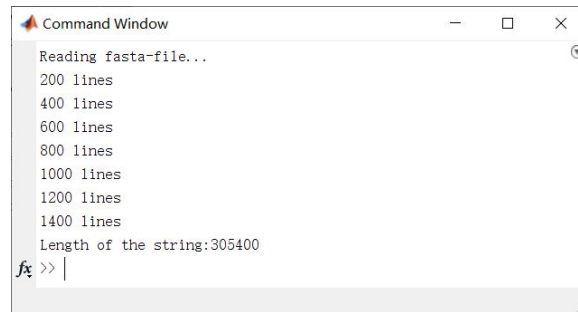


# MATH444 HW6

Jiasen Zhang

## 1 Problem 1

By typing the given Matlab code, I get a string whose length is 305400. It's a fragment of the genomic sequence and contains 4 kinds of nucleotides or codons, represented by a, c, g and t.



```
Command Window
Reading fasta-file...
200 lines
400 lines
600 lines
800 lines
1000 lines
1200 lines
1400 lines
Length of the string:305400
fx >> |
```

Figure 1: Output in command window.

## 2 Problem 2

When  $n = 3$ , the words consist of three letters and there are  $4^3 = 64$  words in the dictionary.

Now I visualize the generated term-document matrix with  $n = 1, 2, \dots, 6$ . I plot the first two principal components of the centered data. As shown in Figure 2, when  $n = 1, 2, 4$ , the data points form only one cluster and they are not separated. When  $n = 5$ , there are about 3 clusters but the clusters stay really close. When  $n = 3, 6$ , there are about 6 clusters, and the clusters are still close to each other when  $n = 6$ . To sum up, when  $n = 3$ , the data points are separated to the best extent into 6 clusters.

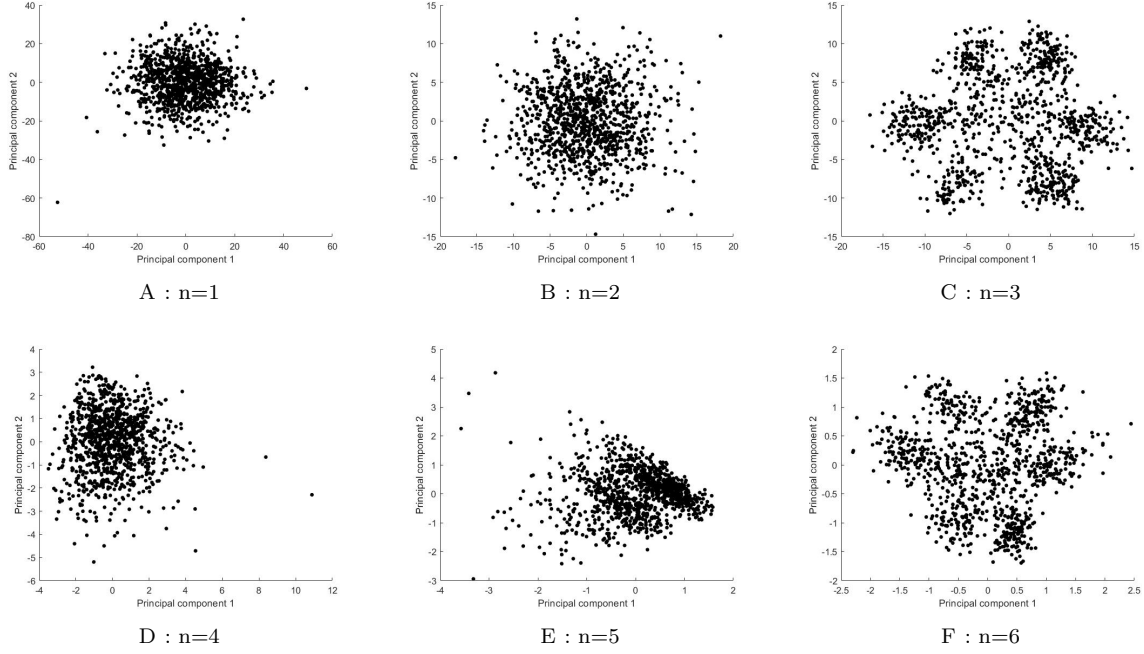


Figure 2: The first two principal components when  $n = 1, 2, \dots, 6$  and  $k = 300$ .

To test the effect of  $k$ , the length of the document, I set  $n = 3$  and try  $k = 100, 200, 300, 400, 500, 600$ . The results are shown in Figure 3. It seems that the results remain the same with  $k$  changing, the data points are separated in the same way. However, when  $k$  is small, there will be more documents and more data points, which make it more difficult to separate the data.

