THE UNIVERSITY OF TEXAS
AT AUSTIN

CS363D STATISTICAL LEARNING AND DATA MINING

Homework 02

Edited by LaTeX

Department of Computer Science

STUDENT

Jimmy Lin

xl5224

INSTRUCTOR

Pradeep Ravikumar

TASSISTANT

Adarsh Prasad

RELEASE DATE

Feb. 25 2014

DUE DATE

Mar. 03 2014

TIME SPENT

7 hours

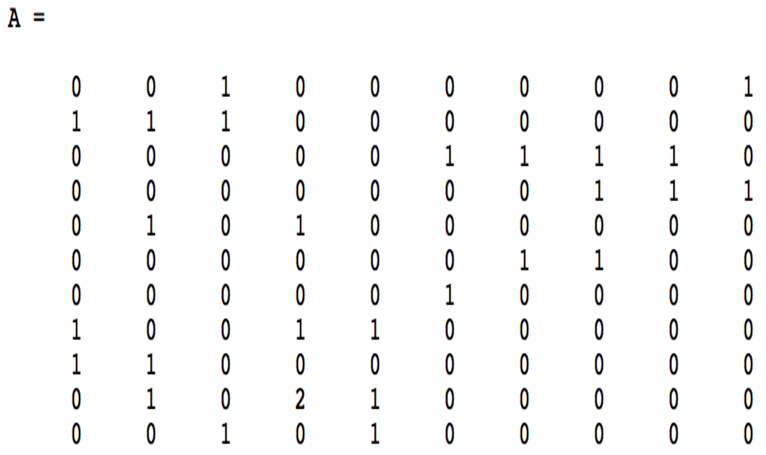
February 26, 2014

Contents

List of Figures

1 Data Transformation and Normalization

1.1 Transformation



```
A =

     0     0     1     0     0     0     0     0     0     1
     1     1     1     0     0     0     0     0     0     0
     0     0     0     0     0     1     1     1     1     0
     0     0     0     0     0     0     0     1     1     1
     0     1     0     1     0     0     0     0     0     0
     0     0     0     0     0     0     1     1     0     0
     0     0     0     0     0     1     0     0     0     0
     1     0     0     1     1     0     0     0     0     0
     1     1     0     0     0     0     0     0     0     0
     0     1     0     2     1     0     0     0     0     0
     0     0     1     0     1     0     0     0     0     0
```

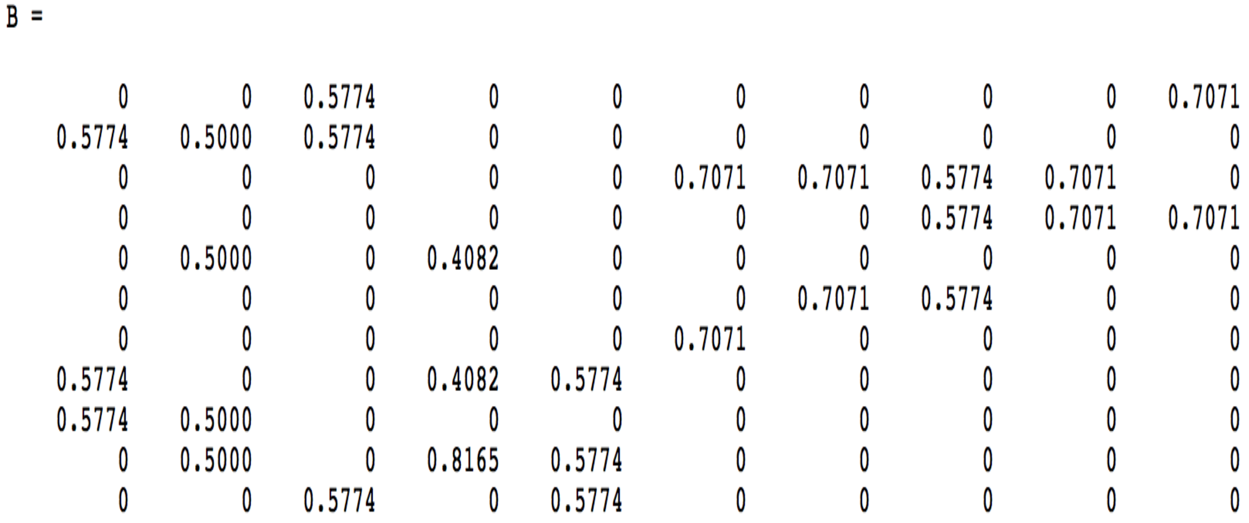
where each document vector (column) correponds to the frequency of terms

