GRAPH REGULARIZED NON-NEGATIVE MATRIX FACTORIZATION WITH SPARSE CODING

Binghui Wang, Meng Pang, Chuang Lin*, and Xin Fan
School of Software of Dalian University of Technology, Dalian, China
Binghui.wang89@gmail.com, PangMeng252484463@163.com; linchuang_78@126.com, xin.fan@dlut.edu.cn
*Corresponding author: Chuang Lin

ABSTRACT

Matrix factorization techniques have been frequently utilized in pattern recognition and machine learning. Among them, Non-negative Matrix Factorization (NMF) has received considerable attention because it represents the naturally occurring data by parts of it. On the other hand, from the geometric perspective, the data is usually sampled from a low dimensional manifold embedded in a high dimensional ambient space. One hopes then to find a compact representation which uncovers the intrinsic geometric structure. In this paper, we propose a novel method, called Graph Regularized Non-negative Matrix Factorization with Sparse Coding (GRNMF_SC), the new model can learn much sparser representation and more discriminating power via imposing sparse constraint and Laplacian regularization explicitly. Experimental results on the ORL and Yale databases demonstrate encouraging performance of the proposed algorithm when compared with the state-of-the-art algorithms.

Index Terms— NMF, Laplacian Regularization, Sparse Representation, Face Recognition

1. INTRODUCTION

The techniques for matrix factorization have become popular in recent years in pattern recognition and machine learning areas. Since the input data matrix is of very high dimension, it makes *learning from example* infeasible [1]. Therefore, researchers hope to find two or more lower dimensional matrices whose product provides a good approximation to the original one. Previous studies have shown that there is psychological and physiological evidence for parts-based representation in the human brain [2-4]. Non-negative Matrix Factorization (NMF) method is thus proposed to learn the parts of objects like human faces and text documents [5-6]. NMF aims to find two non-negative matrices whose product provides a good approximation to the original one. The non-negative constraints lead to a parts-based representation because they allow only additive, not subtractive, combinations. NMF has been shown to be superior to SVD in face recognition [7] and document clustering [8]. Though NMF has so many advantages, several authors have noted the shortcomings of standard NMF, and suggested extensions and modifications.

One of the shortcomings is that NMF can only be applied to data containing non-negative values. Ding [9] proposed a semi-NMF approach which relaxes the non-negative constraint. Xu [10] proposed a Concept Factorization approach in which the input data matrix is factorized into three matrices. Such modification makes it possible to kernerlize concept factorization.

Another shortcoming of NMF is that it does not always result in parts-based representations. Several researchers addressed this problem by incorporating the sparseness constraints [11-13]. These approaches extended the NMF framework to include an adjustable sparseness parameter. With a suitable sparseness parameter, these approaches are guaranteed to result in parts-based representations.

Recently, various researchers [14-17] have considered the case when the data is drawn from sampling a probability distribution that has support on or near to a sub-manifold of the ambient space. All these algorithms use the so-called *locally invariant idea* [18], i.e., the nearby points are likely to have similar embedding. It has been shown that learning performance can be significantly enhanced if the geometrical structure is exploited and the local invariance is considered. Benefiting from the progress in matrix factorization and manifold learning, in the very recent, He and Cai [19-21] proposed several manifold regularized non-negative matrix factorization methods, which explicitly considers the local invariance by constructing a nearest neighbor graph to encode the geometrical information of the data space.

Considering the important property of sparse constraint and intrinsic property captured by Laplacian regularization, we proposed a novel method called Graph Regularized Non-negative Matrix Factorization with Sparse Coding (GRNMF_SC), which can not only lead to a new parts-based representation which respects the geometrical structure of the data space, but also learn a much sparser representation. We also develop an optimization method to solve the objective function and the convergence proof is provided.

The rest of the paper is organized as follows: in Section 2, we give a brief review of NMF. Section 3 introduces our GRNMF_SC algorithm and provides a convergence proof of our optimization method. Experimental results on face recognition are presented in Section 4. Finally, we provide some conclusions and future work in Section 5.

