Supplementary materials of “NLRRC: a novel clustering method of jointing non-negative LRR and random walk graph regularized NMF for single-cell type identification”

Juan Wang, Lin-Ping Wang, Shasha Yuan, Feng Li, Jin-Xing Liu, Member, IEEE, Jun-Liang Shang, Member, IEEE

Support materials

1. The optimization of NLRRC

The objective function of NLRRC is not convex. The multiplication update rule is used to iteratively update the three variables $\mathbf{Z}$, $\mathbf{B}$, and $\mathbf{C}$ in the objective function.

First, the auxiliary variable $\mathbf{S}$ is introduced to discretize the objective function of NLRRC.

Then, Lagrange multipliers $\phi=\left[\phi_{ik}\right]$ and $\varphi=\left[\varphi_{kj}\right]$ are introduced to constrain $\mathbf{B}\geq\mathbf{0}$ and $\mathbf{C}\geq\mathbf{0}$, respectively. The objective function of NLRRC is rewritten as follows:

$$\begin{aligned} L(\mathbf{Z},\mathbf{S},\mathbf{B},\mathbf{C})=&\left\|\mathbf{X}-\mathbf{X}\mathbf{S}\right\|_F^2+\lambda\left\|\mathbf{Z}\right\|_*+\left\|\mathbf{S}-\mathbf{B}\mathbf{C}\right\|_F^2 \\ &+\beta tr(\mathbf{C}\mathbf{L}_{RWG}\mathbf{C}^T)+\mu(\left\|\mathbf{Z}-\mathbf{S}\right\|_F^2+\left\|\mathbf{C}^T\mathbf{C}-\mathbf{I}\right\|_F^2) \\ &+\left\langle\mathbf{C}_1,\mathbf{Z}-\mathbf{S}\right\rangle+\left\langle\mathbf{C}_2,\mathbf{C}^T\mathbf{C}-\mathbf{I}\right\rangle+tr(\phi\mathbf{B})+tr(\varphi\mathbf{C}), \end{aligned} \tag{1}$$

where μ represents the penalty parameter, $\mathbf{C}_1\in\mathbf{R}^{n\times n}$ and $\mathbf{C}_2\in\mathbf{R}^{n\times n}$ are Lagrange multipliers, respectively. $\langle\mathbf{A},\mathbf{B}\rangle$ represents the inner product of $\mathbf{A}$ and $\mathbf{B}$.

Finally, the alternating direction method of multiplier (ADMM) is used to iteratively process each variable to find the local optimal solution. The specific update process for each variable is as follows:

Step 1: Once $\mathbf{S}$, $\mathbf{B}$, and $\mathbf{C}$ are fixed, update $\mathbf{Z}$. The variable $\mathbf{Z}$ is written as

$$L(\mathbf{Z})=\lambda\left\|\mathbf{Z}\right\|_*+\mu\left\|\mathbf{Z}-\mathbf{S}\right\|_F^2+\left\langle\mathbf{C}_1,\mathbf{Z}-\mathbf{S}\right\rangle, \tag{2}$$

Equation (2) is a quadratic optimization problem with low-rank constraints. Equation (2) is solved using the soft-threshold method to obtain the closed-form solution shown below.

$$\mathbf{Z}^{k+1}=soft_{\lambda,\frac{1}{\mu}}(\mathbf{S}^k-\frac{\mathbf{C}_1^k}{\mu}), \tag{3}$$

where $soft_{\lambda,\frac{1}{\mu}}(.)$ represents the soft-threshold operator.

Step 2: When $\mathbf{Z}$, $\mathbf{B}$, and $\mathbf{C}$ are fixed, update $\mathbf{S}$. $\mathbf{S}$ can be expressed mathematically as

$$\begin{aligned} L(\mathbf{S})=&\left\|\mathbf{X}-\mathbf{X}\mathbf{S}\right\|_F^2+\left\|\mathbf{S}-\mathbf{B}\mathbf{C}\right\|_F^2 \\ &+\mu\left\|\mathbf{Z}-\mathbf{S}\right\|_F^2+\left\langle\mathbf{C}_1,\mathbf{Z}-\mathbf{S}\right\rangle. \end{aligned} \tag{4}$$

Equation (4) is optimized as follows:

$$\mathbf{S}^{k+1}=\left(\mathbf{X}^T\mathbf{X}+\mathbf{I}+\mu\mathbf{I}\right)^{-1}\left(\mathbf{X}^T\mathbf{X}+\mathbf{B}^k\mathbf{C}^k+\mu\mathbf{Z}^{k+1}+\mathbf{C}_1^k\right). \tag{5}$$

then, we set $\mathbf{S}^{k+1}=\left((\mathbf{S}^{k+1})^T+\mathbf{S}^{k+1}\right)/2$ to ensure the symmetry of similar matrices.