$(business, computer, economy, growth, operating, recession, recovery, released, software, system, virus)$

and each term vector (row) represents term frequency in documents

$$(c1, c2, c3, c4, c5, m6, m7, m8, m9, m10)$$

1.2 Normalization

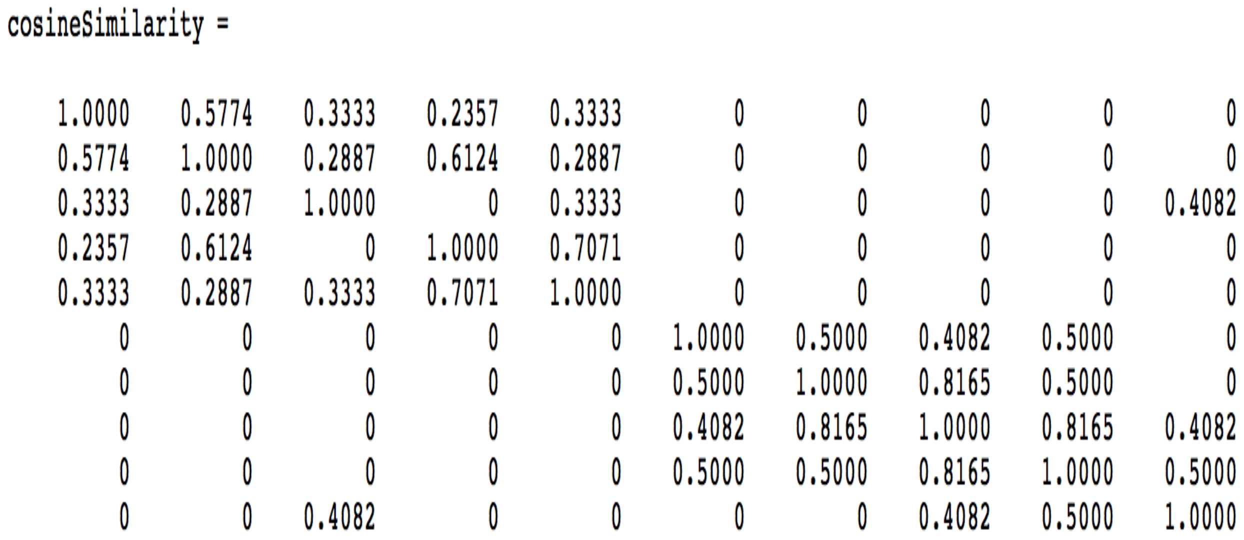


```
B =

         0         0    0.5774         0         0         0         0         0         0    0.7071
    0.5774    0.5000    0.5774         0         0         0         0         0         0         0
         0         0         0         0         0    0.7071    0.7071    0.5774    0.7071         0
         0         0         0         0         0         0         0    0.5774    0.7071    0.7071
         0    0.5000         0    0.4082         0         0         0         0         0         0
         0         0         0         0         0         0    0.7071    0.5774         0         0
         0         0         0         0         0    0.7071         0         0         0         0
    0.5774         0         0    0.4082    0.5774         0         0         0         0         0
    0.5774    0.5000         0         0         0         0         0         0         0         0
         0    0.5000         0    0.8165    0.5774         0         0         0         0         0
         0         0    0.5774         0    0.5774         0         0         0         0         0
```

