Assignment 1—Group:

Chen Chen (480458339), Xiaodan Gan(440581983), Joyce(12312312

17th September 2018

Abstract

Abstract text goes here, justified and in italics. The abstract would normally be one paragraph long.

Contents

1 Introduction

Non-negative matrix factorization (NMF) is a matrix decomposition technique that approximates a multivariate data matrix by two lower dimensional non-negative matrices as follows:

$$vv$$

As NMF only allows additive, non-subtractive combination of matrix factors, it is applicable to an extensive range of domain. Lee and Seung [1999] suggest that NMF is useful for image processing problems including facial recognition. Specifically, NMF generates two matrices W and H which are often referred as the basis images and weights. This is because the observed image V is approximated by a linear combination of W and H. This property also distinguishes NMF from other traditional image processing methods such as principal components analysis (PCA) and K-means clustering. Guillamet and Vitrià [2002] demonstrate that NMF is more robust to corrupted images than PCA. Moreover, NMF is also applicable to text mining such as semantic analysis. More generally, NMF is useful to discover semantic features of an article by counting the frequency of each word than approximating the document from a subset of a large array of features [Lee and Seung, 1999].

In practice, face images could be easily corrupted during data collection by large magnitude noise. Corruption may result from lighting environment, facial expression or facial details. An NMF algorithm that is robust to large noise is desired for real-world application. Therefore, the objective of this project is to analyse the robustness of NMF algorithms on corrupted dataset. We implement two NMF algorithms on real face image datasets, ORL dataset and Extended YaleB dataset. The face images are contaminated by artificial noises.

Plan: ...

2 Related work

Researchers proposed many NMF algorithms. Lee and Seung [2001] first proposes to algorithms which minimises Euclidean distance or Kullback-Leibler divergence between the original matrix and its approximation. Although this algorithm is easy to implement and have reasonable convergent rate [Lee and Seung, 2001], it may fail on seriously corrupted dataset which violates its assumption of Gaussian noise or Poisson noise, respectively [Guan et al., 2017]. To improve the robustness of NMF, many methods have been proposed. Lam [2008] proposes L_1-norm based NMF to model noisy data by a Laplace distribution which is less sensitive to outliers. However, as L_1-norm is not differentiable at zero, the optimization procedure is computationally expensive. Kong et al. [2011] proposed an NMF algorithm using L_{21}-norm loss function which is robust to outliers. The updating rules used in L_{21}-norm NMF, however, converge slowly because of a continual use of the power method [Guan et al., 2017].

Apart from different loss functions, several optimization methods were proposed to improve the performance of NMF. After Lee and Seung [2001] proposed multiplicative update rules MUR, Lin [2007] proposed a modified MUR which guaranteed the convergence to a stationary point. This modified MUR, however, did not improve the convergence rate of traditional MUR [Guan et al., 2012]. Moreover, as MUR is not able to shrink all entries in matrix factors to zero, Berry et al. [2007] proposed a projected nonnegative least square (PNLS) method to overcome this problem. Although, in each nonnegative least square (NLS) subproblem, the least squares solution is directly projected to the negative quadratic, PNLS does not guarantee convergence [Guan et al., 2012]. In contrast to these gradient-based optimization methods, Kim and Park [2008] presented an active set method AS which divides variables into two sets, free set and active set and update free set in each iteration by solving unconstrained equation. Even though AS has good convergence rate, it assumes strictly convex in each NLS subproblem [Kim and Park, 2008].

3 Methods

3.1 NMF and Gaussian noise

Lee and Seung [2001] propose the first NMF with the objective function between imaga V and its NMF factorisation W and H being

$$\|V - WH\| = \sum_{ij} \left[V_{ij} - (WH)_{ij}\right]^2.$$

To minimise this object function of least square, Lee and Seung [2001] prove the convergence of the multiplication update rule

$$H_{jk} \leftarrow H_{jk} \frac{(W^T V)_{jk}}{(W^T WH)_{jk}} \text{ and } W_{ij} \leftarrow W_{ij} \frac{(VH)_{ij}}{(WHH^T)_{ij}}.$$

Here, $()_{ij}/()_{ij}$ denotes elementwise division of the two matrix. Liu and Tao [2016] shows this NMF algorithm minimises Gaussian.