Step 3: Update $\mathbf{B}$ and $\mathbf{C}$. L to $\mathbf{B}$ and $\mathbf{C}$ derivative is

$$\frac{\partial L}{\partial\mathbf{B}^k}=\mathbf{B}^k\mathbf{C}^k\mathbf{C}^{kT}-\mathbf{S}^{k+1}\mathbf{C}^{kT}+\phi, \tag{6}$$

$$\frac{\partial L}{\partial\mathbf{C}^k}=\mathbf{B}^{kT}\mathbf{B}^k\mathbf{C}^k+\mathbf{B}^k\mathbf{C}_2+\beta\mathbf{C}^k\mathbf{L}-\mathbf{B}^{kT}\mathbf{S}^{k+1}+\varphi, \tag{7}$$

The iteration formula of objective functions (6) and (7) is expressed using the Karush-Kuhn-Tucker (KKT) [1] under the conditions of $\phi_{ik}w_{ik}=0$ and $\varphi_{kj}h_{kj}=0$, respectively, as

$$\mathbf{B}_{ik}^{k+1}\leftarrow\mathbf{B}_{ik}\frac{\left(\mathbf{S}^{k+1}(\mathbf{C}^k)^T\right)_{ik}}{\left(\mathbf{B}^k\mathbf{C}^k(\mathbf{C}^k)^T\right)_{ik}}, \tag{8}$$

$$\mathbf{C}_{kj}^{k+1}\leftarrow\mathbf{C}_{kj}\frac{\left((\mathbf{B}^{k+1})^T\mathbf{S}+\beta\mathbf{C}^k\mathbf{W}\right)_{kj}}{\left((\mathbf{B}^{k+1})^T\mathbf{B}^{k+1}\mathbf{C}^k+\mathbf{C}^k\mathbf{C}_2^k+\beta\mathbf{C}^k\mathbf{D}\right)_{kj}}, \tag{9}$$

where $\mathbf{W}$ is the symmetric weight matrix and $\mathbf{D}$ is the degree matrix of the edge.

Step 4. Update $\mathbf{C}_1$ and $\mathbf{C}_2$

$$\mathbf{C}_1^{k+1}=\mathbf{C}_1^k+\mu\left(\mathbf{Z}^{k+1}-\mathbf{S}^{k+1}\right), \tag{10}$$

$$\mathbf{C}_2^{k+1}=\mathbf{C}_2^k+\mu\left((\mathbf{C}^{k+1})^T\mathbf{C}^{k+1}-\mathbf{I}\right). \tag{11}$$

Repeat the preceding steps until the set constraints or the maximum number of iterations are met. Algorithm 1 depicts the specific update process of LRRNC.

2. Time complexity analysis

Algorithm 1 The algorithm for LRRNC.

Input: Gene representation matrix $\mathbf{X}$,
Parameter k, λ and β.

Initialization: Initialize $\mathbf{Z}, \mathbf{S}, \mathbf{B}, \mathbf{C}$,
maximum number of iterations and stop error.

$$\mathbf{Z}=\mathbf{0}, \mathbf{S}=\mathbf{0}, \mathbf{B}=\mathbf{0}, \mathbf{C}=\mathbf{0}, \mu=10^{-4},$$
$$\max Iter=100, tol_1=10^{-5}, tol_2=10^{-5}.$$

While not convergence do

1) Update the variable $\mathbf{Z}$ with Equation (3),
2) Update the variable $\mathbf{S}$ with Equation (5),
3) Update the variable $\mathbf{B}$ and $\mathbf{C}$ with Equation (8) and Equation (9) respectively.
4) Update the Lagrange multiplier $\mathbf{C}_1$ and $\mathbf{C}_2$ with Equation (10) and Equation (11), respectively.