2. A BRIEF REVIEW OF NMF

Non-negative Matrix Factorization (NMF) [6] is a matrix factorization algorithm that focuses on the analysis of data matrices whose elements are nonnegative. Thus, NMF allows only additive, not subtractive, combinations among different basis. For this reason, it is believed to be compatible to the intuitive notion of combining parts to form a whole and how NMF learn parts-based representation.

Given a data matrix $\mathbf{X} = [\mathbf{X}\_1, \ldots, \mathbf{X}\_n] \in \mathbf{R}^{m \times n}$, each column of $\mathbf{X}$ is a sample vector with non-negative values. NMF aims to find two non-negative matrices $\mathbf{U} \in \mathbf{R}^{m \times r}$ and $\mathbf{V} \in \mathbf{R}^{m \times r}$ whose product can well approximate the original matrix. In other words, it is to minimize the following cost function:

$$O = \| \mathbf{X} - \mathbf{U}\mathbf{V}^T \|_F^2 \tag{1}$$

Since the objective function is not convex in $\mathbf{U}$ and $\mathbf{V}$ together, we are not expected to find the global minimum of O. Lee [7] presented the following iterative update rule:

$$u_{ij} \Leftarrow u_{ij} \frac{(\mathbf{XV})_{ij}}{(\mathbf{U}\mathbf{V}^{\mathbf{T}}\mathbf{V})_{ij}} \tag{2}$$

$$v_{ij} \Leftarrow v_{ij} \frac{(\mathbf{X}^{\mathbf{T}}\mathbf{U})_{ij}}{(\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U})_{ij}} \tag{3}$$

It is proved that the above two equations will find local minima of the objective function O.

3. GRAPH REGULARIZED NON-NEGATIVE MATRIX FACTORIZATION WITH SPARSE CODING (GRNMF_SC)

As mentioned earlier, He & Cai [19-21] proposed a spectral graph based method named Graph Regularized Non-negative Matrix Factorization (GNMF). By preserving the intrinsic structure of the data using Laplacian regularization and incorporating it into NMF, GNMF can have more discriminating power than the standard NMF. However, since GNMF did not impose any sparse constraint to the basis matrix $\mathbf{U}$ or encoding matrix $\mathbf{V}$ explicitly, it cannot learn the sparse representation. Thus, considering the important property of sparse constraint, we regard it as a regularization incorporated into GNMF and propose the so-called Graph Regularized Non-negative Matrix Factorization with Sparse Coding (GRNMF_SC) algorithm. In the following, we first introduce the core idea of GNMF and then lead to our GNMF_SC method.

3.1. GNMF

GNMF first constructs an affinity graph to encode the geometrical information and then seeks a non-negative matrix factorization which respects the graph structure.

In order to capture the intrinsic structure of the data space, GNMF finds a new representation space to indicate two data points are sufficiently close to each other if they are connected in the graph. Procedure is as follows:

1. Consider a graph with n vertices where each vertex corresponds to a data point. Define the edge weight matrix $\mathbf{W}$ as follows:

$$\mathbf{W}_{ij} = \begin{cases} 1, & \textit{if } \ \mathbf{x}_i \in N_p(\mathbf{x}_j) \ \textit{or} \ \mathbf{x}_j \in N_p(\mathbf{x}_i) \\ 0, & \textit{otherwise} \end{cases} \tag{4}$$

where $N_p(\mathbf{x}_i)$ denotes the set of p nearest neighbors of $\mathbf{x}_i$. Define $\mathbf{L} = \mathbf{D} - \mathbf{W}$ where $\mathbf{D}$ is a diagonal matrix whose unit entries are column sums of $\mathbf{W}$, $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$.

2. Let $f_k(\mathbf{x}_i) = \mathbf{v}_{ik}$ be the function that produce the mapping of the original data point $\mathbf{x}_i$ onto the axis $\mathbf{u}_k$. GNMF then uses $\| f_k \|_M^2$ to measure the smoothness of the function f_k along the geodesics in the intrinsic geometry of the data. When we consider the case that the data is a compact sub-manifold $\mathrm{M} \in \mathbf{R}^m$, then the discrete approximation of $\| f_k \|_M^2$ is computed as follows:

$$\begin{aligned} \| f_k \|_M^2 &= \frac{1}{2} \sum_{i,j=1}^{n} (f_k(\mathbf{x}_i) - f_k(\mathbf{x}_j))^2 \mathbf{W}_{ij} \\ &= \mathbf{v}_k^T \mathbf{L} \mathbf{v}_k \end{aligned} \tag{5}$$

By minimizing $\| f_k \|_M^2$, we get a mapping function f_k which is sufficiently smooth on the data manifold. An intuitive explanation of minimizing $\| f_k \|_M^2$ is that if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are close, then $f_k(\mathbf{x}_i)$ and $f_k(\mathbf{x}_j)$ are similar to each other.