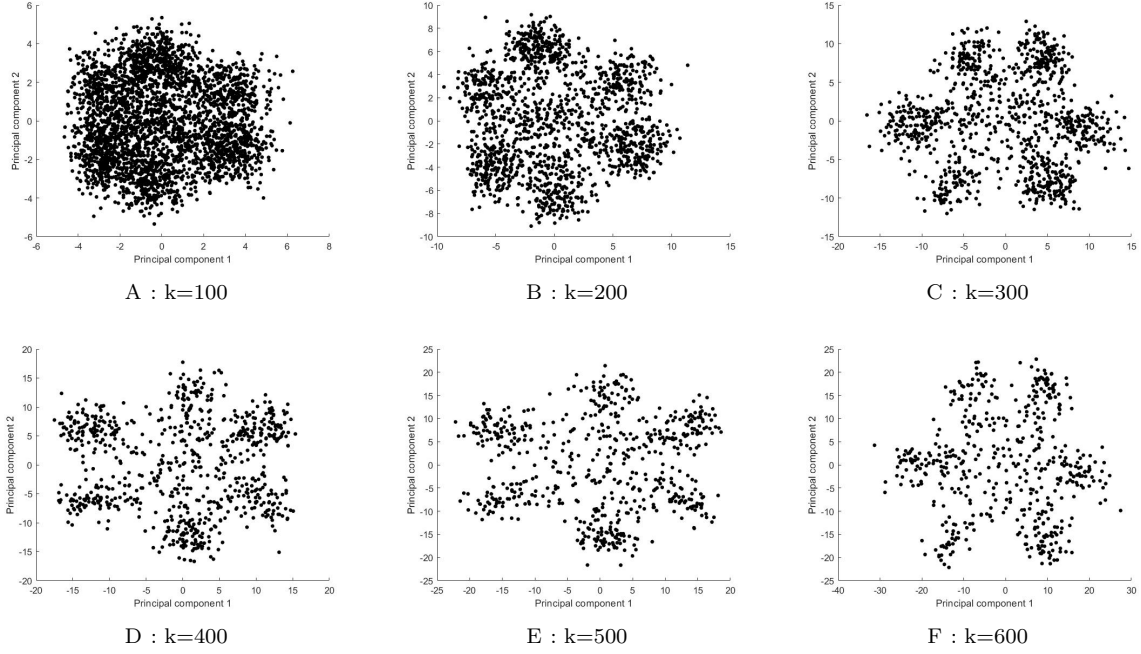


Figure 3: The first two principal components when  $n = 3$  and  $k = 100, \dots, 600$ .

### 3 Problem 3

The PCA result with  $n = 3$  and  $k = 300$  is shown in Figure 4. We can see there are 6 clusters: one sparse and mixed area is surrounded by the six clusters. Of course we can regard the mixed area as the 7th cluster, but when running k-medoids algorithm with 7 clusters, the central area will not be recognized as a cluster because it's sparse.

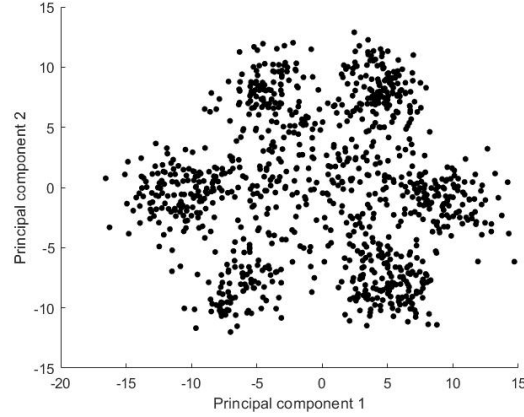


Figure 4: The first two principal components when  $n = 3$  and  $k = 300$ .

Run k-medoids algorithm with 6 clusters and plot the result in Figure 5. The 6 clusters are represented by red points, red circles, red X, blue points, blue circles and blue X. The medoids are green points. As we can see, the 6 clusters are successfully recognized.

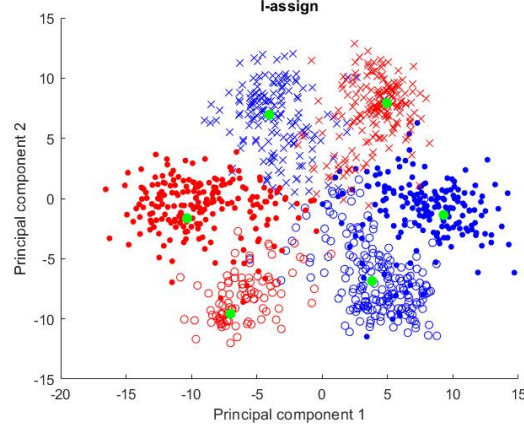


Figure 5: Result of k-medoids method with  $k = 6$ .

## 4 Problem 4

Pick one of the medoids generated in the previous problem and get  $m_j$ . According to Figure 6, the  $m_j$  when  $j$  is in or not in the same cluster are separated but not completely. It corresponds to the existence of mixed area in Figure 5. There are some genes that are difficult to classify.

