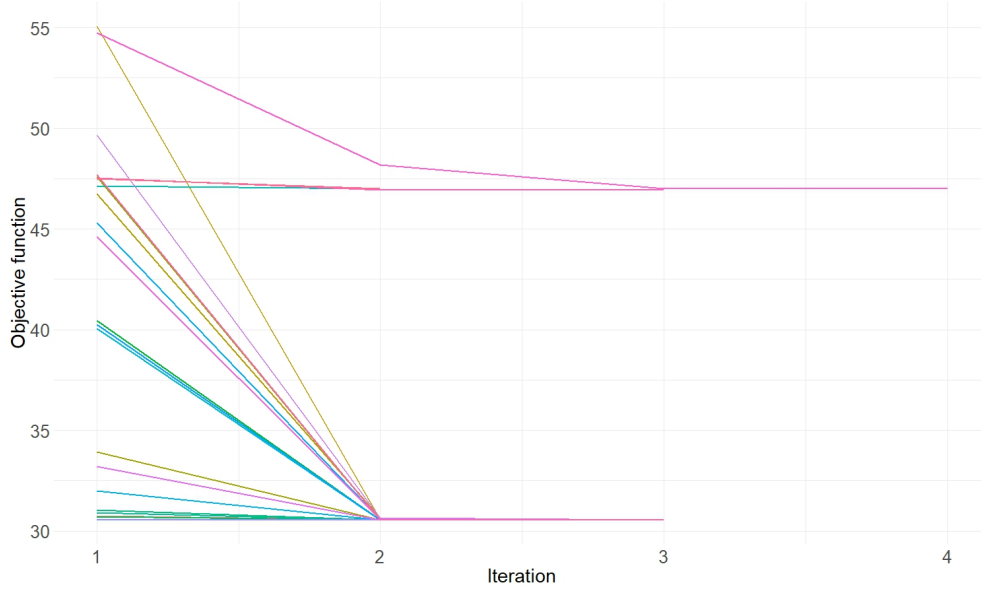


Fig. 1 Local convergence of the algorithm: the graph depicts the objective function, as defined in Szepannek et al. (2024), across iterations for different initializations.

Convergence

Figure 1 presents additional results on convergence, evaluated using the objective function proposed in Szepannek, Aschenbruck, and Wilhelm (2024). To assess local convergence, we adopted a similar methodology, estimating JM-mix under setup 1 with parameters $T = 200$, $P = 20$, using 30 different initializations and allowing up to 50 iterations per run. Even with the objective function from Szepannek et al. (2024), the results demonstrate rapid convergence.

To validate the capability of the algorithm to handle missing data, we apply the procedure in Section 3.4 of the paper across various missing data mechanisms (MAR, MCAR, and MNAR), testing convergence of the objective function with 20% missingness. The findings in Figure 2 confirm that the algorithm effectively reduces the objective function, ensuring convergence during estimation.

Comparison of norm choices

Nystrup, Lindström, and Madsen (2020) employ the l_2 norm to fit a statistical jump model. Since our data includes mixed types, we instead use the Gower distance, which essentially relies on an l_1 norm for continuous variables. This choice might influence clustering performance.

In Bemporad, Breschi, Piga, and Boyd (2018), a general metric $\ell(\cdot, \cdot)$ is proposed, with l_2 as a specific case, making our approach another particular instance. That said, the l_1 norm is more robust to outliers compared to l_2 , as it is less sensitive to extreme

