Supplementary for "Discovering Non-Redundant K-means Clusterings in Optimal Subspaces"

Dominik Mautz[†], Wei Ye[†], Claudia Plant[§], Christian Böhm[†]
[†]Ludwig-Maximilians-Universität München, Munich, Germany
{mautz,ye,boehm}@dbs.ifi.lmu.de
[§]University of Vienna, Vienna, Austria
claudia.plant@univie.ac.at

S.1 SYMBOLS AND DEFINITIONS

We used the following symbols and definitions in our paper:

Symbol	Interpretation
$d \in \mathbb{N}$	Dimensionality of original space
$S \in \mathbb{N}$	Number of subspaces
$k_j \in \mathbb{N}$	Number of Clusters in the j'th subspace
$m_j \in \mathbb{N}$	Dimensionality of the j'th subspace
$\mathcal{D} \subseteq \mathbb{R}^d$	Set of all objects
$C_{j,i}$	Objects of cluster i in subspace j
$\mathbf{x} \in \mathcal{D}$	A data point or object of the dataset
$\boldsymbol{\mu}_{j,i} \in \mathbb{R}^m$	Original space mean of cluster i in subspace j
$P_j \in \mathbb{R}^{d \times m_j}$	Projection onto the j'th subspace
$V \in \mathbb{R}^{d \times d}$	Orthogonal matrix of a rigid transformation
$\Sigma_j \in \mathbb{R}^{d \times d}$	Sum of scatter matrices of clustering j—Eq. 4
$\mathbf{I}_l$	$l \times l$ identity matrix
$\mathbf{0}_{l,r}$	$l \times r$ zero matrix

S.2 FURTHER PROPERTIES OF THE ALGORITHM

S.2.1 Small Eigenvalues and the *noise* space

In practice, we might want to push features with very small eigenvalues (e.g. absolute value $< 10^{-8}$) to the *noise* space. The sign of these very small values might be a result of numerical issues and it makes almost no difference with respect to the cost function. In addition, smaller *clustered* spaces are of an advantage in terms of understandability and explain-ability of the respective clustering.

S.3 EXAMPLE FOR THE OPTIMIZATION OF V

In this section we explain crucial steps for optimizing the cost function w.r.t V on a small example. Let us assume we have a small six-dimensional dataset and we assume that it contains three clusterings with three subspaces S_1, S_2, S_3 and no *noise* space. The concrete number of clusters in each clustering is for this example not relevant. Further, we assume that the current rotation matrix V_{init} rotates the data space arbitrarily. In addition, we assume that we already performed the update step and assigned each data point within each subspace to the nearest cluster center. Further, we assume that we already updated all cluster centers $\boldsymbol{\mu}_{j,i}$ and that we determined the sum of scatter matrices for each subspace Σ_j according to Eq. 4. That is, the next step is to optimize V. For this, we assume that currently the first subspace S_1—corresponding to the first clustering—consists of three features of the *rotated space*: the first, the fourth and the second dimension ordered by their importance for the first clustering. The second subspace S_2 consists only of the fifth dimension. Last but not least, S_3 consists of the third and sixth dimension. Therefore, the projection matrices look like this:

$$P_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad P_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad P_3 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Now we want to update V_{init} such that the costs are minimized. Since we have more than two subspaces, we need to apply the trick described in Section 2.2.3. We want to sequentially update V_{init} with a series of update matrices $V_{i,j}^{\langle \text{f} \rangle}$ which optimize the costs for each pair of subspaces. For our example this would look like this: $V = V_{2,3}^{\langle \text{f} \rangle} V_{1,3}^{\langle \text{f} \rangle} V_{1,2}^{\langle \text{f} \rangle} V_{\text{init}}$.

