CH5540 Multivariate Data Analysis - Assignment 4

AE14B050

April 29, 2018

Introduction

For the purpose of this assignment we create two functions in matlab namely OLSNCA.m and fastNCA.m . The former is the basic rational matrix solving method which incorporates OLS for solving multiple equation cases. while the later uses the fastNCA algorithm to do NCA. Below are both the functions.

Listing 1: Ordinary OLS based NCA

```
function [A P M] = OLSNCA(Z, As)

shape = size(As); %get shape of the connectivity matrix structure
species = shape(2); %determine species from shape
[u s v]=svd(Z,'econ'); %economy SVD of the mixture absorbance/microarray
   data matrix Z
Ahat = u(:,1:species); %denoise by taking only the first n_s columns of u as
    Ahat
M = zeros(species); %Create M matrix for speed
insert1 = @(x, n)cat(2,  x(1:n-1), 1, x(n:end)); %helper function to insert
   1
for i=1:species %iterated over the columns of Astruct
    dum=[];
    for j=1:shape(1) %get rows from Ahat which have zeros in Astruct
        if(As(j,i)==0)
            dum = [dum;Ahat(j,:)];
        end
    end
    LHS = dum;
    LHS(:,i)=[]; %remove the column from dum that corresponds to the 1
       diagnol in M. essentially finding A in AX = b.
```

```
    RHS = -dum(:,i); % set this column as -b in AX=b system
    X = inv(LHS.'*LHS)*(LHS.'*RHS); %OLS solution
    X = insert1(X.',i); %adding 1 in appropriate place
    M(:,i)= X.'; %setting the ith column of M
end
A = Ahat*M; %rotating with scale ambiguity to get A
P = inv(A.'*A)*(A.'*Z);
```

Listing 2: fastNCA implementation

```
function [A P] = fastNCA(Z, As)

shape = size(As);%get shape of the connectivity matrix structure
species = shape(2);%determine species from shape
[u s v]=svd(Z,'econ'); %economy SVD of the mixture absorbance/microarray
    data matrix Z
A = As;%create A to fill in later
Ws = u(:,1:species); %denoised Ws
[val id] = sort(As,'descend'); %sorting to determine which elements in each
    column have zeros and which do not
nonz = sum(As); %number of nonzero elements in each column of the Astruct

%as per fast NCA algorithm given in the question
for i = 1:shape(2);
     Wc = Ws(id(1:nonz(i),i),:);
     Wr = Ws(id(nonz(i)+1:end,i),:);
     Ar = As(id(nonz(i)+1:end,i),2:end);
     [ur sr vr] = svd(Wr);
     S = vr(:,end);
     WcS = Wc*S;
     [uf sf vf] = svd(WcS);
     a1 = uf(:,1);
     A(id(1:nonz(i),i),i) = a1;
end
P = inv(A.'*A)*(A.'*Z);
```

Solution 1:

(a) NCA compliance: From the given network we shall create a structure for the connectivity matrix,

$$C_{struct} = \begin{bmatrix} x & x & 0 \\ x & 0 & x \\ 0 & x & x \\ x & 0 & x \\ x & x & 0 \\ x & 0 & x \\ 0 & x & x \end{bmatrix}$$

Here the unknown values are denoted by x while the known disconnection is given by 0s.

- The given structure has [2,3,2] zeros in each column which conforms to the basic rules of compliance where at least $(n_s - 1)$ zeros are required in each column. Here $n_s = 3$ hence this criterion is satisfied.
- The given structure is also full column ranked as no two columns have the same structure.

The above two conditions are enough to establish NCA compliance and hence the given structure is NCA compliant.

(b) OLS-NCA: As mentioned before we use the above OLSNCA function created to find all the required matrices. Denoising of the data is also taken care of the algorithm internally. Following is the rotation matrix,

$$M = \begin{bmatrix} 1 & 2.5966 & 9.3257 \\ 0.2702 & 1 & -11.7212 \\ 1.5566 & -0.2113 & 1 \end{bmatrix}$$

The correlation matrix between pure absorbance spectra and estimated pure absorbance spectra is obtained as,

$$Corr = \begin{bmatrix} -0.9425 & 0.4291 & 0.4343 \\ 0.3662 & -0.9968 & -0.3815 \\ 0.4940 & -0.4074 & -0.9967 \end{bmatrix}$$

Negative values just mean a negative scaling of the spectra but are essentially highly correlated. We hence observe that the estimated absorbance spectrum conforms to the pure absorbance spectrum.

Listing 3: Matlab code for solution 1

```
clc
```

```
clear all
load ncadata
Ctrue = [1,1,0;1,0,1;0,1,1;1,0,1;1,1,0;1,0,1;0,1,1];
S = pureabs;
A = measabs;
%[Cstar Sstar M]= OLSNCA(A, Ctrue);  %OLS NCA
[Cstar Sstar]= fastNCA(A, Ctrue);  %fast NCA use as required
cor = corr(S.',Sstar.'); %correlation between true and estimated pure
    absorbance spectrums
```

Solution 4(1c):

fastNCA: We now use the fastNCA function created. Denoising of the data is also taken care of in the algorithm internally. The correlation matrix between pure absorbance spectra and estimated pure absorbance spectra is obtained as,

$$Corr = \begin{bmatrix} -0.9425 & -0.4291 & -0.4343 \\ 0.3661 & 0.9968 & 0.3815 \\ 0.4940 & 0.4074 & 0.9967 \end{bmatrix}$$

Negative values just mean a negative scaling of the spectra but are essentially highly correlated. We hence observe that the estimated absorbance spectrum conforms to the pure absorbance spectrum.