3. Finally, GNMF incorporates the constraint and minimize the new objective function:

$$O = \| \mathbf{X} - \mathbf{U}\mathbf{V}^T \|_F^2 + \lambda Tr(\mathbf{V}^{\mathbf{T}}\mathbf{L}\mathbf{V})) \tag{6}$$

with the constraint that u_{ij} and v_{ij} are non-negative. $Tr(\cdot)$ denotes the trace of a matrix. $\lambda \geq 0$ is a regularization parameter.

3.2 GRNMF_SC

In order to improve the degree of sparseness of coefficient matrix $\mathbf{V}$ as well as preserving the intrinsic structure of the high dimensional data, we add an L1-norm regularization to the coefficient matrix. By this way, we expect each sample in $\mathbf{X}$ can be represented by a linear combination of only few basis vectors in $\mathbf{U}$, thus the sparseness can be guaranteed. The new objective function is as follows:

$$\begin{aligned} & O = \| \mathbf{X} - \mathbf{U}\mathbf{V}^T \|_F^2 + \lambda Tr(\mathbf{V}^{\mathbf{T}}\mathbf{L}\mathbf{V}) + \alpha \sum_{i=1}^{n} \| v_i \|_1 \\ & s.t \quad \mathbf{U}, \mathbf{V} \geq 0, \| \mathbf{u}_j \| = 1, \forall j \end{aligned} \tag{7}$$

Finally, the multiplicative updating rules toward this objective function is:

$$v_{ij} \Leftarrow v_{ij} \frac{2[\mathbf{X}^{\mathrm{T}}\mathbf{U} + \lambda\mathbf{W}\mathbf{V}]_{ij} - \alpha}{2[\mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U} + \lambda\mathbf{D}\mathbf{V}]_{ij}} \tag{8}$$

$$u_{ij} \Leftarrow u_{ij} \frac{(\mathbf{X}\mathbf{V})_{ij}}{(\mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V})_{ij}} \tag{9}$$

$$u_{ij} \Leftarrow \frac{u_{ij}}{\sum_i u_{ij}} \tag{10}$$

GRNMF_SC has fully multiplicative update rules with two parameters. When setting $\lambda = 0$, then GRNMF_SC reduces to NNSC [11]. When setting $\alpha = 0$, GRNMF_SC reduces to GNMF. Also, we find that the updating rules of encoding Matrix $\mathbf{V}$ could be rewritten as the following gradient decent format:

$$v_{ij}^{(t+1)} = v_{ij}^{(t)} - \eta v_{ij}^{(t)} \tag{11}$$

$$\eta = \frac{[-2\mathbf{X}^{\mathrm{T}}\mathbf{U} + 2\mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U} + 2\lambda\mathbf{L}\mathbf{V}]_{ij} + \alpha}{2[\mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U} + \lambda\mathbf{D}\mathbf{V}]_{ij}} \tag{12}$$

In order to preserve the non-negative property of coefficient matrix $\mathbf{V}$, we should control the parameter α and λ to make η positive as well as small. The proof of our optimization scheme of GRNMF_SC is given in Appendix.

4. EXPERIMENTAL RESULTS

Our experiments were performed on two benchmarks: the ORL 48×48 database and Yale 32×32 database to test the recognition rates of our proposed GRNMF_SC. The famous methods: NMF [5], LNMF [7], SNMF [14] and GNMF [21] are used for comparison.

In all experiments results, we firstly report the optimal average recognition rages over 10 random splits (i.e. running 10 times) to see all methods' ability dealing with training sets with different Gm s. Secondly, to gain an insight into the intrinsic dimension of different datasets, we plot the average recognition rates versus feature dimensions of all competing methods when given different Gm s. The nearest neighbor (NN) classifier was used for all face recognition experiments. The Euclidean metric is used as our distance measure.

