**Supplementary Materials**

*The Integrated Latent Space Model (ILSM)*

An important limitation of ISM and of other multi-view latent space approaches is the required availability of multi-view data for all observations in the training set. For financial and/or logistical reasons, a particular view may be missing in a subset of the observations, and this subset is in turn dependent on the view under consideration.

Workflow S1 describes a variant of ISM that can process multi-view data with missing views. This approach is called the Integrated Latent Space Model (ILSM) because the ISM itself is applied to a collection of ISM-transformed datasets, each of them coming from a subset of views whose intersection contains a suitable number of observations to be processed by ISM. Within each subset, an additional expansion process allows the integration of all observations inside and outside each view, resulting in much larger transformed views than the original intersection would allow, as shown in Figure S2.

---

**Workflow S1** Integrated Latent Space Model (ILSM)

---

**Input**: $m$ views $\{\boldsymbol{X}^1, \cdots, \boldsymbol{X}^m\}, \boldsymbol{X}^v \in \mathbb{R}_+^{n \times d_v}$ where $n$ is the number of rows common to all views and $d_v$ is the number of columns in the $v^{th}$ view (it is assumed for each column that its values lie between 0 and 1 after normalization by the maximum row value).

**Output**: NTF factors $\boldsymbol{W}^*, \boldsymbol{H}^*, \boldsymbol{Q}^*, \boldsymbol{W}^* \in \mathbb{R}_+^{n_+ \times d_l}, \boldsymbol{H}^* \in \mathbb{R}_+^{d_l \times d_l}, \boldsymbol{Q}^* \in \mathbb{R}_+^{m \times d_l}$ and updated view-mapping matrix $\boldsymbol{H}$,where $n_+$ is the number of rows in the union of all observations in all views and $d_l$ is the dimension of the latent space.

1: Partition: Create subsets of views, each with an intersection of views that contains a suitable number of observations to be processed by ISM;

2: Local integration: Apply ISM on each subset of views;

3: Projection**:** For each view in a given subset, project the non-missing observations outside the intersection onto the latent space by using the ISM workflow 2;

(For other views for which the corresponding observations are missing, ISM view-scores remain missing)

4: Expansion: Estimate missing view-scores by the weighted average of existing view-scores, where the weights are the ISM view loadings;

5: Unified Integration: Apply ISM to the expanded transformed data from all subsets of views;

---

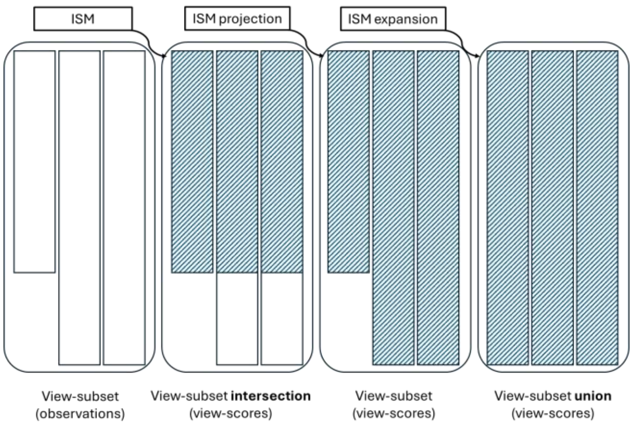


**Figure S1.** Illustration of the ILSM expansion process.

*Application of ILSM to the UCI Digits Data*

UCI Digits Data: The data can be found at Datasets - [Datasets - UCI Machine Learning Repository] and contains 6 heterogenous views: 76 Fourier coefficients of the character shapes, 216 profile correlations, 64 Karhunen-Love coefficients, 240 pixel averages of the images from 2x3 windows, 47 Zernike moments and 6 morphological features, where each class contains 200 labeled examples.

In the first 2 views, the last 500 examples were set to missing.

Two sets of views were considered, consisting of the first three and last three views, respectively. Two separate ISM analyses were performed for each view-set. In the first analysis, the first 1500 examples were included, while in the second analysis the last 1500 examples were included. Thus, only 1000 out of the 2000 examples were analyzed with all available views.

