Assignment 1

Tutors: Yu Yao; Jazlyn Lin
Group members: Ke Wang 490538133 kwan7498; Hui-Yun Huang 520172904 hhua9634; Zijie Zhao 490576560 zzha9494

Abstract

Non-negative matrix factorization (NMF) has become a well-established means for big data analysis since it remedies some knotty problems and smartly produces essential and sparse features from a group of non-negative vectors. In this study, we first introduce the origin of NMF with some notable applications. Then we concisely bring out how an NMF model can be revised according to data distribution and distinct designs. Among these algorithms, we develop standard NMF and $L_{2,1}$-Norm Based NMF by reviewing their cost functions along with optimization steps. Experimental results for NMF implementations on two datasets are given to illustrate the robustness when facing contaminated data. Finally, we encapsulate the results and add some comments for future work.

1 Introduction

Unsupervised learning algorithms such as principal components analysis (PCA) can be considered as data matrix factorization with various constraints [1]. However, enforcing a weak orthogonality constraint cannot solve a variety of problems and shows its setbacks. In the natural world, a dataset is likely to be required as a non-negative matrix, such as the image intensities, document-term counts and stock market values. Facing such a non-negative dataset, applying nonnegativity is effective and useful. Non-negative matrix factorization (NMF) is a set of algorithms where a non-negative matrix is factorized into two non-negative matrices [2]. This non-negativity brings the simplicity of data inspection in many fields such as astronomy, document clustering, missing data imputation, chemometrics and audio signal processing [3]. Standard NMF was first proposed in 1999 with experiments on parts of faces and semantic features of text. It also addressed some problems in neural networks.

On the other hand, datasets are easier to be contaminated by large magnitude noise or corruption. NMF with higher robustness shows its advantages compared to other algorithms. $L_{2,1}$-Norm Based NMF was proposed in 2006 [4] which provides robustness to accommodate the widely existing noises and outliers than standard NMF. In this study, two NMF algorithms are implemented using the ORL dataset and the Extended YaleB dataset with intentional noise. The accuracy and robustness are also evaluated with the reconstructed dataset.

2 Related work

NMF is a special type of matrix factorization which satisfies the condition:

$$X \approx DR \quad s.t.\ D \geq 0, R \geq 0 \tag{1}$$

where $X \in \mathcal{X}, D \in \mathcal{D}, R \in \mathcal{R}$, while $\mathcal{X} = \mathbb{R}^{d \times n}, \mathcal{D} = \mathbb{R}_{+}^{d \times k}$ and $\mathcal{R} = \mathbb{R}_{+}^{k \times n}$. The goal is to find a new representation R and a basis D that could help reconstruct the input data X. If $k \ll n$, the

newly obtained representation R could be taken as another form of input with reduced dimension. However, the choice of k usually depends on the purpose of the applications.

NMF refers to a group of models that vary according to the variables (such as D, R in standard NMF), the loss function (such as the least squares error) and constraints (such as the orthogonality constraint in ONMF). In practice, the performance of different models depends on the task. Therefore, there is no certain model that is absolutely better than others. A good measure of the performance of NMF algorithms is using reconstruction error to approximate the difference between the original and reconstructed data. Generally, we wish to find a good reconstruction that is close enough to the original data, which means the reconstruction error should be as small as possible. The reconstruction error could be defined in numerous ways. Most NMF models that are commonly used, such as standard NMF, use the norms to define the reconstruction error. Based on the different definitions of error, the properties of the algorithm can be finely tuned and yield better results for some specific tasks.

2.1 Error and loss function

With different error measures, the loss function can be various. Given that the noise in a dataset is inevitable and usually unknown. A strategy is to assume the distribution of the noise. Suppose the noise follows normal distribution:

$$p\{y|x,h,\beta\} = \mathcal{N}(y|h(x),\beta^{-1}) \tag{2}$$

The log-likelihood can be written as:

$$-\ln p(S|X,h,\beta^{-1}) = -\frac{n}{2}\ln\beta + \frac{n}{2}\ln(2\pi) + \frac{\beta}{2}\sum_{i=1}^{n}\left(y_i - h(x_i)\right)^2 \tag{3}$$

which is equivalent to solving the loss function of standard NMF:

$$\min_{D,R}\|X - DR\|_F^2 = \min_{D,R}\sum_{i,j}(X - DR)_{ij}^2 \tag{4}$$

One of the notable advantages of the standard NMF is its high interpretability [5]. However, from the loss function shown above, the solution is vulnerable to large noise due to its squared format.

Similarly, if the noise follows the Laplace distribution, L_1-Norm NMF can be derived:

$$\min_{D,R}\|X - DR\|_1 = \min_{D,R}\sum_{i,j}|X - DR|_{ij} \tag{5}$$

L_1-Norm NMF is more robust than the standard NMF due to its removed squared loss function. However, this loss function is harder to solve than the standard one.

Moreover, some other variants can be derived:

- ℓ_∞-NMF [6] if the noise follows the Uniform distribution.
- KL-NMF [2] if the noise follows the Poisson distribution.
- IS-NMF [7] if the noise follows the multiplicative Gamma distribution.
- β-NMF [8] if the noise follows the Tweedie distribution.

The most critical drawback that cannot be neglected is the uniqueness: sometimes the NMF merely leads to local minima that can be diverse based on the initial values [9]. Therefore, comprehensive knowledge of applications and data domains is required for the implementation of a model.

