Assignment 1—Group:

Chen Chen (123123123), Dania(123123123), Joyce(123123123)

9th September 2018

Abstract

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

Contents

1 Introduction

This template should be used as a starting point for your report. Guan et al. [2012]

2 Methods

2.1 KLNMF

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

$$D(V||WH) = \sum_{ij} \left(V_{ij} \log \frac{V_{ij}}{(WH)_{ij}} - V_{ij} + (WH)_{ij} \right).$$

Define $C(V_{ij})$ to be arbitrary function of the observed matrix only. As this original matrix V is observed, minimising this Kullback-Leibler divergence is equivalent to minimising

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

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.

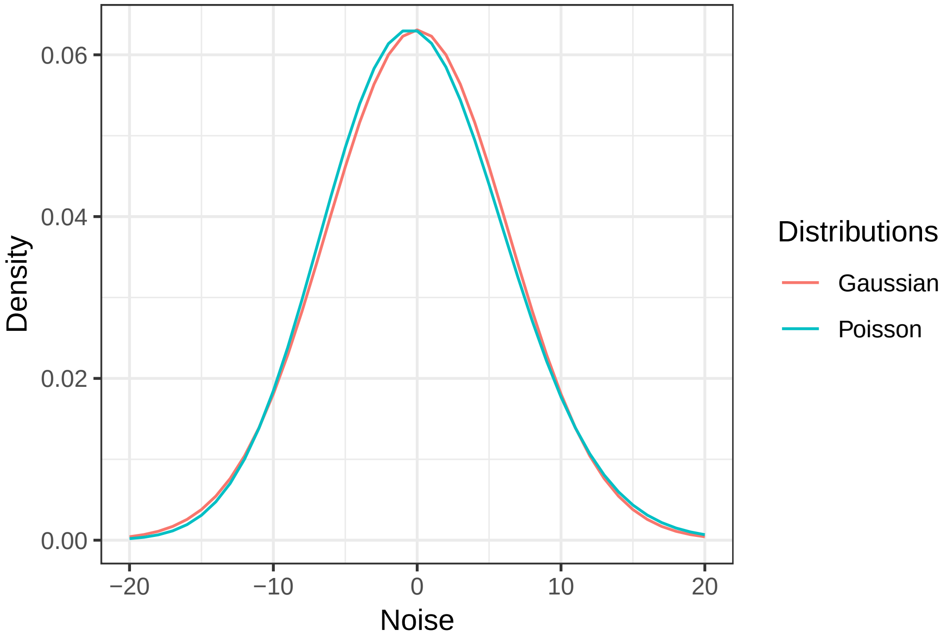

Figure 1: Compare a Gaussian noise $N(0, 40)$ with Poisson noise $\text{Poi}(40) - 40$

2.2 Asymptotic equivalence of noise distributions

We design an Gaussian noise and a Poisson noise with similar magnitude. Poisson distribution with parameter λ (integer) is equivalent to the sum of λ Poisson distributions with parameter 1 (reference a book!!!!!!!!!!here). 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 fairly compare the robustness of KLNMF with NMF, we choose the variance of Gaussian noise to be the magnitude of the pixel, that is, $N(0, V) \approx \text{Poi}(V) - V$. Figure 1 visualises the this derivation with $V = 40$.

3 Experiments

123

4 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 1: The Levenberg–Marquardt algorithm iteratively finds optimal macroscale 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

N. Guan, D. Tao, Z. Luo, and B. Yuan. Nenmf: An optimal gradient method for nonnegative matrix factorization. *IEEE Transactions on Signal Processing*, 60(6):2882–2898, June 2012. ISSN 1053-587X. doi:10.1109/TSP.2012.2190406.

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.