Both the OLSNCA and fastNCA give the same results here as the difference between the number of variables and equations is just 1 (7 equations or zeros but only 6 variables). If the number of equations are much larger, the correlation matrices in both cases will be comparatively different.

Solutions 2:

Using NCA as before we obtain the following list of temporal expression levels ordered from highest to lowest. As we observe we are able to identify most of the TFs as given in the question with just 1 error (YAP6) in the first 11 major TFs. The missing STB1 is found close by at 14 .

1-5: $['ACE2','SWI4','MBP1','NDD1','FKH2']$

6-10:$['SWI6','SKN7','MCM1','SWI5','FKH1']$

11-15:$['YAP6','PHD1','STE12','STB1','RAP1']$

16-20:$['PHO4','ABF1','HSF1','RLM1','YAP1']$

21-25:['SMP1','CIN5','NRG1','RFX1','RME1']

26-30:['HIR2','REB1','MSN4','FHL1','DIG1']

31-33:['GRF10(Pho2)','HIR1','GCN4']

Listing 4: Solution 2

```
clc
clear all
load yeastdata


Cs = Astruct;
MAD = microarraydata;
[Cstar Sstar] = OLSNCA(MAD,Cs);
Cstar = normc(Cstar);%normalize connectivity matrix
Cstar(Cs==0)=0;% set zero exressions from known structure
expr = var(Cstar); %variance of Cstar
[val id] = sort(expr,'descend'); %sort to variance
maxe = tfa(id); %ids of the TFs corresponding to maximum variance
```

Solutions 3:

(a) First samples: We use the following matlab code for both (a) and (b). nmf. mat has been used to find implement the nmf algorithm. The initialized variables are given as guided by the question. The following is the correlation matrix in tabulated form.

Est↓\True→	Ni	Cr	Co
Unknown1	0.8844	0.5887	-0.5728
Unknown2	0.2879	0.8324	-0.1036
Unknown3	-0.0296	-0.5106	0.5370

From the correlation values it very plain to observe that the Unknown1 is most probably Ni while Unknown2 is Cr with correlation coefficients greater than 0.8 . Unknown 3 is most correlated with Co but also closely correlated with Cr.

Est	Connected True
Unknown1	Ni
Unknown2	Cr
Unknown3	Co

We can say Both Ni and Cr are being extracted very well while Co is very poorly extracted from the data.

(b) Averaged samples: The above procedure is again followed and we tabulate our correlation matrix as follows,

Est↓\True→	Ni	Cr	Co
Unknown1	0.8947	0.5406	-0.4725
Unknown2	-0.2481	-0.4381	0.8728
Unknown3	0.4132	0.9603	-0.4456

From the correlation values we observe that the Unknown1 is most probably Ni, Unknown2 is Co and Unknown3 is Cr with correlation coefficients greater than 0.8 . The following is the corresponding table showing which Unknown is correlated to which pure component spectra.

Est	Connected True
Unknown1	Ni
Unknown2	Co
Unknown3	Cr

There is a huge improvement in that the correlation coeeficients are much larger and it is very easy to distinguish using the averaged data.

Listing 5: Matlab code for solution 3

```
clc
clear all
load Inorfull

A = DATA;
C = CONC;
S = [;PureNi/PureNiCONC;PureCr/PureCrCONC;PureCo/PureCoCONC;]; %pure
    absorbances scaled to unitary concentrations
shape =size(C);
species = shape(2);

fsamples= 1:5:size(A,1);
A1 = A(fsamples,:);%matrix with first samples
A1(A1<0)=0;%setting negatives to zero

Aavg = A1; %creating the averaged data matrix with A1 structure
for i = fsamples
    Aavg((i+4)/5,:) = mean(A(i:i+4,:));
end
Aavg(Aavg<0)=0;
```

```
[u s v] = svd(A1); %pca using first samples
Sinit = v(:,1:species).'; %loadings
Cinit = A1*Sinit.'; %scores
Sinit = abs(Sinit); %absolute values
Cinit = abs(Cinit);

[Cest,Sest] = nmf(A1,Cinit,Sinit,1e-12,100,10000); %nmf
cor1 = corr(Sest.',S.'); %correlation of true and estimated pure absorbance
    spectrums

[u s v] = svd(Aavg); %pca using averaged data, use and comment as required
Sinit = v(:,1:species).'; %loadings
Cinit = A1*Sinit.'; %scores
Sinit = abs(Sinit); %absolute values
Cinit = abs(Cinit);

[Cest,Sest] = nmf(Aavg,Cinit,Sinit,1e-12,100,10000); %nmf
cor = corr(Sest.',S.'); %correlation of true and estimated pure absorbance
    spectrums
```