2.2 Regularizations

Another way to revise models is by using additional regularizers. In recent years, a great number of regularizations in the literature have been proposed. The major purpose behind those regularizations

is to utilize prior information to refine the model and obtain better estimates. Basically, a constrained NMF model with regularizations can be written as:

$$\min_{D,R} \ell(X, D, R) + \alpha_D f_D(D) + \alpha_R f_R(R) \tag{6}$$

where $f_D(D)$ and $f_R(R)$ are regularizers or the penalty terms that enforce certain constraints, and α_D and α_R are positive parameters that balance the trade-off between the fitting and constraints.

By applying different regularizers, the model can achieve different features, such as sparsity, orthogonality, smoothness and minimum volume. Some widely used examples are as followed:

- ONMF [10]: $D \geq 0, R \geq 0, RR^T = I_r$
- semi-NMF [11]: $R \geq 0$
- sparse NMF [12]: $D \geq 0, R \geq 0$, D and/or R sparse
- projective NMF [13]: $D = XR^T, R \geq 0$
- symNMF [14]: $D = R^T \geq 0$

These specific structures can be a remedy to the issues shared by some NMF algorithms that cannot get a unique solution with only non-negativity constraint [15] as discussed before.

Nevertheless, selecting a proper model can be essential to the applications in practice. This requires a better knowledge of the domain. In addition, tuning the trade-off parameters can be empirical. Thus, a model may lose its interpretability and also brings some additional computational cost.

3 Methods

3.1 Algorithms

3.1.1 Standard (Frobenius-Norm based) NMF

Standard NMF [1] is the initial NMF whose reconstruction error is defined by the Frobenius-Norm, i.e., the squared Euclidean distance between the original data (input) and reconstructed data (output). As mentioned previously, reconstruction error is an important indicator that measures the quality of factorization results. One intuitive way to measure this error is by using the Frobenius-Norm, which is similar to the concept of L_2-Norm for vectors. More formally, assume A and B are two matrices with the same dimension, the Frobenius-Norm of the difference between them can be defined as:

$$\|A - B\|^2 = \sum_{ji} (A - B)_{ji}^2 \tag{7}$$

In standard NMF, A is the input data X, while B corresponds to the reconstruction that can be obtained by multiplying the basis D with the new representation R. As a result, given that the input data $X = (x_1, x_2, ..., x_n)$, the reconstruction error of standard NMF can then be expressed by:

$$\|X - DR\|_F^2 = \sum_{ji} (X - DR)_{ji}^2 = \sum_{i=1}^{n} \|x_i - Dr_i\|^2 \tag{8}$$

where x_i and r_i represent the i-th column in X and R, respectively. Since the goal of NMF is to find a good reconstruction of the original data, the above equation can be taken as the loss function of standard NMF as well. However, it should be noticed that Equation 8 is only convex with respect to either D or R but not to both of them. Therefore, it is not guaranteed to find the global minima when optimization methods are applied. Still, many approaches had been invented to help find local minima.

Gradient descent (or additive update rule) is one of the most well-known techniques among them. This method updates the variable by step in a direction opposite to the slope of the current location until the minima is found. Since we have two variables (D and R) to be solved, the alternative approach applied for NMF is to solve one variable while the others remain fixed. In this case, it is to first solve for D when R is fixed, then fix this newly updated D to solve for R. This process

would alternatively update both of them until the solution converges to the minima. However, such an updating process could take a long time. Even though there exist some other methods such as conjugate gradient that could give a faster convergence rate, it is hard to implement compared to gradient descent. Also, it is reported that gradient-based methods are sensitive to the choice of step size, which thus makes them hard to apply for large applications.

The gradient is obtained from the partial derivative of the target function 8 as

$$\frac{\partial \|X - DR\|_F^2}{\partial R} = (-2D^\top X + 2D^\top DR)_{i,j} \tag{9}$$

By setting the learning rate $\eta_{i,j}^m$ at the iteration m as

$$\eta_{i,j}^m = \frac{R_{i,j}^m}{(D^{mT} D^m R^m)_{i,j}} \tag{10}$$

the multiplicative update rule is a more elegant method which finds a balance between convergence rate and ease of implementation. It is derived as:

$$R_{ij}^{m+1} \Leftarrow R_{ij}^m \frac{(D^{m\top} X)_{ij}}{(D^{m\top} D^m R^m)_{ij}} \tag{11}$$

$$D_{ij}^{m+1} \Leftarrow D_{ij}^m \frac{(X R^{m+1\top})_{ij}}{(D^m R^{m+1} R^{m+1\top})_{ij}} \tag{12}$$

where R_{ij}^m and D_{ij}^m are the metrics R and D of iteration m respectively.

In this method, D and R are updated iteratively by equations 11 and 12. This multiplicative factor would be equal to unity once a perfect reconstruction, i.e., $X = DR$, is reached. Hence, it guarantees that the factorization results will not change anymore once a local minima is reached.

Standard NMF has a lot of admirable properties such as good interpretability and easy implementation as stated above. It had been shown that standard NMF can extract the features of human faces more effectively than the PCA [2]. However, one of the most critical weaknesses of standard NMF is its robustness [4]. Recall that in Equation 8, the error of each data point is expressed as the squared residual error in the form of $\|x_i - Dr_i\|^2$. As a result, outliers would tend to dominate the loss function due to this squared term. Many studies had proved that L_2-Norm is not robust to noises and may produce an unreliable result when noises exist in a dataset.

Another disadvantage of standard NMF is the convergence issue. As discussed above, standard NMF does not provide a unique solution since its loss function is not convex wholly. The final solution using the rule defined in Equation 11 and 12 may converge to a local minima rather than the desired global minima, which means a higher domain knowledge might be required for practical applications.