In the following we show, how we can find the first rotation matrix $V_{1,2}^{\langle \text{f} \rangle}$ that updates V_{init} w.r.t the first two clusterings and their corresponding subspaces S_1 and S_2. For this, we have to project them onto a combined subspace. The projection matrix for the combined space $P_{1,2}$ looks like this:

$$P_{1,2} = [P_1 \;\; P_2] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

This means we have the following mapping between the dimensions:

$$\begin{aligned}\text{Rotated Space} &\leftrightarrow \text{Combined Subspace}\\ 1 &\leftrightarrow 1\\ 4 &\leftrightarrow 2\\ 2 &\leftrightarrow 3\\ 5 &\leftrightarrow 4\end{aligned}$$

As described in the paper, we do not have to project the whole dataset onto the combined space, but only need the sum of the scatter matrices $\Sigma_1^{\langle c\rangle}$ and $\Sigma_2^{\langle c\rangle}$ of each clustering in the combined subspace, which can be calculated based on the Σ_1 and Σ_2 of the original data space. Following the definition of Σ_j in Eq. 4 for our combined subspace this means:

$$\begin{aligned}\Sigma_1^{\langle c\rangle} &= \sum_{i=1}^{k_1} \sum_{\mathbf{x}\in C_{1,i}} \left(P_{1,2}^\top V_{\text{init}}^\top \mathbf{x} - P_{1,2}^\top V_{\text{init}}^\top \boldsymbol{\mu}_{1,i}\right)\\ &\qquad \left(P_{1,2}^\top V_{\text{init}}^\top \mathbf{x} - P_{1,2}^\top V_{\text{init}}^\top \boldsymbol{\mu}_{1,i}\right)^\top\\ &= P_{1,2}^\top V_{\text{init}}^\top \Sigma_1 V_{\text{init}} P_{1,2}\end{aligned}$$

and analogous for $\Sigma_2^{\langle c\rangle}$.

We can therefore perform the eigen-decomposition on $[\Sigma_1^{\langle c\rangle} - \Sigma_2^{\langle c\rangle}] = P_{1,2}^\top V_{\text{init}}^\top [\Sigma_1 - \Sigma_2] V_{\text{init}} P_{1,2}$.

Let us assume that we get the eigenvalues $e_{1,2} = [w\, x\, y\, z]$ *sorted in ascending order*, where $w, x < 0$ and $y, z > 0$ and we get the following corresponding eigenvectors (columns)

$$V_{1,2}^{\langle c\rangle} = \begin{bmatrix} a & e & i & m\\ b & f & j & n\\ c & g & k & o\\ d & h & l & p\end{bmatrix},$$

that build up the rotation matrix $V_{1,2}^{\langle c\rangle}$ that minimizes the costs within the combined subspace.

We need to translate this rotation within the combined subspace into an update for the rotation in the full-dimensional space. This can be done via Eq. 5. For the mapping given above, the corresponding rotation looks like this:

$$V_{1,2}^{\langle f\rangle} = \begin{bmatrix} a & i & 0 & e & m & 0\\ c & k & 0 & g & o & 0\\ 0 & 0 & 1 & 0 & 0 & 0\\ b & j & 0 & f & n & 0\\ d & l & 0 & h & p & 0\\ 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix}.$$

Since we have in this example two negative and two positive eigenvalues, it means that the optimal dimensionality of both subspaces has changed and is now equal to two: $m_1 = m_2 = 2$. In addition, we have to update the projection matrices P_1 and P_2. We still want to account for the feature importance and therefore we have to invert the order for the second clustering: the direction corresponding to the eigenvector z is the most important feature and the direction y is the second most important feature. Thus,—after updating the rotation matrix—the dimension of the rotated space mapped onto the fourth dimension of the combined subspace is the most important dimension for the second clustering and the dimension of the rotated space mapped onto the third dimension of the combined space is the second most important feature for the second clustering:

$$P_1 = \begin{bmatrix} 1 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 1\\ 0 & 0\\ 0 & 0\end{bmatrix}, \quad P_2 = \begin{bmatrix} 0 & 0\\ 0 & 1\\ 0 & 0\\ 0 & 0\\ 1 & 0\\ 0 & 0\end{bmatrix}.$$

Now we can update the rotation matrix $V = V_{1,2}^{\langle f\rangle} V_{\text{init}}$ and perform the same operation for the next clustering pair.