After applying the ILSM expansion process, the two ISM-transformed data containing the 2000 examples were integrated using ISM to obtain the meta-scores. Following the article's analysis workflow, the 10 classes were identified (Figure S2) with a purity index of 5.31, which is slightly lower than the purity index of 5.81 obtained in the original data analysis.

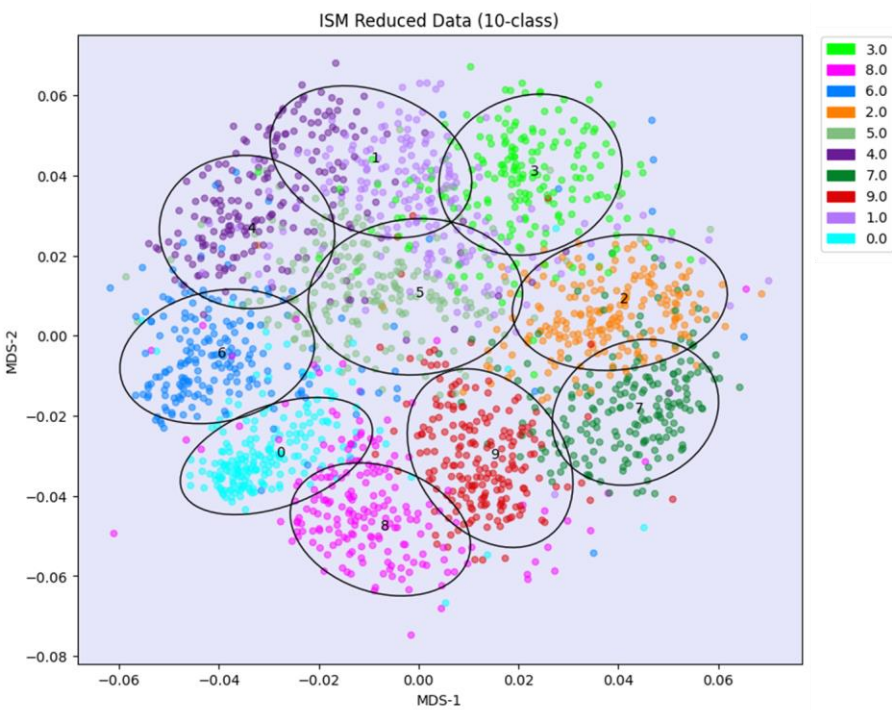


**Figure S2.** Analysis of the dataset of UCI digits using ILSM after masking a large number of views.

*Robustness results for MVMDS, PCA, GFA and MOFA+ (UCI-digits data)*

**Table S1.** Number of found clusters, purity index and adjusted Rand index for 4 latent-space methods, as a function of the rank used.

| Method(rank) | Number of clusters | Purity | Adjusted Rand index |
|---|---|---|---|
| MVMDS(9) | 8 | 4.82 | 0.9155 |
| **MVMDS(10)** | **7** | **4.06** | **0.9077** |
| MVMDS(11) | 9 | 4.56 | 0.9039 |
| PCA(9) | 4 | 1.93 | 0.8816 |
| **PCA(10)** | **4** | **1.93** | **0.8816** |
| PCA(11) | 4 | 1.91 | 0.8811 |
| GFA(8) | 9 | 4.45 | 0.9001 |
| **GFA(9)** | **9** | **4.39** | **0.8998** |
| GFA(10) | 8 | 3.92 | 0.8935 |
| MOFA+(9) | 4 | 1.31 | 0.8378 |
| **MOFA+(10)** | **7** | **2.91** | **0.8670** |
| MOFA+(11) | 4 | 1.31 | 0.8553 |

*Robustness results for MVMDS, PCA, GFA and MOFA+ (Signature 915 data)*

**Table S2.** Number of found clusters, purity index and adjusted Rand index for 4 latent-space methods, as a function of the rank used.

| Method | Number of clusters | Purity | Adjusted Rand index |
|---|---|---|---|
| MVMDS(9) | 12 | 10.75 | 0.9923 |
| **MVMDS(10)** | **12** | **11.19** | **0.9927** |
| MVMDS(11) | 11 | 10.28 | 0.9951 |
| PCA(9) | 10 | 6.54 | 0.9835 |
| **PCA(10)** | **9** | **6.89** | **0.9874** |
| PCA(11) | 10 | 7.67 | 0.9902 |
| GFA(11) | 13 | 11.10 | 0.9935 |
| **GFA(12)** | **13** | **11.84** | **0.9953** |
| GFA(13) | 12 | 11.02 | 0.9962 |
| MOFA+(12) | 11 | 9.81 | 0.9874 |
| **MOFA+(13)** | **13** | **12.13** | **0.9860** |
| MOFA+(14) | 11 | 8.27 | 0.9532 |