As discussed above, the objective function of standard NMF can be defined in terms of the Frobenius-Norm of reconstruction error by:

$$\min_{D,R} \|X - DR\|_F^2 \quad s.t.\ D \geq 0, R \geq 0 \tag{13}$$

Considering implementing the algorithm in a feasible time, a convergence criterion of the updating process was defined as well. The update on D and R described above would stop when the following condition is satisfied:

$$e_D < 10^{-5}\ and\ \ e_R < 10^{-5} \tag{14}$$

where e_D and e_R can be expressed in terms of the size of D and R by:

$$e_D^m = \frac{\|D^{m+1} - D^m\|_F^2}{|D^m|} \tag{15}$$

$$e_R^m = \frac{\|R^{m+1} - R^m\|_F^2}{|R^m|} \tag{16}$$

where $|\cdot|$ denotes the matrix size, and the superscript m and $m+1$ represent the iteration, which specifies the relationship between the updated and the original matrix. The above convergence criterion would be checked at the end of each iteration during the update. The iterative update process would be stopped if condition 14 has not been achieved but the iteration has reached the maximum criterion M_{max}.

3.1.2 $L_{2,1}$-Norm Based NMF

The concept of $L_{2,1}$-Norm (originally called rotational invariant L_1-Norm) was first proposed in 2006 [16] and defined as:

$$\|A\|_{2,1} = \sum_{i=1}^{n} \sqrt{\sum_{j=1}^{d} (A)_{ji}^2} = \sum_{i=1}^{n} \|a_i\| \tag{17}$$

where A is a $d \times n$ matrix. It can be interpreted as the summation of l_2-Norm of each row in A, i.e., the l_1-Norm of vector $(\|a_1\|, \|a_2\|, ..., \|a_n\|)$.

$L_{2,1}$-Norm Based NMF has many attractive advantages compared to standard NMF. First, it is robust to outliers and noises. Recall that in standard NMF, outliers or noises with large errors would tend to dominate the loss function due to the squared error term. However, $L_{2,1}$-Norm Based NMF has the ability to deal with such a problem. The loss function of $L_{2,1}$ NMF is defined as follows:

$$\|X - DR\|_{2,1} = \sum_{i=1}^{n} \sqrt{\sum_{j=1}^{d} (x - DR)_{ji}^2} = \sum_{i=1}^{n} \|x_i - Dr_i\| \tag{18}$$

It can be observed that the error of each data point is not squared; hence, the loss function is less likely to be influenced by the points with large errors, i.e., it is robust to outliers and noises. The robustness of $L_{2,1}$ NMF had been proved to make it outperform standard NMF in several machine learning tasks, including PCA, K-means clustering, and regression [4, 17]. Also, it had been shown that $L_{2,1}$ NMF could generate more reliable bases when applied to image processing [4]. In addition, many extensions such as $L_{2,1}$-Norm Based regularized NMF had been invented to improve the feature selection process with sparsity constraints in genomics and bioinformatics [18, 19].

However, finding an optimal solution for $L_{2,1}$ NMF could be much harder than standard NMF. Without an efficient algorithm, the computational cost for $L_{2,1}$ NMF would be significantly more expensive than standard NMF, which is not an obviously desirable property for practical implementations. However, this problem had been solved in 2011 when Deguang et al. [4] proposed a new update rule for $L_{2,1}$ NMF. It was proved that by following this update rule, the computational cost of $L_{2,1}$ NMF is almost the same as standard NMF.

Based on Equation 18, the objective function of $L_{2,1}$-Norm Based NMF can be defined as:

$$\min_{D,R} \|X - DR\|_{2,1} \quad s.t.\ D \geq 0, R \geq 0 \tag{19}$$

As discussed above, the function is not easy to solve compared with standard NMF, but an elegant updating rule, which is very similar to standard NMF, had been proposed by Deguang et al. [4] and can be expressed as:

$$R_{ij}^{m+1} \Leftarrow R_{ij}^{m} \frac{(D^{m\top} X Q^m)_{ij}}{(D^{m\top} D^m R^m Q^m)_{ij}} \tag{20}$$

$$D_{ij}^{m+1} \Leftarrow D_{ij}^{m} \frac{(X Q^m R^{m+1\top})_{ij}}{(D^m R^{m+1} Q^m R^{m+1\top})_{ij}} \tag{21}$$

where Q is a diagonal matrix with the diagonal elements given by:

$$Q_{ii}^m = \frac{1}{\sqrt{\sum_{j=1}^{d} (X - D^m R^m)_{ji}^2}} = \frac{1}{\|x_i - D^m r_i^m\|} \tag{22}$$

Apart from the form, the concept of this optimization method is also very similar to standard NMF. The value of D and R would be alternatively updated by the equations 20 and 21 until a convergence criterion is reached. The convergence criterion of $L_{2,1}$ NMF is the same as standard NMF which has already been defined in Equation 14.

3.2 Noises

In machine learning problems, robustness has always been a feature of great concern as it could have a significant impact on the performance of an algorithm. An algorithm is said to be robust if it is less likely to be affected by the outliers or large noises presented in the input data, i.e., its performance can still remain high-quality. Different types of noises may have different impacts on the robustness of the same algorithm. As a result, to investigate the robustness of the chosen NMF algorithms, two types of noises would be introduced to the datasets, and the implementation details would be elaborated in section 4.1.