4.1. Recognition Results on the ORL Database

In this part, the face recognition experiment is carried out on the ORL database. As illustrated before, ORL contains $N = 400$ images with $C = 40$ person, each person has 10 samples. Firstly, to evaluate all methods' ability to dealing with small training set with different Gm s (Actually, when the size of training set is larger, the recognition rate is higher), a random subset with $Gm(= G3, G4, G5, G6, G7)$ is taken with labels to form the training set respectively. The corresponding remaining part $Pn(= P7, P6, P5, P4, P3)$ with labels is to form the testing set. For each given Gm / Pn , we average the results over 10 random splits. TABLE I shows the optimal average recognition rates obtained by NMF, LNMF, SNMF, GNMF and GRNMF_SC with the corresponding feature dimension in different Gm s over 10 random splits. Then, Fig 1 shows the average recognition rates versus feature dimensions of all competing methods when given different Gm s. It is need to explain that the optimal average recognition rates are obtained cover the subspace with entire feature dimensions. However, in the latter experiment, the results are gained cover the subspace with feature dimension chosen from 0 to 100 with a gap 10 for its convenience and representativeness. What's more, we utilize the similar way in the following Yale databases.

From TABLE I, we can see that our GRNMF_SC algorithm obtains the better optimal compared with other the other's optimal results. GNMF and SNMF methods perform comparatively to ours. However, as a whole, the LNMF scheme performs the worst in the 48×48 ORL database. What's more, NMF has a poor result next to SNMF.

4.2. Recognition Results on the Yale Database

In this subsection, the face recognition experiment is carried out on the Yale database. Yale contains $N = 165$ images with $C = 15$ individuals (classes), each class has 11 samples. Similar to ORL, a random subset with $Gm(= G4, G5, G6, G7, G8)$ is taken with labels to form the training set, and the remaining part $Pn(= P7, P6, P5, P4, P3)$ is adopted as the testing set. For each Gm / Pn , we average the results over 10 random splits. We show the optimal average recognition results in TABLE II. The average recognition rates versus dimensions varies from 0 to 100 with a gap 10 with $G7 / P4$ are displayed in Fig. 2.

Similar to the results on the ORL database, it can be seen from TABLE II our GRNMF_SC algorithm also obtains the best optimal result. However, the biggest difference between TABLE II and TABLE III is that the LNMF algorithm performs very well in the Yale face database, just next to the GRNMF_SC. What's more, The recognitions of SNMF and GNMF with different feature dimensions are very close to each other. The original NMF performs worst in the Yale database.

TABLE I
OPTIMAL AVERAGE RECOGNITION RATES (DIMENSIONS=50) ON THE ORL DATABASE WITH DIFFERENT NUMBER OF TRAINING SAMPLES OF EACH PERSON

Method	G3/P7	G4/P6	G5/P5	G6/P4	G7/P3
NMF	73.8%	79.8%	83.6%	86.0%	89.0%
	(2.0%)	(1.5%)	(2.0%)	(3.5%)	(3.5%)
LNMF	61.4%	65.4%	67.5%	71.3%	74.2%
	(8.0%)	(6.0%)	(4.0%)	(3.9%)	(4.0%)
SNMF	73.6%	80.3%	83.8%	86.3%	90.0%
	(1.0%)	(1.0%)	(2.0%)	(2.0%)	(1.5%)
GNMF	73.9%	81.9%	84.8%	87.2%	90.5%
	(1.0%)	(1.0%)	(2.0%)	(1.0%)	(1.0%)
GRNMF_SC	73.2%	82.3%	85.5%	88.7%	92.5%
	(2.0%)	(2.0%)	(1.0%)	(1.0%)	(2.0%)