2 Cosine Similarity

We compute the $B^T B$ and get the following result,

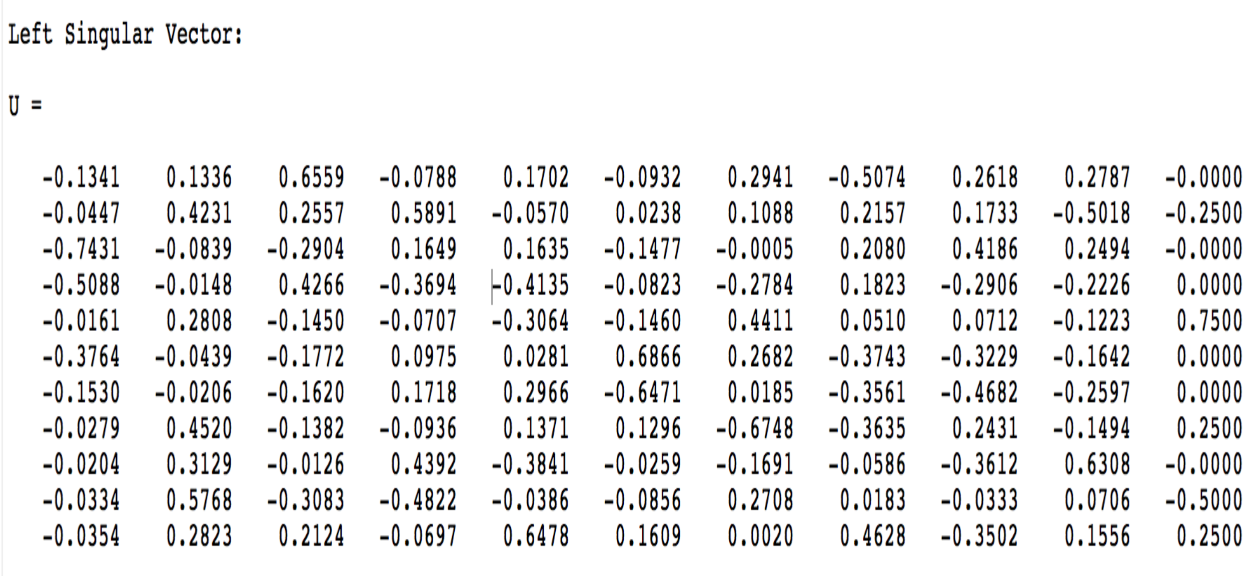


```
cosineSimilarity =

    1.0000    0.5774    0.3333    0.2357    0.3333         0         0         0         0         0
    0.5774    1.0000    0.2887    0.6124    0.2887         0         0         0         0         0
    0.3333    0.2887    1.0000         0    0.3333         0         0         0         0    0.4082
    0.2357    0.6124         0    1.0000    0.7071         0         0         0         0         0
    0.3333    0.2887    0.3333    0.7071    1.0000         0         0         0         0         0
         0         0         0         0         0    1.0000    0.5000    0.4082    0.5000         0
         0         0         0         0         0    0.5000    1.0000    0.8165    0.5000         0
         0         0         0         0         0    0.4082    0.8165    1.0000    0.8165    0.4082
         0         0         0         0         0    0.5000    0.5000    0.8165    1.0000    0.5000
         0         0    0.4082         0         0         0         0    0.4082    0.5000    1.0000
```

Note that each entry e_{ij} represents the cosine similarity between document i and document j.