The first type of noise is random additional noise which is a type of universal image noise presented through a random variation of brightness (for grayscale images) and colour information (for coloured images) in images. It is also frequently introduced in machine learning tasks for testing the robustness of an algorithm due to its simple implementation. Normally, it can be simulated by adding random integers that follow a specific distribution (e.g., Gaussian distribution) at each pixel of the image. Images with this type of noise could still display a recognizable outline of objects due to the random distribution of noises, while details of some important features may still be blurred.

The second type of noise is block-occlusion noise. Unlike random noise, block-occlusion noise is a kind of information loss that only presents in a certain local area of the image. It is generally displayed as a block that occludes the image content. Algorithms with an input contaminated by block-occlusion noise may have variant performances depending on the position and the size of the blocks. For instance, blocks that cover more important features or with a larger size may yield a model that fits the data poorly as the information contained in the basis is not enough to reconstruct the missing information within the images.

3.3 Evaluation metrics

To compare the performance and robustness of different NMF algorithms, three evaluation metrics, including relative reconstruction errors (RRE), average accuracy, and normalized mutual information (NMI) would be applied in this study.

3.3.1 Relative reconstruction errors (RRE)

Relative reconstruction error (RRE) is a measure of the difference between original and reconstructed data. Let X be the original input data, and D and R be the basis and new representation obtained from the NMF algorithm, it is defined as:

$$RRE = \frac{\|X - DR\|_F}{\|X\|_F} \tag{23}$$

RRE is a good indicator of the factorization result as a good reconstruction would yield a smaller value of RRE based on Equation 23. It is also suitable for the comparisons of different algorithms based on the fact that reconstruction error is normalized with respect to the input data X.

3.3.2 Average accuracy

The accuracy of NMF algorithms can be calculated when clustering is applied to the reconstructed data. The true labels would be defined by the original data with a unique label assigned to each example, while the predicted labels are the ones obtained from clustering. The average accuracy of a clustering result is formulated by:

$$Acc(Y, Y_{pred}) = \frac{1}{n} \sum_{i=1}^{n} 1\{Y_{pred}(i) == Y(i)\} \tag{24}$$

where Y, Y_{pred} denotes the true label and the predicted label, respectively, and n denotes the size of the input data. A higher accuracy indicates a better clustering performance, i.e., a better reconstruction from the factorization.

3.3.3 Normalized mutual information (NMI)

Normalized mutual information (NMI) is a metric that evaluates the quality of clustering from an entropy perspective [4]. More formally, it measures the entropy change in terms of the label assign-

ment between the ground truth and clustering result. Let $H(Y)$ and $H(Y_{pred})$ be the entropy of the true label and cluster label, respectively, the normalized mutual information can then be defined as:

$$NMI(Y, Y_{pred}) = \frac{2 * I(Y, Y_{pred})}{H(Y) + H(Y_{pred})} \tag{25}$$

where $I(Y, Y_{pred})$ is the mutual information of Y and Y_{pred} that can be calculated by:

$$I(Y, Y_{pred}) = H(Y) - H(Y|Y_{pred}) \tag{26}$$

Note that in Equation 25, mutual information is normalized with respect to the term $\frac{H(Y)+H(Y_{pred})}{2}$ which is the averaged entropy of the true label and predicted label. Similar to the average accuracy defined in Equation 24, a higher NMI indicates a better clustering result.

4 Experiment

4.1 Setup

The image datasets used in this study are the ORL Database of Faces [20] and the Extended Yale Face Database B [21]. The ORL dataset is a set of images that were taken at AT&T Laboratories and was used for face recognition projects. There are 400 images of 40 distinct subjects in total. Although some of the images were taken under a variety of conditions such as different brightness, varying time, dissimilar expressions, and facial details, the face in each image was all taken in an upright, frontal position against a dark background. Although the original size of each image is 92×112 pixels, to achieve a feasible computational complexity, images are resized to 30×37 pixels. Another is the Extended YaleB which contains 2414 frontal face images over 38 subjects with variant illumination and facial expressions. Each image is 168×192 pixels originally but is resized into 42×48 pixels for simple computation as well.

Apart from the different hurdles that are inherent in the original datasets, noises were introduced in this study to emulate the way by which a dataset is contaminated. First, the random noise was simulated by adding random positive integers which follow a uniform distribution between 0 and 40 to each pixel in an image. Examples are shown in Figures 1 and 2 for demonstrating their impacts on images. It can be observed that most features of human faces can still be recognized even after the noises were introduced.

On the other hand, the block-occlusion noise was simulated by setting the grayscale of a certain part of an image to 0. This process could completely replace the original information contained in that part and thus results in a block-like noise in the designated location while the rest remains unchanged. Considering the faces are both aligned without many variations, the location of the block in each image would be randomized using uniform distribution to prevent an absence of a particular feature. It is expected that the NMF algorithms could learn an information-rich basis and thus reconstruct the image more effectively under this setting. The results of the simulation of block-occlusion noise for each dataset are shown in Figures 3 and 4. By looking at the first two subjects in each dataset, it can be easily verified that the blocks generated by the aforementioned preprocessing step cover the random part of the images.

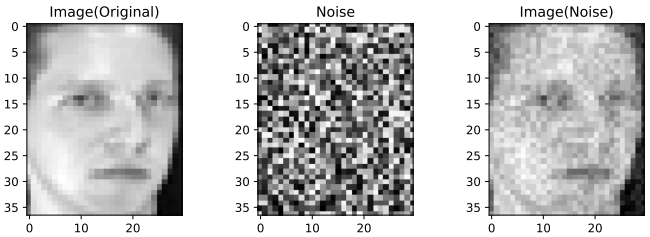

Figure 1: Examples of simulated random additional noises of the ORL dataset.