Until the maximum iteration is reached or satisfies the following convergence conditions:

$$\max\left(\left\|\mathbf{X}-\mathbf{X}\mathbf{Z}\right\|_F^2, \left\|\mathbf{Z}-\mathbf{B}\mathbf{C}\right\|_F^2\right) \Big/ \max\left(\left\|\mathbf{X}\mathbf{Z}\right\|_F^2, \left\|\mathbf{B}\mathbf{C}\right\|_F^2\right) < tol_1,$$

$$\max\left(\left\|\mathbf{Z}^{k+1}-\mathbf{Z}^k\right\|_F^2, \left\|\mathbf{S}^{k+1}-\mathbf{S}^k\right\|_F^2, \left\|\mathbf{B}^{k+1}-\mathbf{B}^k\right\|_F^2, \left\|\mathbf{C}^{k+1}-\mathbf{C}^k\right\|_F^2, \right) \Big/ \max\left(\left\|\mathbf{X}\right\|_F^2\right) < tol_2.$$

End while

Output: Optimal matrix $\mathbf{C}$

The computational complexity of the NLRRC method is primarily caused by the updating of variables $\mathbf{Z}, \mathbf{S}, \mathbf{B}$ and $\mathbf{C}$. The symbol O represents the computational complexity. Let n represent the number of cell samples. The m represents the feature numbers of each sample. The k is the number of clusters. The t represents the number of model iterations. **Algorithm 1** depicts the four main steps of NLRRC, with the first, second, and third steps requiring the highest calculation cost. The calculation cost of the first step is primarily generated by SVD solving the soft threshold function of $\mathbf{Z}$, and the calculation complexity is $O\left(n^3\right)$. The second step is $O\left(n^3\right)$, which is primarily generated by solving the similarity matrix. The time complexity generated by NMF is $O\left(n^2 k\right)$ in the third step. Additionally, $O\left(m n^2\right)$ is required for data graph construction. The computational complexity for RWGR is approximately $O\left(n^2\right)$. The total computational complexity of NLRRC is $O\left(2n^3 t + n^2 k t + m n^2 + n^2\right)$, if the model is iterated t times.

3. Evaluation metrics

Normalized Mutual Information (NMI) [2] and the Adjusted Rand Index (ARI) [3] are used to evaluate clustering performance. The NMI theoretically explains the consistency between the predicted cluster label $\mathbf{Y}=\{Y_1, Y_2, ..., Y_k\}$ and the actual cluster label $\mathbf{T}=\{T_1, T_2, ..., T_k\}$. The following is the definition of NMI.

$$NMI(\mathbf{Y}, \mathbf{T}) = \frac{MI(\mathbf{Y}, \mathbf{T})}{\left[E(\mathbf{Y})+E(\mathbf{T})\right]/2}, \tag{12}$$

where $MI(.,.)$ and $E(.)$ represent mutual information and information entropy, respectively.

ARI evaluates performance by calculating similarity between the predicted label $\mathbf{Y}$ and the actual label $\mathbf{T}$. ARI is mathematically expressed as follows:

$$ARI(\mathbf{Y}, \mathbf{T}) = \frac{\binom{n}{2}\left(a_{\mathbf{YT}}+a\right) - \left[\begin{matrix}\left(a_{\mathbf{YT}}+a_{\mathbf{T}}\right)\left(a_{\mathbf{YT}}+a_{\mathbf{Y}}\right) \\ +\left(a_{\mathbf{Y}}+a\right)\left(a_{\mathbf{T}}+a\right)\end{matrix}\right]}{\binom{n}{2}^2 - \left[\begin{matrix}\left(a_{\mathbf{YT}}+a_{\mathbf{T}}\right)\left(a_{\mathbf{YT}}+a_{\mathbf{Y}}\right) \\ +\left(a_{\mathbf{Y}}+a\right)\left(a_{\mathbf{T}}+a\right)\end{matrix}\right]}, \tag{13}$$

where $a_{\mathbf{YT}}$ denotes the number of cells shared by prediction label $\mathbf{Y}$ and real label $\mathbf{T}$. $a_{\mathbf{T}}$ and $a_{\mathbf{Y}}$ are the numbers of single-cell in real label $\mathbf{T}$ and prediction label $\mathbf{Y}$, respectively. a denotes the number of cells that are not shared by real label $\mathbf{T}$ and prediction label $\mathbf{Y}$.

The value range of NMI is [0,1], and the value range of ARI is [- 1,1]. The higher the values of NMI and ARI are, the better the consistency between the predicted and actual labels.

4. Convergence

The convergence of the NLRRC is described in detail in this section. The results are displayed in Fig. 1. The vertical axis Y represents the iteration error, and the horizontal axis X represent the number of iterations. After the parameters of the model are fixed, count the error of each iteration until the maximum number of iterations or the convergence conditions set by us are met. In Fig. 1, with the increase of iteration times, we found that NLRRC converges to a lower error and tends to be stable. As a result, the NLRRC method converges well.

5. Selecting the number of clusters

Estimating the number of clusters is one of the most important clustering steps. The Gap Statistics [4] is a reliable method for predicting the number of clusters. We use the Gap Statistic to determine the number of clusters, as shown in TABLE I. The number of correctly predicted clusters is shown in bold. The following is the operating principle of the Gap Statistic.