*Distributed NMF, workflow*

**Workflow S2 Distributed NMF using ISM**

**Input**: matrix $\boldsymbol{X} \in \mathbb{R}_{+}^{n \times p}$.

**Output**: factoring matrices $\boldsymbol{W} \in \mathbb{R}_{+}^{n \times k}, \boldsymbol{H} \in \mathbb{R}_{+}^{p \times k}$.

1: Consider a random partition of $\boldsymbol{X}$ into $m$ matrices $\boldsymbol{X}_v \in \mathbb{R}_{+}^{n \times p_v}$(matrices may be of different size, e.g. if $p/m$ not an integer).

2: Factorize each view $\boldsymbol{X}_v$ using **NMF** and same rank $k$:

$$\boldsymbol{X}_v = \boldsymbol{W}_v \boldsymbol{H}_v{}^{\boldsymbol{T}} + \boldsymbol{E}_v, \boldsymbol{W}_v \in \mathbb{R}_{+}^{n \times k}, \boldsymbol{H}_v \in \mathbb{R}_{+}^{p_v \times k}, \boldsymbol{E}_v \in \mathbb{R}^{n \times p_v}.$$

3: Apply **ISM** on the list of views $\{\boldsymbol{W}_v\}_{1 \le v \le m}$:

$\boldsymbol{W}_v = \boldsymbol{W}^* \boldsymbol{H}_v^{*\boldsymbol{T}} + \boldsymbol{E}_v^*, \boldsymbol{W}^* \in \mathbb{R}_{+}^{n \times k}, \boldsymbol{H}_v^* \in \mathbb{R}_{+}^{k \times k}, \boldsymbol{E}_v^* \in \mathbb{R}^{n \times k}$ where $\boldsymbol{W}^*$ contains ISM meta- scores and $\{\boldsymbol{H}_v^*\}_{1 \le v \le m}$ contains the view-mapping matrices to the $\{\boldsymbol{W}_v\}_{1 \le v \le m}$.

4: Factorize each view $\boldsymbol{X}_v$ using $\boldsymbol{H}_v$ from step 2 and $\boldsymbol{W}^* \boldsymbol{H}_v^{*\boldsymbol{T}}$ from step 3:

$$\boldsymbol{X}_v = \boldsymbol{W}^* \boldsymbol{H}_v^{*\boldsymbol{T}} \boldsymbol{H}_v{}^{\boldsymbol{T}} + \boldsymbol{E}_v', \boldsymbol{W}^* \in \mathbb{R}_{+}^{n \times k}, \boldsymbol{H}_v^* \in \mathbb{R}_{+}^{k \times k}, \boldsymbol{H}_v \in \mathbb{R}_{+}^{p_v \times k}, \boldsymbol{E}_v' \in \mathbb{R}^{n \times p_v}.$$

5: $\boldsymbol{X}$ can now be factorized:

$$\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H}^{\boldsymbol{T}} + \boldsymbol{E}, \boldsymbol{W} = \boldsymbol{W}^* \in \mathbb{R}_{+}^{n \times k}, \boldsymbol{H} = \{\boldsymbol{H}_v \boldsymbol{H}_v^*\}_{1 \le v \le m} \in \mathbb{R}_{+}^{p \times k}, \boldsymbol{E} = \{\boldsymbol{E}_v'\}_{1 \le v \le m} \in \mathbb{R}^{n \times p}.$$

*Distributed NMF, example*

A dense matrix of size (76 x 10000) was analyzed using either standard NMF or distributed NMF with 10 slices of size (76 x 1000). When 4 components were used, the relative errors were very similar (0.40 for NMF versus 0.41 for Distributed NMF). However, the computational time required by Distributed NMF was reduced by 13% assuming no parallelism and by 92% assuming perfect parallelism.