After all the preprocessing steps, two NMF algorithms were then applied to both clean and contaminated datasets to obtain the reconstruction results. Both D and R were initialized with random

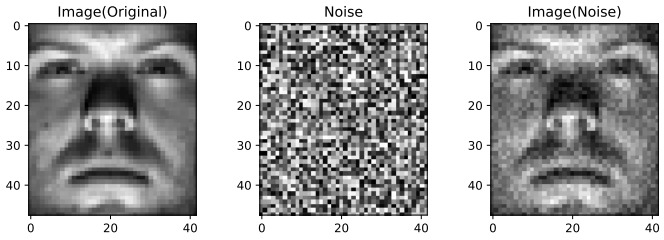

Figure 2: Examples of simulated random additional noises of the YaleB dataset.

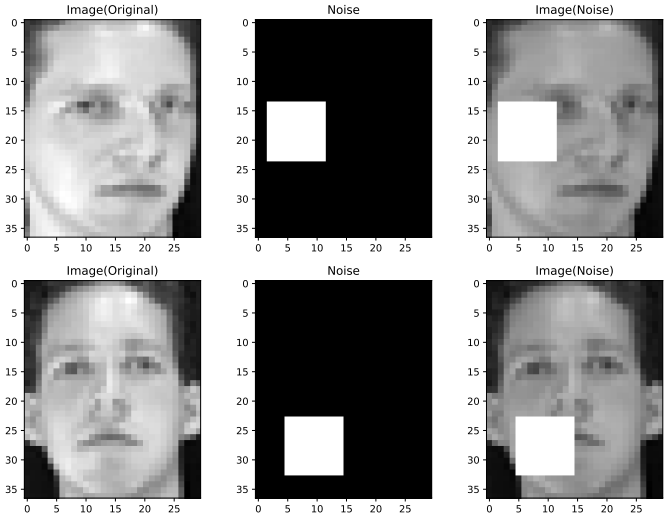

Figure 3: Examples of simulated random block-occlusion noises of the ORL dataset.

numbers ranging between 0 and 1. Firstly, the effect of dimension reduction which is described by the value of k in the algorithm performance would be first analyzed. An optimal k would be chosen based on some criteria and then set to constant for the following applications. The objective functions and the corresponding update rules that were used in this study have already been stated in section 3.1. The updating process would be terminated once the convergence criteria defined in 14 were reached, or the maximum iteration $M_{max} = 5000$ has been reached. Then, the metrics introduced in section 3.3 were applied to evaluate the quality and robustness of the reconstruction results. The RRE scores were calculated based on the definitions in Equation 23, while average accuracy and NMI were obtained by using the K-means clustering algorithm [22], which aims to partition the observations into K sets minimizing the within-cluster sum of squares.

$$\underset{S}{\operatorname{argmin}} \sum_{i=1}^{K} \sum_{x \in S_i} \|x - \mu_i\|^2 \tag{27}$$

where μ_i is the mean of the points in S_i.

The evaluation process would be repeated 100 times in order to estimate the performance of the algorithms by the evaluation metrics. The mean and the standard deviation out of the 100 samples of the metrics are reported in the following result and discussion section.

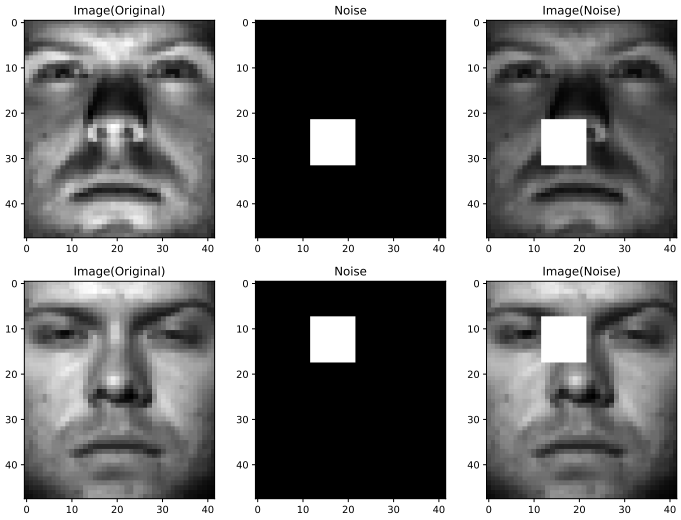

Figure 4: Examples of simulated random block-occlusion noises of the YaleB dataset.

4.2 Choice of k

In NMF algorithms, the extent of dimension reduction is a hyperparameter. NMF could be utilized as an approach to dimension reduction when the dimension of the new representation, which is denoted as k, is much smaller than the original data. However, the value of k would have a great impact on the reconstruction error of the input data. Normally, a larger k could result in a smaller reconstruction error, while a small k is desired for dimension reduction but usually with a larger reconstruction error. As a result, the value of k should be carefully chosen.

One intuitive way to determine k is based on the "elbow" of the variation of RRE, which is similar to the K decision in the K-means algorithm [22]. As shown in Figure 5, the changes of RRE were plotted under varying k. It can be observed that RRE usually decreases as the value of k increases, i.e., there is a trade-off between the extent of dimension reduction and reconstruction error. The increase of k gradually reduces the reconstruction error, as more information can be retained in the matrix D with a higher dimension. The aim is to determine the "optimal" k value when such decreasing in RRE is stable, which is the elbow of the RRE versus k graph. The k value is selected based on a decision boundary T_d, instead of a subjective human decision. The k value is iteratively increased from $k = 1$ until the difference in RRE of iteration $m + 1$ and m is less than T_d, i.e. $RRE^m - RRE^{m+1} < T_d$. The "optimal" k value is chosen as $k = m$. We set $T_d = 0.001$ and the optimal k values for the ORL and the YaleB dataset are found to be 27 and 29 respectively, by implementing the normal NMF algorithm. The two values of k are used for further evaluation and discussion of the NMF algorithms.