3 Singular Value Decomposition

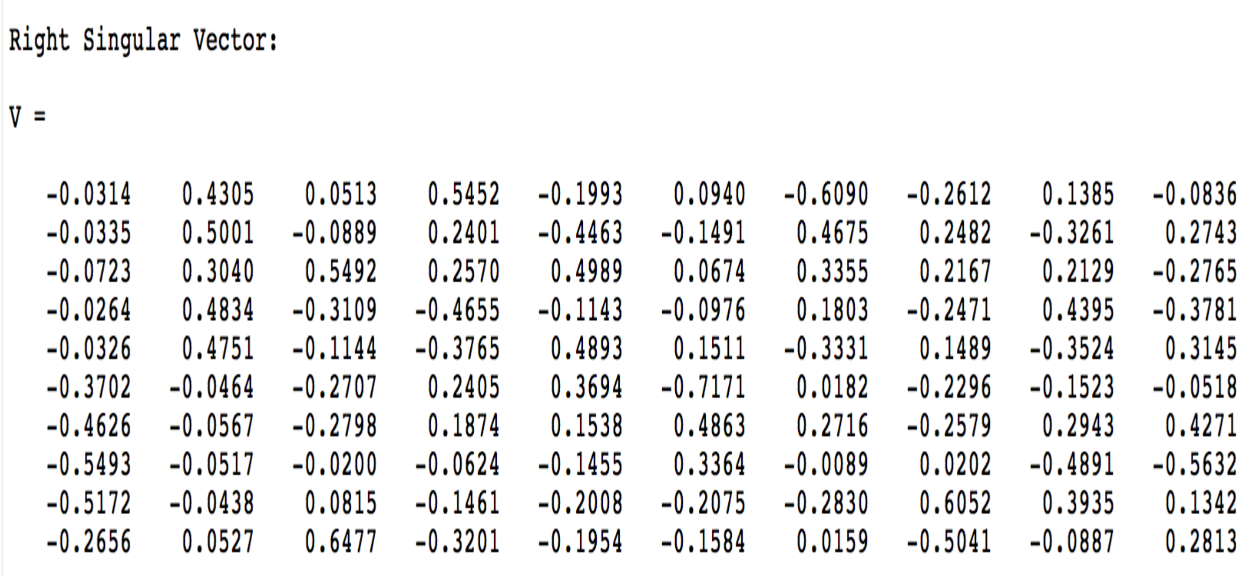


```
Left Singular Vector:

U =

  -0.1341    0.1336    0.6559   -0.0788    0.1702   -0.0932    0.2941   -0.5074    0.2618    0.2787   -0.0000
  -0.0447    0.4231    0.2557    0.5891   -0.0570    0.0238    0.1088    0.2157    0.1733   -0.5018   -0.2500
  -0.7431   -0.0839   -0.2904    0.1649    0.1635   -0.1477   -0.0005    0.2080    0.4186    0.2494   -0.0000
  -0.5088   -0.0148    0.4266   -0.3694   -0.4135   -0.0823   -0.2784    0.1823   -0.2906   -0.2226    0.0000
  -0.0161    0.2808   -0.1450   -0.0707   -0.3064   -0.1460    0.4411    0.0510    0.0712   -0.1223    0.7500
  -0.3764   -0.0439   -0.1772    0.0975    0.0281    0.6866    0.2682   -0.3743   -0.3229   -0.1642    0.0000
  -0.1530   -0.0206   -0.1620    0.1718    0.2966   -0.6471    0.0185   -0.3561   -0.4682   -0.2597    0.0000
  -0.0279    0.4520   -0.1382   -0.0936    0.1371    0.1296   -0.6748   -0.3635    0.2431   -0.1494    0.2500
  -0.0204    0.3129   -0.0126    0.4392   -0.3841   -0.0259   -0.1691   -0.0586   -0.3612    0.6308   -0.0000
  -0.0334    0.5768   -0.3083   -0.4822   -0.0386   -0.0856    0.2708    0.0183   -0.0333    0.0706   -0.5000
  -0.0354    0.2823    0.2124   -0.0697    0.6478    0.1609    0.0020    0.4628   -0.3502    0.1556    0.2500
```

Figure 1: Each Column is a Left Singular Vector



```
Right Singular Vector:

V =

  -0.0314    0.4305    0.0513    0.5452   -0.1993    0.0940   -0.6090   -0.2612    0.1385   -0.0836
  -0.0335    0.5001   -0.0889    0.2401   -0.4463   -0.1491    0.4675    0.2482   -0.3261    0.2743
  -0.0723    0.3040    0.5492    0.2570    0.4989    0.0674    0.3355    0.2167    0.2129   -0.2765
  -0.0264    0.4834   -0.3109   -0.4655   -0.1143   -0.0976    0.1803   -0.2471    0.4395   -0.3781
  -0.0326    0.4751   -0.1144   -0.3765    0.4893    0.1511   -0.3331    0.1489   -0.3524    0.3145
  -0.3702   -0.0464   -0.2707    0.2405    0.3694   -0.7171    0.0182   -0.2296   -0.1523   -0.0518
  -0.4626   -0.0567   -0.2798    0.1874    0.1538    0.4863    0.2716   -0.2579    0.2943    0.4271
  -0.5493   -0.0517   -0.0200   -0.0624   -0.1455    0.3364   -0.0089    0.0202   -0.4891   -0.5632
  -0.5172   -0.0438    0.0815   -0.1461   -0.2008   -0.2075   -0.2830    0.6052    0.3935    0.1342
  -0.2656    0.0527    0.6477   -0.3201   -0.1954   -0.1584    0.0159   -0.5041   -0.0887    0.2813
```