3.2 KLNMF and Poisson noise

Lee and Seung [2001] suggest that KLNMF is a algorithm that minimising the Kullback-Leibler divergence

$$\begin{aligned} D(V\|WH) &= \sum_{ij} \left(V_{ij} \log \frac{V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij} \right) \\ &= \sum_{ij} \left(-V_{ij} \log (WH)_{ij} + (WH)_{ij} + C(V_{ij}) \right). \end{aligned} \tag{1}$$

where $C(V_{ij}) = V_{ij} \log V_{ij} - V_{ij}$. $C(V_{ij})$ is a function of the observed image matrix V only. Lee and Seung [2001] also suggest a multiplication update rule to find as the optimisation procedure of KLNMF

$$H_{jk} \leftarrow H_{jk} \frac{\sum_i W_{ij} V_{ik}/(WH)_{jk}}{\sum_{i'} W_{i'j}} \text{ and } W_{ij} \leftarrow W_{ij} \frac{\sum_k H_{jk} V_{ik}/(WH)_{jk}}{\sum_{k'} H_{ik'}}.$$

As this original image matrix V is observed, minimising this Kullback-Leibler divergence (1) is equivalent to minimising

$$\sum_{ij} \left(-V_{ij} \log (WH)_{ij} + (WH)_{ij} + C(V_{ij}) \right).$$

, for arbitrary bounded function $C(V_{ij})$. Taking exponential of the negative of this score function, the problem transforms to maximising the following likelihood function

$$L(WH|V) = \prod_{ij} \left((WH)_{ij}^{V_{ij}} e^{-(WH)_{ij}} + C(V_{ij}) \right).$$

Choosing constant $C(V_{ij})$ to be $-\log V_{ij}!$ gives

$$L(WH|V) = \prod_{ij} \left(\frac{(WH)_{ij}^{V_{ij}} e^{-(WH)_{ij}}}{V_{ij}!} \right).$$

Hence, the probability density function of each element of the original matrix V is Poisson

$$P(V_{ij}) = \frac{(WH)_{ij}^{V_{ij}} e^{-(WH)_{ij}}}{V_{ij}!}$$

is a sufficient condition to yield this likelihood. Hence KLNMF is most suitable for images with Poisson noise.

3.3 Asymptotic equivalence of noise distributions

We design an Gaussian noise and a Poisson noise with different magnitude. Poisson distribution with parameter λ (integer) is equivalent to the sum of λ Poisson

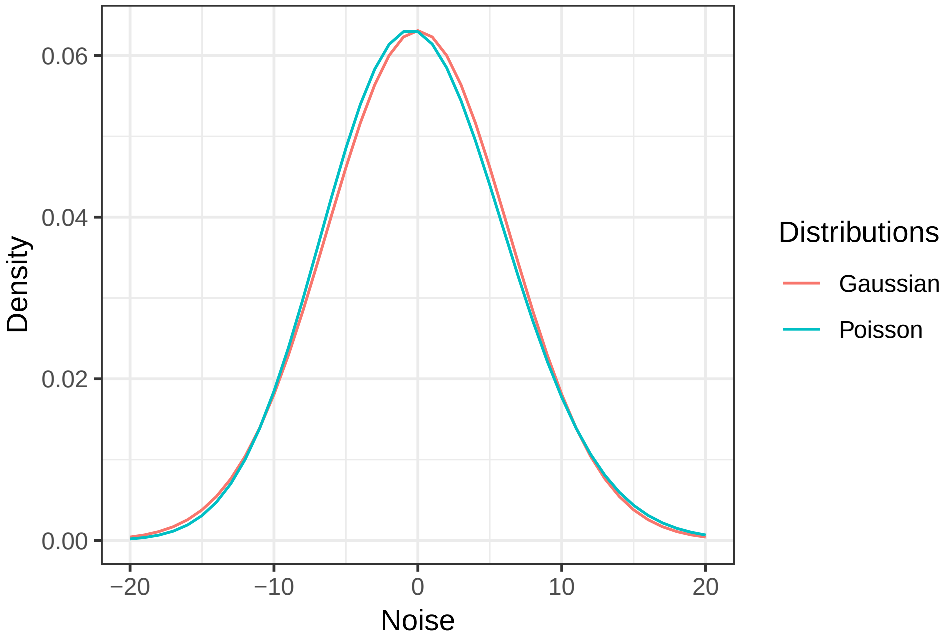

Figure 1: Compare a Gaussian noise $N(0,40)$ with Poisson noise $\text{Poi}(40)-40$. They two distributions are asymptotically equivalent and have similar density functions.

distributions with parameter 1 [Walck, 1996, p. 45]. Hence for λ large, Central Limit Theorem implies that Poisson distribution with parameter λ is well approximated by $N(\lambda,\lambda)$. When applying Poisson noise to an image, we do not degree of freedom to choose any parameter. The variance is the magnitude of the pixels. To compare the robustness of KLNMF with NMF with different noise, we choose the variance of Gaussian noise to be the different from the magnitude of the pixel, that is, $N(0,\text{Var}) \neq N(0,V) \approx \text{Poi}(V)-V$. Figure 1 visualises the similarity of Poisson distribution and Normal distribution with parameter $V=40$.