Steps are the number of update iterations taken to satisfy the convergence criterion defined in Equation 14. Unlike RRE which measures the error of factorization result, this parameter puts more emphasis on the efficiency of the NMF algorithm's convergence. Generally, when the latent features k is sufficiently large, the dimension reduction is easier to be achieved. A small k value close to 1 then makes the factorization impossible to retain the data utility. Therefore, as the results shown in Figure 5, higher k generally has a faster convergence rate.

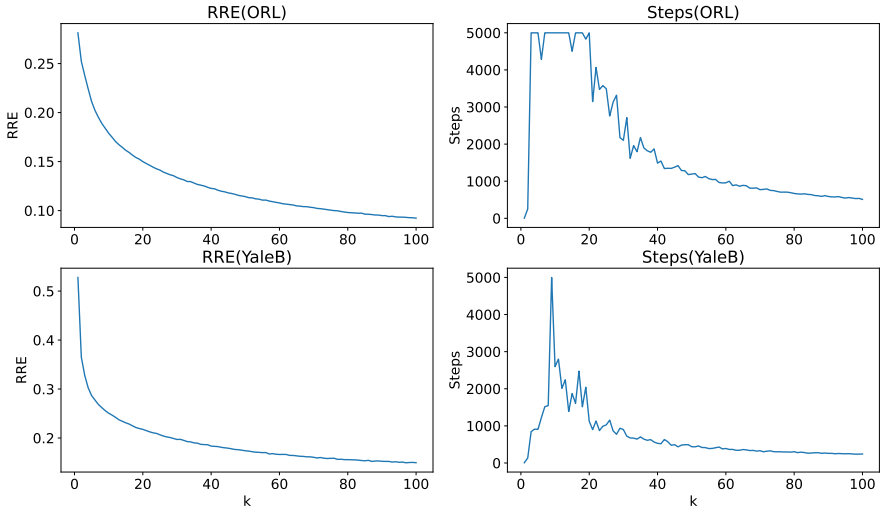

Figure 5: Standard NMF latent features k analysis.

4.3 Results and discussions

From Figures 6 and 7, we may first have a heuristic understanding that our NMF algorithms have successfully reconstructed the input image. The reconstructed images are similar to their inputs while keeping the noise pattern.

Further analysis of the robustness is based on the evaluation metrics, which are shown in Table 1. The averaged RREs of both algorithms show a good reconstruction achieved by them, as the relative difference between the reconstructed and original data is around 13.8% for the ORL dataset, while that of the YaleB dataset is around 20.0%. The standard deviations of both cases are very small, indicating a good consistency in dimension reduction with different initialization. Generally, according to all three evaluation metrics, the overall performance of standard (Frobenius-Norm) NMF and $L_{2,1}$-Norm Based NMF are very similar. The standard NMF has a slight advantage when faced with random additional noise, as RRE suggests an insignificant difference in performance but standard NMF has better performance considering accuracy and NMI. $L_{2,1}$-Norm Based NMF is more robust to block-occlusion noise, as the accuracy and NMI are increased by $3-6\%$ in the ORL dataset and only a slight performance degradation occurs in the YaleB dataset. Kong et al. [4] suggest a superior performance of $L_{2,1}$-Norm Based NMF compared with the Frobenius-Norm one. However, in our experimental setting, the optimal latent feature k is allocated by the application of the Frobenius-Norm NMF. The gap in the performance difference may be reduced under this hyper-parameter setting.

Both of the NMF approaches are more robust to the random additional noise compared with the block noise, under all three evaluation metrics. This is consistent with our expectation as adding random noise to the data affects the utility of the data less, because of the retained relative information of the local pixels. From the examples shown in Figures 1 and 2, the image after adding uniform noise can still be recognized by human eyes. However, the block-occlusion noise subtracts the information contained in the block, which makes the reconstruction more difficult. Especially, the accuracy and NMI suggest an improved performance after adding uniform noise compared with the original data. The noisy input without losing the universal information to the clustering algorithm, which is K-means clustering in our experiment, helps with the robustness of clustering and prevents over-fitting. Therefore, the resulting clustering performance may have a better generalized performance.

Besides, the computation time of the two NMFs is shown in Table 2. The computation time of the $L_{2,1}$ norm NMF is significantly longer than that of the normal one, which is consistent with our