Figure 2: Each Column is a Right Singular Vector

Singluar Values are respectively,

1.7114 , 1.5933 , 1.1817 , 0.9899 , 0.8807 , 0.7837 , 0.6969 ,0.4560 , 0.2300 , 0.1409

4 Plot first two left/right singular matrix

Two Left singular Vectors:
U1 = (-0.1341 , -0.0447 , -0.7431, -0.5088 , -0.0161 , -0.3764 , -0.1530 , -0.0279 , -0.0204 -0.0334 -0.0354)
U2 = (0.1336 , 0.4231 , -0.0839 , -0.0148 , 0.2808 , -0.0439 , -0.0206 , 0.4520 , 0.3129 , 0.5768 , 0.2823)

Two Right singular Vectors:
V1 = (-0.0314 , -0.0335 , -0.0723 , -0.0264 , -0.0326 , -0.3702 , -0.4626, -0.5493 , -0.5172 , -0.2656)
V2 = (0.4305 , 0.5001 , 0.3040 , 0.4834 , 0.4751 , -0.0464 , -0.0567, -0.0517 , -0.0438 , 0.0527)

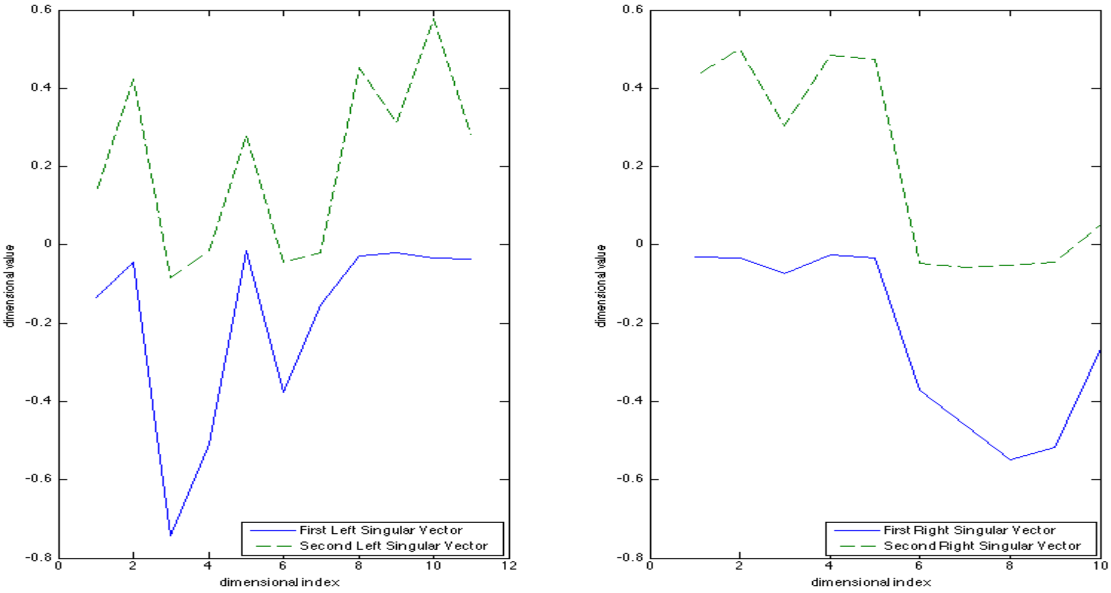


Figure 3: Two Left Columns and Right Columns

5 Plot the projected document vectors

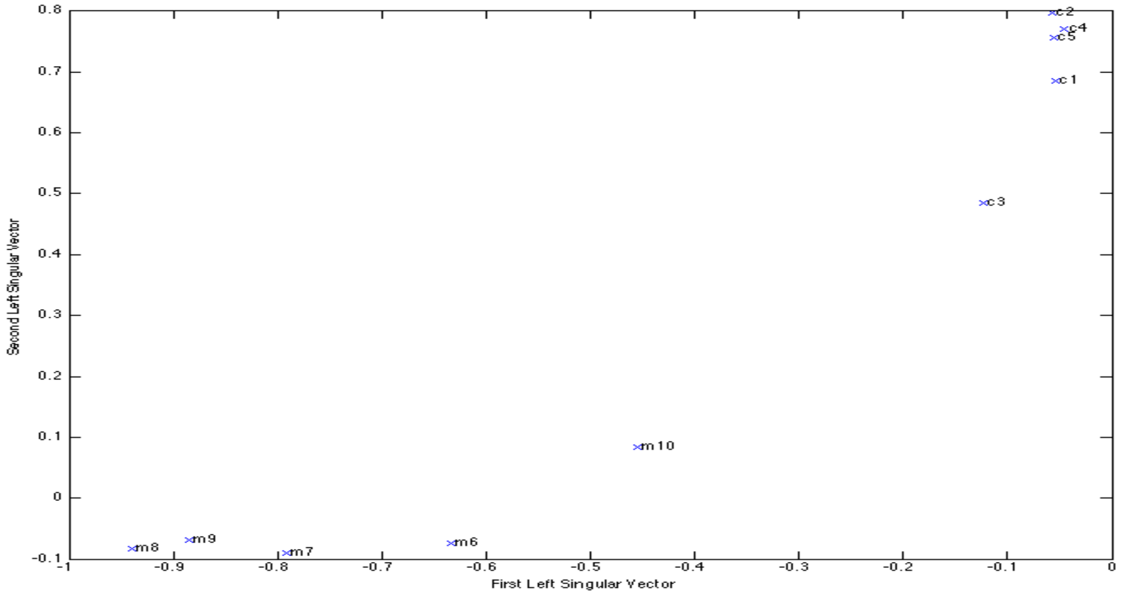


Figure 4: Projected Document Vectors

6 Plot the projected term vectors

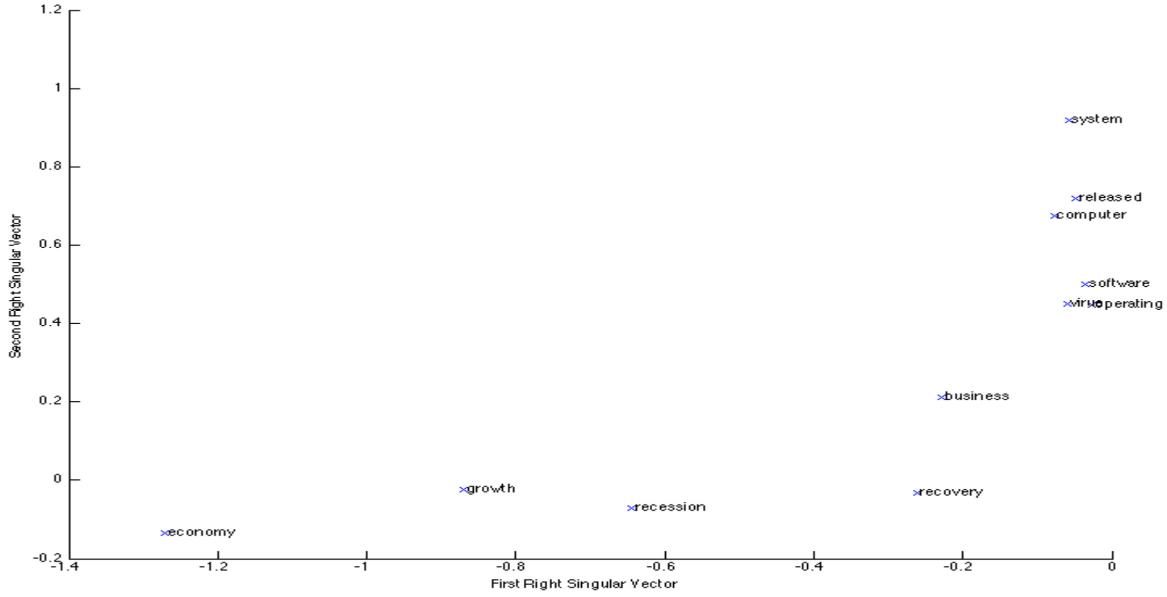


Figure 5: Projected Term Vectors

7 Source Code

```
function y = homework2 ()

%% data input
c1 = {'released' 'software' 'computer'};
c2 = {'operating' 'system' 'computer' 'software'};
c3 = {'computer','virus', 'business'};
c4 = {'system','released', 'operating', 'system'};
c5 = {'released', 'virus', 'system'};