	rre	acc
Multiplication Euclidean Normal	0.15683265883959555	0.728611111111
Multiplication Euclidean Poisson	0.1591413186921236	0.72499999999
Multiplication KL Divergence Normal	0.15703871615765036	0.728333333333
Multiplication KL Divergence Poisson	0.1589874246949232	0.742499999999

3.4 Preprocessing

We apply global centring and local centring to preprocess the image data `Vhat`

Algorithm 1: Centring image data

```
n_samples = len(Vhat)
# global centering
Vhat = Vhat - Vhat.mean(axis=0)
# local centering
Vhat -= Vhat.mean(axis=1).reshape(n_samples, -1)
Vhat -= Vhat.min()
```

4 Experiments

123

5 Conclusion

Your conclusion goes at the end, followed by References, which must follow the Vancouver Style (see: www.icmje.org/index.html). References begin below with a header that is centered. Only the first word of an article title is capitalized in the References.

A Appendix

Algorithm 2: The Levenberg–Marquardt algorithm iteratively finds optimal macro-scale Robin boundary conditions.

```
%A=imread('C:\Users\chenc\OneDrive - UNSW\machine learning\assignment 1\data\CroppedYaleB\yaleB01\yaleB01_P00A+000E+00.pgm');
red=3;
k=40;

imagefiles = dir('data/ORL/*/*.pgm');
imagefiles2=struct2cell(imagefiles);
imagefiles=imagefiles((~endsWith(imagefiles2(1,:),'Ambient.pgm'))');
imagefiles2=struct2cell(imagefiles);
A=imread(strcat(imagefiles(1).folder,'\',imagefiles(1).name));
if size(A,1)==112
    A=A(1:111,1:90);
end
A_list=zeros(size(A,1)/red,size(A,2)/red,red);
nfiles = length(imagefiles);    % Number of files found
matrix_image=zeros(prod(size(A))/red^2,nfiles);
temp=struct2cell(imagefiles);
names=temp(1,:)
for ii=1:nfiles
   currentfilename = imagefiles(ii).name;
   currentfilename=strcat(imagefiles(ii).folder,'\',currentfilename)
   currentimage = imread(currentfilename);
   if abs(size(A,1)-112)<=1
    currentimage=currentimage(1:111,1:90);
   end

   for i=1:red
    A_list(:,:,i)=currentimage(i:red:end,i:red:end);
   end
   A2=uint8(mean(A_list,3));
   matrix_image(:,ii) = A2(:);
end
```

```
[w h]=NeNMF(matrix_image,k);
idx = kmeans(h',k)
Y_pred=zeros(size(matrix_image,2),1)
namess=str2mat(string(imagefiles2(2,:))')
namess=str2num(namess(:,end-1:end))
for ii=unique(idx)'
   ind= (idx==ii);
   Y_pred(ind)=mode(namess(ind,:));
end
nmi(Y_pred,namess)
```

References

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

David Guillamet and Jordi Vitrià. Non-negative matrix factorization for face recognition. In *Topics in artificial intelligence*, pages 336–344. Springer, 2002.

Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 556–562. MIT Press, 2001. URL http://papers.nips.cc/paper/1861-algorithms-for-non-negative-matrix-factorization.pdf.

Naiyang Guan, Tongliang Liu, Yangmuzi Zhang, Dacheng Tao, and Larry Steven Davis. Truncated cauchy non-negative matrix factorization for robust subspace learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

Edmund Y Lam. Non-negative matrix factorization for images with laplacian noise. In *Circuits and Systems, 2008. APCCAS 2008. IEEE Asia Pacific Conference on*, pages 798–801. IEEE, 2008.

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. ACM, 2011.

Chih-Jen Lin. On the convergence of multiplicative update algorithms for nonnegative matrix factorization. *IEEE Transactions on Neural Networks*, 18(6): 1589–1596, 2007.

Naiyang Guan, Dacheng Tao, Zhigang Luo, and Bo Yuan. Nenmf: An optimal gradient method for nonnegative matrix factorization. *IEEE Transactions on Signal Processing*, 60(6):2882–2898, 2012.

Michael W Berry, Murray Browne, Amy N Langville, V Paul Pauca, and Robert J Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. *Computational statistics & data analysis*, 52(1):155–173, 2007.

Hyunsoo Kim and Haesun Park. Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method. *SIAM journal on matrix analysis and applications*, 30(2):713–730, 2008.

Tongliang Liu and Dacheng Tao. On the performance of manhattan nonnegative matrix factorization. *IEEE Transactions on Neural Networks and Learning Systems*, 27(9):1851–1863, September 2016. doi:10.1109/TNNLS.2015.2458986.

Christian Walck. *Hand-book on statistical distributions for experimentalists*. 1996. URL `http://www.fysik.su.se/~walck/suf9601.pdf`.