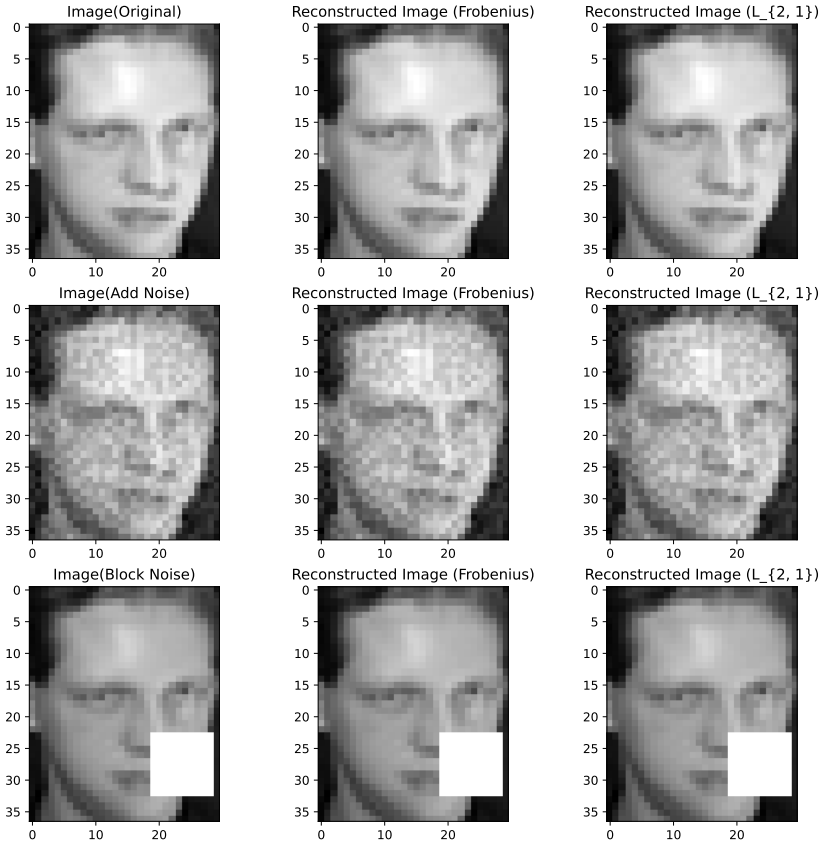

Figure 6: Examples of reconstructed image of the ORL dataset by the two NMFs.

expectation as the iterative update of the Q matrix makes the $L_{2,1}$ norm one more computational heavy.

5 Conclusion

We have implemented two NMF algorithms and analysed their performance using three metrics on two contaminated datasets. Both NMFs show robustness against noises, although the improvement might be limited by a different setting given our experimental conditions. It is worth noting that the $L_{2,1}$-Norm Based NMF is more robust compared to the standard one on block noise, while the computational cost has also increased. This observation is consistent with our research: a better model is determined by the particular application, which requires not only a deep knowledge of the data itself but also experimental understanding.

We are interested in different applications of these two NMFs, such as inpainting for text removal and RGB image restoration. These applications are meaningful and attractive due to their value in use. However, as mentioned before, no such good algorithm suits every task. Future development could also be extended to other algorithms which will deepen our acquaintance with NMFs.

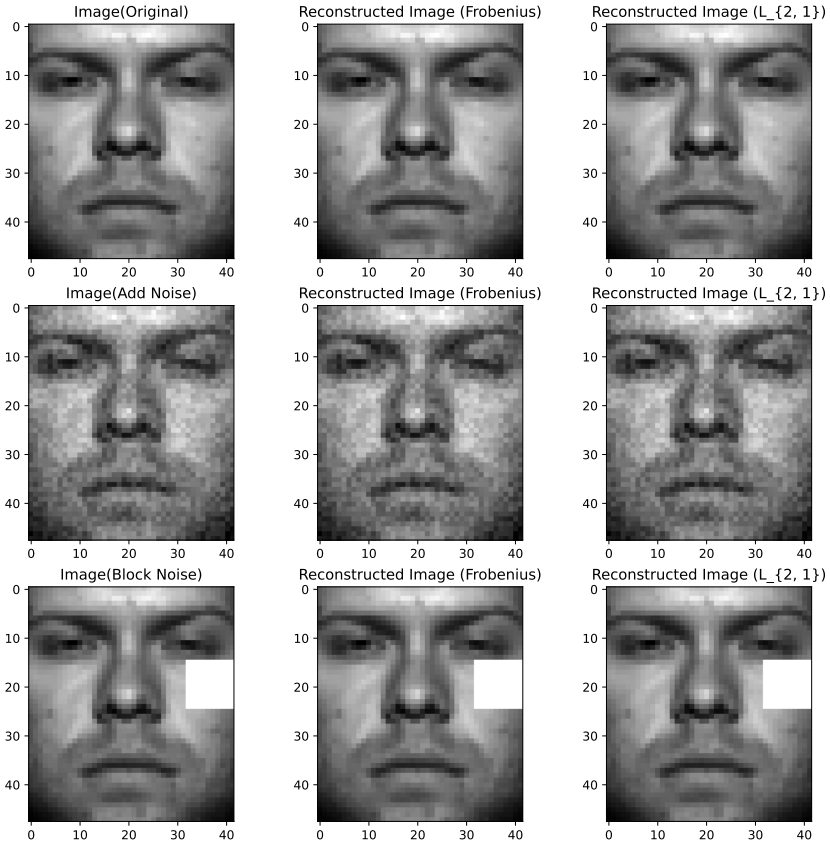

Figure 7: Examples of reconstructed image of the YaleB dataset by the two NMFs.

Table 1: Experimental results - robustness & performance.

