Input: $\mathcal{X}, k, \mathcal{L}, d$

Step 1 $*$*Preprocessing, Section 5.1*$*$

Construct $G(\mathcal{X}, E)$ by connecting each element in $\mathcal{X}$ with its k-nearest neighbors.

for all $x \in \mathcal{X}$ **do**

$N_x = \{y \in \mathcal{X} : (x, y) \in E\}$ $*$*Compute neighborhoods*$*$

$[N_x^0] = USV^T$

where $[N_x^0]$ is the data matrix formed by the elements in N_x shifted to the origin of $\mathbb{R}^N$ and U, S, V are the results of its d-rank SVD.

$M_x = U$ $*$*Compute tangent spaces*$*$

end for

Step 2 $*$*Greedy computation of partition* $\mathbf{C}^*_{\mathcal{L}}$$*$

$n = |\mathcal{X}|$, $\lambda = 0$, $\mathbf{C}^*_n = \{\{x\} : x \in \mathcal{X}\}$ $*$*Initialization*$*$

for $\lambda < n - \mathcal{L}$ **do** $*$*Greedy merging, Section 5.2*$*$

$$\left(C'_i, C'_j\right) = \underset{\substack{C_i, C_j \in \mathbf{C}^*_{n-\lambda} \\ \psi(C_i, C_j) \text{ is true}}}{\operatorname{argmin}} \tilde{d}\left(C_i, C_j\right) \quad *\textit{Eq. (23)}*$$

$\mathbf{C}^*_{n-\lambda+1} = (\mathbf{C}^*_{n-\lambda} \backslash \{C'_i, C'_j\}) \cup \{C'_i \cup C'_j\}$

Compute $M_{C'_i \cup C'_j}$ $*$*Eq. (3)*$*$

$\lambda = \lambda + 1$

end for

for $C_i \in \mathbf{C}^*_{\mathcal{L}}$ **do** $*$ *Compute the final flats* $F_i$$*$

$[C^0_{m_i}] = USV^T$

where m_i is the sample mean of C_i, $[C^0_{m_i}]$ is the data matrix formed by the samples in C_i shifted by m_i and U, S, V are the results of its d-rank SVD.

$F_i = U$

end for

Output: $\mathbf{C}^*_{\mathcal{L}}$, $\mathbf{F}$