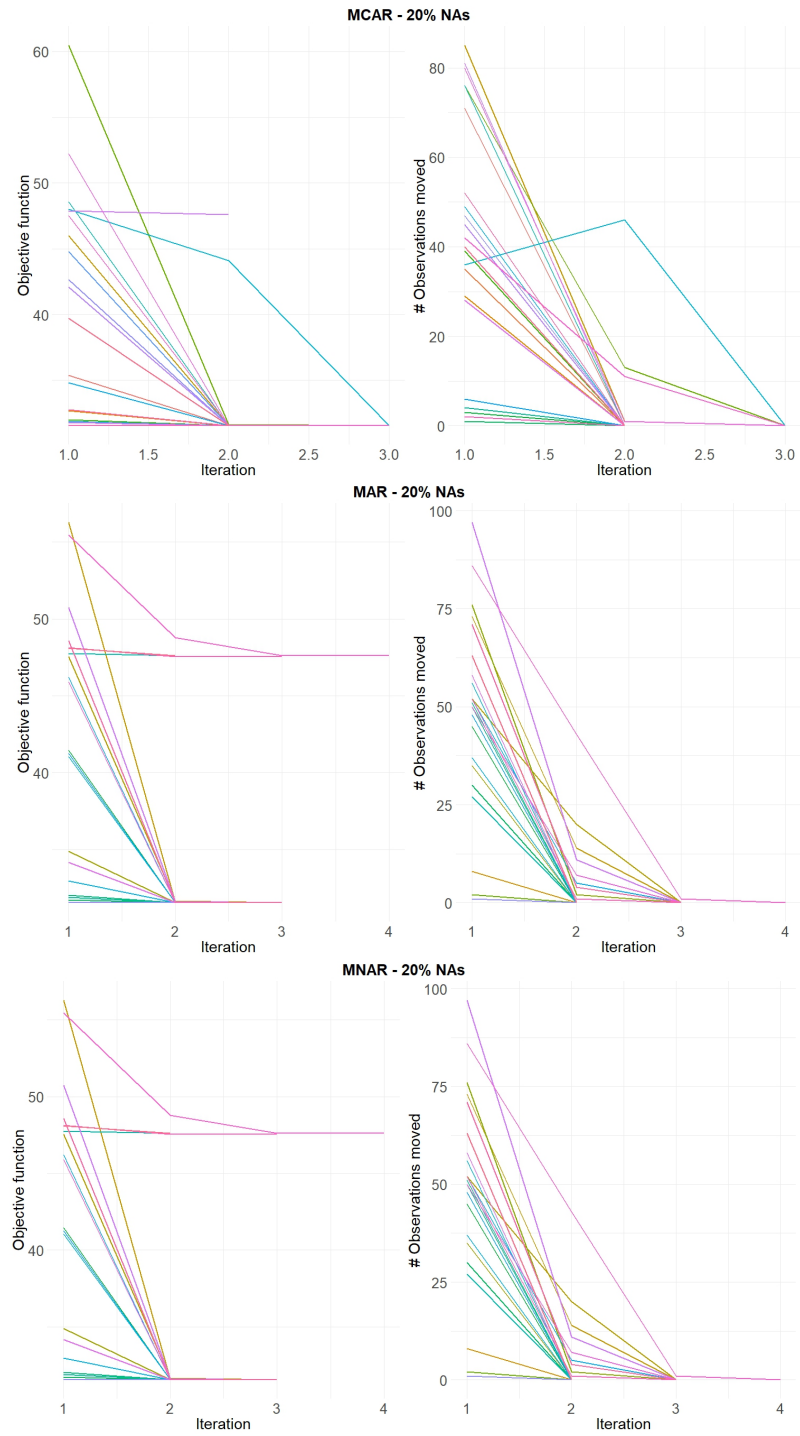


Fig. 2 The left panels show the convergence of the objective function over iterations, while the right panels depict the number of observations moved after each iteration. Each row refers to a different missing mechanism.

values. Moreover, using the JM-mix with median-based prototypes is expected to produce more compact clusters.

To explore these effects, we perform an additional simulation study. Features are generated from a multivariate Student- t distributed hidden Markov model with 3 latent states:

$$\mathbf{y}_{t,\cdot} \mid s_t \sim \mathcal{S}_P(\tilde{\boldsymbol{\mu}}_{s_t}, \tilde{\boldsymbol{\Sigma}}_P, \nu),$$

where the non-centrality parameter vectors are $(1, \dots, 1)'$, $(0, \dots, 0)'$, $(-1, \dots, -1)'$, and $\nu = 4$ to account for fat tails. The state-conditional scale matrix $\tilde{\boldsymbol{\Sigma}}_P$ is derived as $\tilde{\boldsymbol{\Sigma}}_P = \frac{\nu-2}{\nu} \boldsymbol{\Sigma}_P$, where $\boldsymbol{\Sigma}_P$ is the covariance matrix with diagonal elements 1 and off-diagonal elements 0.

We vary $T \in \{50, 100, 500\}$ and $P \in \{4, 20, 50\}$, generating 100 datasets for each combination of T and P . For each dataset, we fit a JM-mix model using the Gower distance (here denoted as l_1) and compare it to the statistical jump model proposed in Nystrup et al. (2020), which employs the Euclidean distance (denoted as l_2). We recall that, unlike JM-mix, which relies on state-conditional medians, the original jump model is based on state-conditional means. The parameter λ is varied in steps of 0.05 within $[0, 1]$, and the optimal λ is selected to maximize the ARI between true and estimated state sequences.

Table 3 presents the ARI for the two jump models when l_1 and l_2 metrics are used. The results indicate that l_1 consistently outperforms l_2 , particularly in scenarios with limited data, demonstrating higher accuracy and more reliable outputs.

Table 3 Adjusted Rand Index (ARI) computed between true and estimated latent sequences for the two metrics, l_1 and l_2 , for the three setups. Monte Carlo standard deviations are in parentheses.

T	Metric	$P = 4$	$P = 20$	$P = 50$
50	l_1	0.77 (0.24)	0.87 (0.20)	0.88 (0.21)
	l_2	0.50 (0.27)	0.68 (0.29)	0.75 (0.31)
100	l_1	0.88 (0.11)	0.96 (0.12)	0.98 (0.04)
	l_2	0.64 (0.17)	0.89 (0.09)	0.95 (0.08)
500	l_1	0.92 (0.03)	0.98 (0.01)	0.99 (0.02)
	l_2	0.72 (0.07)	0.94 (0.02)	0.98 (0.04)

References

- Aghabozorgi, S., Shirkhorshidi, A.S., Wah, T.Y. (2015). Time-series clustering—A decade review. *Information Systems*, 53, 16–38,
- Bemporad, A., Breschi, V., Piga, D., Boyd, S.P. (2018). Fitting jump models. *Automatica*, 96, 11–21,

- Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2, 193–218,
- Nystrup, P., Lindström, E., Madsen, H. (2020). Learning hidden Markov models with persistent states by penalizing jumps. *Expert Systems with Applications*, 150, 113307,
- Szepannek, G., Aschenbruck, R., Wilhelm, A. (2024). Clustering large mixed-type data with ordinal variables. *Advances in Data Analysis and Classification*, 1–19,