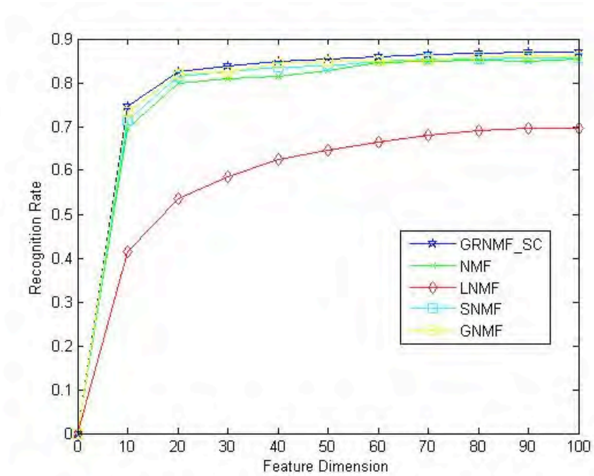

Fig.1. Recognition of NMF, LNMF, SNMF, GNMF and GRNMF_SC with different feature dimensions.(ORL 48×48)

TABLE II
OPTIMAL AVERAGE RECOGNITION RATES (DIMENSIONS=50) ON THE YALE DATABASE WITH DIFFERENT NUMBER OF TRAINING SAMPLES OF EACH PERSON

Method	G4/P7	G5/P6	G6/P5	G7/P4	G8/P3
NMF	62.8% (5.7%)	64.4% (3.5%)	68.3% (4.0%)	77.3% (1.0%)	81.8% (2.0%)
LNMF	61.5% (1.0%)	68.9% (1.0%)	70.7% (1.5%)	79.0% (2.0%)	83.1% (1.0%)
SNMF	63.6% (3.5%)	65.3% (2.5%)	70.0% (2.0%)	77.7% (3.0%)	82.2% (2.0%)
GNMF	64.8% (2.9%)	66.7% (3.0%)	72.0% (2.0%)	80.0% (1.5%)	83.1% (2.0%)
GRNMF_SC	65.3% (2.5%)	70.2% (3.0%)	73.3% (3.0%)	81.7% (1.5%)	84.4% (2.0%)

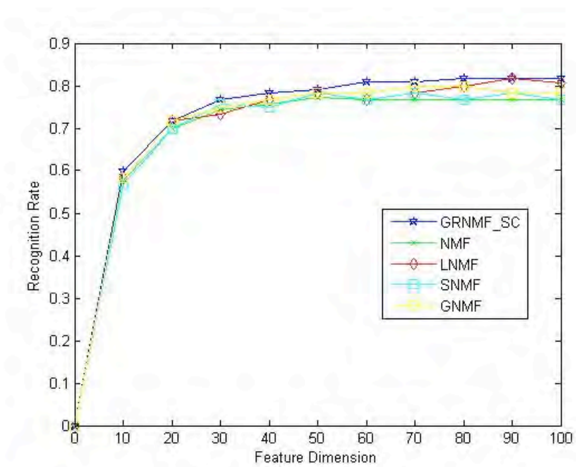

Fig.2. Recognition of NMF, LNMF, SNMF, GNMF and GRNMF_SC with different feature dimensions.(Yale 32×32)

5. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a novel method for matrix factorization, called Graph regularized Non-negative Matrix Factorization with sparse coding (GRNMF_SC). GRNMF_SC algorithm absorbs advantages of GNMF which has more discriminating power than original NMF as well as the concept of sparse coding. GRNMF_SC improves the recognition rates when comparing with Classical NMF, SNMF and GNMF.

GRNMF_SC algorithm is a novel way of NMF which incorporates discriminative theory and property of sparseness. It also creates a research direction for us to do further research into the two fields simultaneously.

It should be noted that, because the proposed GRNMF_SC method has a fully multiplicative update rules and two parameters, one should carefully select the two parameters to strike a balance between the weight of sparseness and discrimination according to the demands of real applications.

6. REFERENCES

[1] R. O. Duda, P. E. Hart. *Pattern Classification*. Wiley-Interscience, Hoboken, NJ, 2nd edition, 2000.

[2] S. E. Palmer. “Hierarchical structure in perceptual representation.” *Cognitive Psychology*, 9:441–474, 1977

[3] E. Wachsmuth, and M. W. Oram. “Recognition of objects and their component parts: Responses of single units in the temporal cortex of the macaque.” *Cerebral Cortex*, 4:509–522, 1994.

[4] N. K. Logothetis and D. L. Sheinberg, “Visual object recognition.” *Annual Review of Neuroscience*, 19:577–621, 1996.

[5] D. D. Lee and H. S. Seung. “Learning the parts of objects by nonnegative matrix factorization.” *Nature*, 401:788–791, 1999.