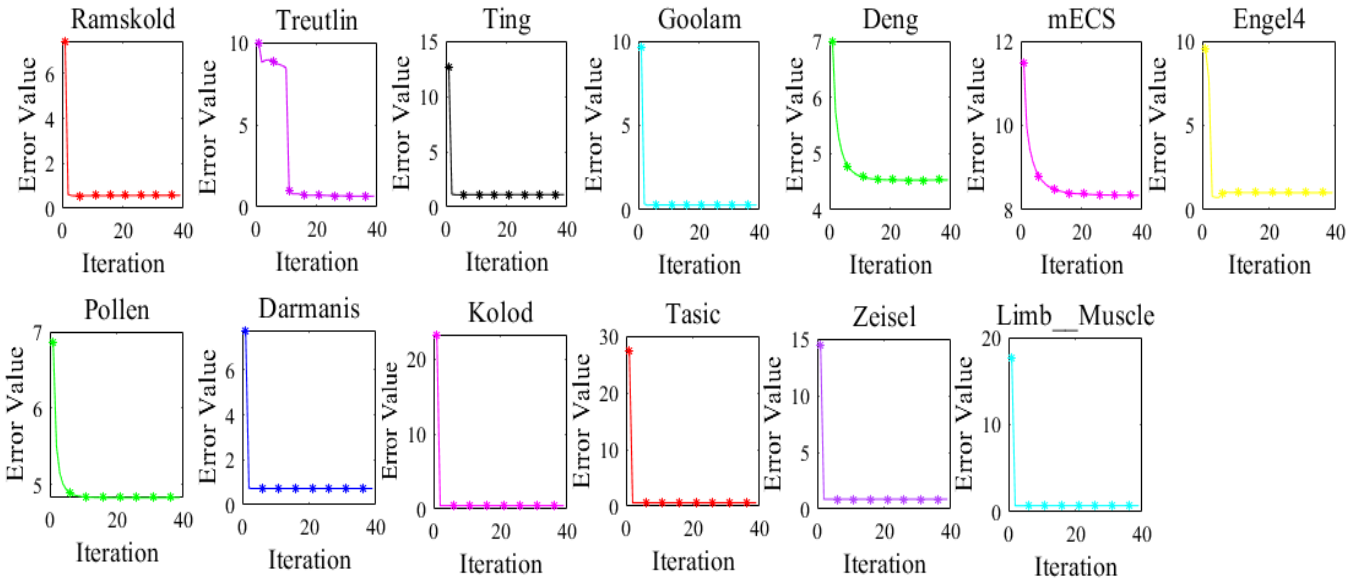

Fig. 1. Convergence curves of NLRRC on scRNA-seq datasets.

TABLE I
EVALUATE CLUSTER NUMBER k

Datasets	Ramskold	Treutlein	Ting	Goolam	Deng	mECS	Engel4	Pollen	Darmanis	Kolod	Tasic	Zeisel	Limb_Muscle
Real cluster number	7	5	5	5	7	3	4	11	8	3	48	9	6
Predict cluster number	6	**5**	**5**	**5**	**7**	**3**	**4**	12	9	**3**	**48**	**9**	7

Let $Gap_{(k)}$ represent the difference between the expected value of the random sample and the Euclidean distance between the observed sample points in the clusters. If the optimal cluster number is k, the sum of Euclidean distances between observation samples within each cluster is the smallest and the value of $Gap_{(k)}$ is the largest. The Gap Statistics employs this principle to determine the optimal cluster number by searching for the k that maximizes $Gap_{(k)}$.

In TABLE I, we discover that the Gap Statistics accurately predict the number of clusters in nine datasets, including Treutlein, Ting, Goolam, Deng, mECS, Engel4, Kolod, Tasic, and Zeisel.

REFERENCES

[1] A. Dreves, et *al.*, "On the solution of the KKT conditions of generalized Nash equilibrium problems," *Siam J. Optim*, vol. 21, no. 3, pp. 1082-1108, 2011.

[2] A. Strehl, and J. Ghosh, "Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions," *J Mach Learn Res*, vol. 3, no. 3, pp. 583-617, 2002.

[3] S. Wagner, and D. Wagner, "Comparing Clusterings - An Overview," *Karlsruhe: Universität Karlsruhe, Fakultät für Informatik*, no.1,pp.1-19, 2007.

[4] R. Tibshirani, and W. Hastie, "Estimating the number of clusters in a data set via the gap statistic," J*ournal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 2, pp. 411-423, 2001.