m6 = {'recovery', 'economy'};
m7 = {'economy', 'recession'};
m8 = {'growth', 'economy', 'recession'};
m9 = {'economy', 'growth'};
m10 = {'growth', 'business'};

documents = {c1, c2, c3, c4, c5, m6, m7, m8, m9, m10};

%% keyword generation
collections = [c1, c2, c3, c4, c5, m6, m7, m8, m9, m10];
keywords = unique(collections);
keywords

%% 1(a). generate A matrix
A = [];
for i = 1:size(documents,2),
    term_vector = zeros(size(keywords));
    document = documents{i};
    for j = 1:size(document,2),
        word = document{j};
        indx = find(strcmp(word, keywords));
        term_vector(indx) = term_vector(indx) + 1;
    end
    A = [A term_vector'];
end
A

%% 1(b). Normalize matrix A to get matrix B,
%  where each column (document) vector of B has unit l -norm.
%for col = 1:size(A,2),
%    ltnorm = norm(A(:,col));
%    B(:,col) = A(:,col) / ltnorm;
%end
B = normc(A)


%% 2. Cosine Similarity
cosineSimilarity = B'* B

%% 3. Singular Value Decomposition
[U,Sigma,V] = svd(B);
disp('Left Singular Vector:')
U

disp('Right Singular Vector:')
V

disp('Singluar Values: ')
singularValues = [];
for i = 1:min(size(Sigma)),
    singularValues = [singularValues Sigma(i,i)];
end
disp(singularValues)

%% 4. Plot the first two left and right singular vectors
```

```
U1 = U(:,1)'
U2 = U(:,2)'