[6] D. D. Lee and H. S. Seung. “Algorithms for non-negative matrix factorization.” *In Advances in Neural Information Processing Systems*, 13. 2001.

[7] S. Z. Li, X. Hou, H. Zhang, and Q. Cheng. “Learning spatially localized, parts-based representation.” *IEEE Computer Society Conf.on Computer Vision and Pattern Recognition,*207–212, 2001.

[8] W. Xu, X. Liu, and Y. Gong. “Document clustering based on non-negative matrix factorization.” *In Proc. 2003 Int. Conf. on Research and Development in Information Retrieval (SIGIR’03)*, 267–273, Toronto, Canada, Aug. 2003.

[9] C. Ding, T. Li, and M. Jordan. “Convex and semi-nonnegative matrix factorizations for clustering and low-dimension representation.” Technical report, LBNL-60428, , 2006.

[10] W. Xu and Y. Gong. “Document clustering by concept factorization.” *In Proc. Inter. Conf. on Research and Development in Information Retrieval*. 202–209, Sheffield, UK, July 2004.

[11] P. O. Hoyer. “Non-negative sparse coding.” *IEEE Workshop on Neural Networks for Signal Processing*, 557–565, 2002.

[12] P. O. Hoyer. “Non-negative matrix factorization with sparseness constraints.” *Journal of Machine Learning Research*, 5:1457–1469, 2004.

[14] W. Liu, N. Zheng, and X. Lu. “Non-negative matrix factorization for visual coding.” *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP’2003),* 3:293-296, 2003.

[14]M. Belkin and P. Niyogi. “Laplacian eigenmaps and spectral techniques for embedding and clustering.” *In Advances in Neural Information Processing Systems 14*, pages 585–591. MIT Press, Cambridge, MA, 2001.

[15] M. Belkin, P. Niyogi. “Manifold regularization: A geometric framework for learning from examples.” *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[16]S. Roweis and L. Saul. "*Nonlinear dimensionality reduction by locally linear embedding*." Science, 290:2323–2326, 2000.
[17] J. Tenenbaum, V. de Silva, and J. Langford. "*A global geometric framework for nonlinear dimensionality reduction*." Science, 290:2319–2323, 2000.
[18] R. Hadsell, S. Chopra, and Y. LeCun. "Dimensionality reduction by learning an invariant mapping." *In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, pages 1735–1742, 2006.
[19] D. Cai, X. He, X. Wu, and J. Han. "Non-negative matrix factorization on manifold." I*n Proc. Int. Conf. on Data Mining (ICDM'08)*, 2008.
[20] D. Cai, X. He, X. Wang, H. Bao, and J. Han. "Locality preserving nonnegative matrix factorization." *In Proc. 2009 Int. Joint Conference on Artificial Intelligence (IJCAI'09)*, 2009.
[21] D. Cai, X.F. He, J.W. Han, and T. S. Huang, "Graph Regularized Non-negative Matrix Factorization for Data Representation." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2011.

ACKNOWLEDGMENT

Chuang Lin thanks for the support of Natural Science Foundation of China under grant Nos. 61033012, 611003177, and 61272371, Fundamental Research Funds for the Central Universities, No.DUT12JR07, and Specialized Research Fund for the Doctoral Program of Higher Education, No. 20120041120046.

APPENDIX

The core idea of the proof process is that we can make use of an auxiliary function technique as the EM algorithm, then take turns updating Basis matrix $\mathbf{U}$ and Encoding matrix $\mathbf{V}$. We begin with the definition of the auxiliary function.

Definition 1 $G(v,v')$ is an auxiliary function for $F(v)$ if the conditions

$$G(v,v') \geq F(v), \quad G(v,v) = F(v)$$

are satisfied.

The reason why the auxiliary function is vital for proving because of the following lemma .