	ORL (Original) Frobenius	**ORL (Add Noise) Frobenius**	**ORL (Block Noise) Frobenius**	**ORL (Original)** $L_{2,1}$	**ORL (Add Noise)** $L_{2,1}$	**ORL (Block Noise)** $L_{2,1}$
RRE (mean)	0.1380	0.2156	0.3258	0.1382	0.2154	0.3246
RRE (std)	0.0002	0.0002	0.0003	0.0002	0.0002	0.0003
Accuracy (mean)	0.7276	0.7317	0.2577	0.7188	0.7271	0.2739
Accuracy (std)	0.0230	0.0252	0.0176	0.0253	0.0234	0.0171
NMI (mean)	0.8467	0.8485	0.4231	0.8430	0.8469	0.4375
NMI (std)	0.0131	0.0142	0.0178	0.0146	0.0137	0.0181
	YaleB (Original) Frobenius	**YaleB (Add Noise) Frobenius**	**YaleB (Block Noise) Frobenius**	**YaleB (Original)** $L_{2,1}$	**YaleB (Add Noise)** $L_{2,1}$	**YaleB (Block Noise)** $L_{2,1}$
RRE (mean)	0.1996	0.2923	0.4464	0.1997	0.2922	0.4474
RRE (std)	0.0004	0.0003	0.0014	0.0005	0.0004	0.0012
Accuracy (mean)	0.2170	0.2207	0.1020	0.2118	0.2141	0.0955
Accuracy (std)	0.0096	0.0093	0.0059	0.0096	0.0101	0.0067
NMI (mean)	0.2963	0.3036	0.1135	0.2900	0.2962	0.1000
NMI (std)	0.0117	0.0136	0.0124	0.0148	0.0128	0.0122

Table 2: Experimental results - running time.

	ORL	**YaleB**
Frobenius	65m 2.3s	168m 2.9s
$L_{2,1}$	191m 57.9s	1064m 35.9s

*65m 2.3s refers to 65 minutes and 2.3 seconds, etc. The running time is for 100 times experiments under all the three evaluation metrics. When it's reduced to 5 times, the running time can be reduced to around 3 minutes to 53 minutes.

References

[1] Daniel Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13, 2000.

[2] Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[3] Wikipedia contributors. Non-negative matrix factorization — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Non-negative_matrix_factorization&oldid=1110427808`, 2022. [Online; accessed 5-October-2022].

[4] Deguang Kong, Chris Ding, and Heng Huang. Robust nonnegative matrix factorization using l21-norm. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 673–682, 2011.

[5] Russell Albright, James Cox, David Duling, Amy N Langville, and C Meyer. Algorithms, initializations, and convergence for the nonnegative matrix factorization. Technical report, Tech. rep. 919. NCSU Technical Report Math 81706. http://meyer. math. ncsu . . . , 2006.

[6] Nicolas Gillis and Yaroslav Shitov. Low-rank matrix approximation in the infinity norm. *Linear Algebra and its Applications*, 581:367–382, 2019.

[7] Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu. Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis. *Neural computation*, 21(3):793–830, 2009.

[8] Cédric Févotte and Jérôme Idier. Algorithms for nonnegative matrix factorization with the β-divergence. *Neural computation*, 23(9):2421–2456, 2011.

[9] Amy N Langville, Carl D Meyer, Russell Albright, James Cox, and David Duling. Initializations for the nonnegative matrix factorization. In *Proceedings of the twelfth ACM SIGKDD international conference on knowledge discovery and data mining*, pages 23–26. Citeseer, 2006.

[10] Megasthenis Asteris, Dimitris Papailiopoulos, and Alexandros G Dimakis. Orthogonal nmf through subspace exploration. *Advances in neural information processing systems*, 28, 2015.

[11] Chris HQ Ding, Tao Li, and Michael I Jordan. Convex and semi-nonnegative matrix factorizations. *IEEE transactions on pattern analysis and machine intelligence*, 32(1):45–55, 2008.

[12] Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, 5(9), 2004.

[13] Zhijian Yuan and Erkki Oja. Projective nonnegative matrix factorization for image compression and feature extraction. In *Scandinavian Conference on Image Analysis*, pages 333–342. Springer, 2005.

[14] Kejun Huang, Nicholas D Sidiropoulos, and Ananthram Swami. Non-negative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition. *IEEE Transactions on Signal Processing*, 62(1):211–224, 2013.

[15] Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on knowledge and data engineering*, 25(6):1336–1353, 2012.

[16] Chris Ding, Ding Zhou, Xiaofeng He, and Hongyuan Zha. R 1-pca: rotational invariant l 1-norm principal component analysis for robust subspace factorization. In *Proceedings of the 23rd international conference on Machine learning*, pages 281–288, 2006.

[17] Feiping Nie, Heng Huang, Xiao Cai, and Chris Ding. Efficient and robust feature selection via joint 2, 1-norms minimization. *Advances in neural information processing systems*, 23, 2010.

[18] Jun Liu, Shuiwang Ji, and Jieping Ye. Multi-task feature learning via efficient l2, 1-norm minimization. *arXiv preprint arXiv:1205.2631*, 2012.

[19] Dong Wang, Jin-Xing Liu, Ying-Lian Gao, Jiguo Yu, Chun-Hou Zheng, and Yong Xu. An nmf-l2, 1-norm constraint method for characteristic gene selection. *PloS one*, 11(7):e0158494, 2016.

[20] ATT Laboratories Cambridge. The database of faces.

[21] UCSD Computer Vision. Extended yale face database b.

[22] M A Syakur, B K Khotimah, E M S Rochman, and B D Satoto. Integration k-means clustering method and elbow method for identification of the best customer profile cluster. *IOP Conference Series: Materials Science and Engineering*, 336:012017, apr 2018.

A Contribution

Work is equally allocated to each member of the group. The coding part was done by Ke Wang who was also responsible for the presentation and analysis of the final results in the report. Zijie did a lot of research work and completed the report writing in abstract, introduction, related works, and conclusion. All the other parts that have not been mentioned, such as method and experimental setup, were finished by Hui-Yun who also participated in the discussion of the final results in collaboration with Ke Wang.