a=subplot(1,2,1)
title(a, 'Two Left Singular Vectors')
plot(1:size(U1,2), U1, '-', 1:size(U2,2), U2, '--')
xlabel('dimensional index')
ylabel('dimensional value')
legend('First Left Singular Vector', 'Second Left Singular Vector', 4)
hold on

V1 = V(:,1)'
V2 = V(:,2)'
b=subplot(1,2,2)
title(b,'Two Right Singular Vectors')
plot(1:size(V1,2), V1, '-', 1:size(V2,2), V2, '--')
xlabel('dimensional index')
ylabel('dimensional value')
legend('First Right Singular Vector', 'Second Right Singular Vector', 4)
hold on

%% 5. Plot the projected document vectors in the space
% spanned by the first two left singular vectors.
docNames = {'c1', 'c2', 'c3', 'c4', 'c5', 'm6', 'm7', 'm8', 'm9', 'm10'};
figure;
for i = 1:size(B,2),
   doc_vector = B(:,i);
   x = U1 * doc_vector;
   y = U2 * doc_vector;
   plot([x],[y],'x')
   text(x+0.005,y, docNames{i})
   hold on
end
xlabel('First Left Singular Vector')
ylabel('Second Left Singular Vector')


%% 6. Plot the projected term vectors in the space
% spanned by the first two right singular vectors.
figure;
hold on
for i = 1:size(B,1),
   term_vector = B(i,:);
   x = term_vector * V1';
   y = term_vector * V2';
   plot([x],[y],'x')
   text(x+0.005,y, keywords(i))
   hold on
end
xlabel('First Right Singular Vector')
ylabel('Second Right Singular Vector')
```