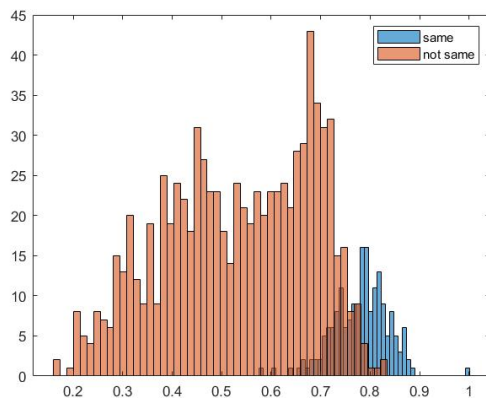


Figure 6: Result of k-medoids method with  $k = 6$ .

## 5 Matlab Code

The code of k-medoids are the same as the previous homeworks. The two function files LoadSeq.m and CalcFreq.m are not included.

### 5.1 Problem 1 and 2

```
1 clear all;clc;
2 tic
3
4 genseq = LoadSeq('ccrescentus.fa');
5
6 k = 300; % number of letters in each document
7 n = 3; % definition of a term as n letters
8 TermDoc = CalcFreq(genseq,n,k);
9 TermDoc = TermDoc';
10
11 % PCA
12 X = TermDoc;
13 Xbar = mean(X,2);
14 Xc = X - Xbar;
15 [u,~,~] = svd(Xc);
16 z = u(:,1:2)'*Xc;
17 scatter(z(1,:),z(2:,:), 'k', 'k.', 'SizeData',200);
18 xlabel('Principal component 1');
19 ylabel('Principal component 2');
20
21 toc
```

## 5.2 Problem 3 and 4

```
1 clear all;clc;
2 tic
3
4 genseq = LoadSeq('ccrescentus.fa');
5
6 k = 300; % number of letters in each document
7 n = 3; % definition of a term as n letters
8 TermDoc = CalcFreq(genseq,n,k);
9 TermDoc = TermDoc';
10
11 % PCA
12 X = TermDoc;
13 Xbar = mean(X,2);
14 Xc = X - Xbar;
15 [u,~,~] = svd(Xc);
16 z = u(:,1:2)'*Xc;
17 % figure();
18 % scatter(z(1,:),z(2,:), 'k', 'k.', 'SizeData',200);
19 % xlabel('Principal component 1');
20 % ylabel('Principal component 2');
21
22 % k-medoids
23 nCluster = 6; % number of clusters
24 tau=1e-5; % tolerance
25 ninit = 20; % times of initialization
26 [~,p]=size(X);
27 % Get distance matrix D
28 D = zeros(p,p);
29 for i=1:p
30     for j=1:p
31         D(i,j)=norm(X(:,i)-X(:,j),2);
32     end
33 end
34 [Iassign, Ibar] = my_k_medoids(nCluster, D, tau, ninit); % k-medoids
35
36 % plot
37 figure();
38 scatter(z(1,Iassign==1),z(2,Iassign==1),'b','k.','SizeData',200);hold on;
39 scatter(z(1,Iassign==2),z(2,Iassign==2),'b','o','SizeData',30);
40 scatter(z(1,Iassign==3),z(2,Iassign==3),'b','x','SizeData',50);
41 scatter(z(1,Iassign==4),z(2,Iassign==4),'r','k.','SizeData',200);
42 scatter(z(1,Iassign==5),z(2,Iassign==5),'r','o','SizeData',30);
43 scatter(z(1,Iassign==6),z(2,Iassign==6),'r','x','SizeData',50);
44 scatter(z(1,Iassign==7),z(2,Iassign==7),'y','k.','SizeData',200);
45
46 % plot medoids
47 if exist('I_bar','var')==1
48     for L=1:nCluster
49         scatter(z(1,I_bar(L)),z(2,I_bar(L)),'g','k.','SizeData',600);
50     end
51 end
52 xlabel('Principal component 1');
53 ylabel('Principal component 2');
54 title('I-assign')
55
56 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
57 % pick one medoid
58 nm = 1; % choosen medoid
59 q = X(:,I_bar(nm));
60 cluster = Iassign(I_bar(nm));
61 % get m
62 m = zeros(p,1);
63 for j=1:p
64     m(j) = q'*X(:,j)/(norm(q,2)*norm(X(:,j),2));
```

```
65 end
66
67 % t(j): if m(j) is in the same cluster, t(j)=1, otherwise t(j)=0
68 t = zeros(p,1);
69 t(I_assign==cluster)=1;
70
71 figure();
72 histogram(m(t==1),50);hold on;
73 histogram(m(t==0),50);
74 legend('same','not same');
75 toc
```