Lemma 2 If G is an auxiliary function of F, then F is non-increasing under the update

$$v^{t+1} = \arg\min_{v} G(v,v^t)$$

Proof: $F(v^{t+1}) \leq G(v^{t+1},v^t) \leq G(v^t,v^t) = F(v^t)$

Firstly, we start to derive the multiplicative update steps of Encoding Matrix V. As we know, the objective function of GRNMF_SC can be rewritten as follows:

$$O = \sum_{i=1}^{m}\sum_{j=1}^{n}(x_{ij} - \sum_{l=1}^{k}u_{il}v_{jl})^2 + \lambda\sum_{l=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{n}v_{jl}\mathbf{L}_{ji}v_{il} + \alpha\sum_{l=1}^{k}\sum_{j=1}^{n}v_{lj}$$

Considering any element v_{ij} in $\mathbf{V}$, we use F_{ij}' and F_{ij}'' to denote the first-order derivative and second-order derivative of objective function O .

$$F_{ij}' = (\frac{\partial O}{\partial \mathbf{V}})_{ij} = [-2\mathbf{X}^{\mathbf{T}}\mathbf{U} + 2\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U} + 2\lambda\mathbf{L}\mathbf{V}]_{ij} + \alpha$$

$$F_{ij}'' = 2(\mathbf{U}^{\mathbf{T}}\mathbf{U})_{jj} + 2\lambda\mathbf{L}_{ii}$$

We define the auxiliary function :

$$G(v,v_{ij}^{(t)}) = F_{ij}(v_{ij}^{(t)}) + F_{ij}'(v_{ij}^{(t)})(v - v_{ij}^{(t)}) + \frac{(\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U})_{ij} + \lambda(\mathbf{D}\mathbf{V})_{ij}}{v_{ij}^{(t)}}(v - v_{ij}^{(t)})^2$$

We need to prove

$$G(v,v) = F(v)$$

$$G(v,v_{ij}^{(t)}) \geq F_{ij}(v).$$

It is trivial to prove $G(v,v) = F(v)$, next we begin to prove $G(v,v_{ij}^{(t)}) \geq F_{ij}(v)$.We then derive the Taylor series expansion of $F_{ij}(v)$.

$$F_{ij}(v) = F_{ij}(v_{ij}^{(t)}) + F_{ij}'(v_{ij}^{(t)})(v - v_{ij}^{(t)}) + [(\mathbf{U}^{\mathbf{T}}\mathbf{U})_{jj} + \lambda\mathbf{L}_{ii}](v - v_{ij}^{(t)})^2$$

Because

$$(\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U})_{ij} = \sum_{l=1}^{k}v_{il}^{(t)}(\mathbf{U}^{\mathbf{T}}\mathbf{U})_{lj} \geq v_{ij}^{(t)}(\mathbf{U}^{\mathbf{T}}\mathbf{U})_{jj}$$

$$\lambda(\mathbf{D}\mathbf{V})_{ij} = \lambda\sum_{l=1}^{n}\mathbf{D}_{il}v_{lj}^{(t)} \geq \lambda\mathbf{D}_{ii}v_{ij}^{(t)} \geq \lambda(\mathbf{D}-\mathbf{W})_{ii}v_{ij}^{(t)} = \lambda\mathbf{L}_{ii}v_{ij}^{(t)}$$

We finally prove

$$\frac{(\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U})_{ij} + \lambda(\mathbf{D}\mathbf{V})_{ij}}{v_{ij}^{(t)}} \geq (\mathbf{U}^{\mathbf{T}}\mathbf{U})_{jj} + \lambda\mathbf{L}_{ii}$$

In other words, $G(v,v_{ij}^{(t)}) \geq F_{ij}(v)$.

According to Lemma2, the update rule of $\mathbf{V}$ is:

$$v_{ij} \Leftarrow v_{ij}\frac{2[\mathbf{X}^{\mathbf{T}}\mathbf{U} + \lambda\mathbf{W}\mathbf{V}]_{ij} - \alpha}{2[\mathbf{V}\mathbf{U}^{\mathbf{T}}\mathbf{U} + \lambda\mathbf{D}\mathbf{V}]_{ij}}$$

In a similar way, we can also get the update rule of $\mathbf{U}$ and do regularization:

$$u_{ij} \Leftarrow u_{ij}\frac{(\mathbf{X}\mathbf{V})_{ij}}{(\mathbf{U}\mathbf{V}^{\mathbf{T}}\mathbf{V})_{ij}}$$

$$u_{ij} \Leftarrow \frac{u_{ij}}{\sum_{i}u_{ij}}$$