

## PROGRAMMING PROJECT 2: SINGULAR VALUE DECOMPOSITION

Prepare a report of this project as a single document. It should address each question in turn and include all of your MATLAB code and output. Try to justify all of your assertions. Print this document to be turned in during class on December 7. You should also email copies of your code to our TA Yuji Nakatsukasa at `ynakam at math dot ucdavis dot edu`.

(1) Write a MATLAB function that uses the singular value decomposition to produce a rank  $k$  approximation of an  $m \times n$  real matrix  $A$ .

Your input should include:

- A real  $m \times n$  matrix  $A$ .
- A parameter  $k$  which is at least 1 and less than or equal to the rank of  $A$ .

Your output should include:

- A matrix  $B$  that is the rank  $k$  approximation of  $A$  obtained from the  $k$  most significant singular values of  $A$ .

You can use the command `[U,S,V] = svd(A)` in MATLAB to obtain the singular value decomposition, if you like.

(2) Test your program from (1) by performing image compression.

(a) Load the  $300 \times 450$  matrix `IM.mat` into MATLAB using the `load IM` command. (You may need to first change to the directory where you saved the file.) This matrix contains image data, where each entry of the matrix represents the greyscale value of a pixel. You can view the matrix as an image using the `imshow(IM)` command.

(b) Use your program from (1) to obtain rank  $k$  approximations of `IM` with  $k = 1, 10, 40, 80$ . You don't need to turn these in but you might want to view them with `imshow`.

(c) How much image compression occurs with each value of  $k$  from (b)?

(d) Do any of the values of  $k$  from (b) give a good visual approximation to the picture?

(3) Suppose you are working for a search engine company that is indexing a subnet of technical documents. The web-indexing program at your company has produced the following term-by-document matrix  $D$  whose entries give the occurrences of 12 terms among 9 documents. The first 5 documents are related to human-computer interaction, while the last 4 documents are related to graph theory. You can download this matrix as the file `D.mat` from the class website.

d1	Human machine interface for Lab ABC computer applications
d2	A survey of user opinion of computer system response time
d3	The EPS user interface management system
d4	System and human system engineering testing of EPS
d5	Relation of user-perceived response time to error measurement
d6	The generation of random, binary, unordered trees
d7	The intersection graph of paths in trees
d8	Graph minors IV: Widths of trees and well-quasi-ordering
d9	Graph minors: A survey

q1	computer
q2	EPS
q3	human
q4	interface
q5	response
q6	system
q7	time
q8	user
q9	graph
q10	minors
q11	survey
q12	trees

$$D = \begin{bmatrix} & d1 & d2 & d3 & d4 & d5 & d6 & d7 & d8 & d9 \\ q1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ q2 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ q3 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ q4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ q5 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ q6 & 0 & 1 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\ q7 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ q8 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ q9 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ q10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ q11 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ q12 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

(a) Consider the query “user interface survey.” In your opinion, which documents do you think are relevant to this query?

(b) Write a MATLAB program to perform query matching against a term-by-document matrix. Your input should include a term-by-document matrix  $D$ , a query vector  $q$ , and a threshold value  $t$ . Your program should compute the cosine of the angle between each column of  $D$  and the query vector  $q$ . If the cosine is greater than  $t$ , then the program should output the document number together with the cosine.

(c) Express the query “user interface survey” as a vector and use your program from (b) to find the most relevant documents from the term-by-document matrix  $D$ . Use a threshold value of 0.2.

(d) Use your program from (1) to compute a rank 2 approximation  $E$  of  $D$ .

(e) Now use your program from (b) to find the most relevant documents from the rank 2 term-by-document matrix  $E$  for the query “user interface survey.” Use a threshold value of 0.2.

(f) Figure 1 shows a 2-dimensional scatterplot of “concept space” from the low-rank approximation of  $D$ . The search terms are diamonds labeled by  $q_1, \dots, q_{12}$  while the documents are squares labeled by  $d_1, \dots, d_9$ .

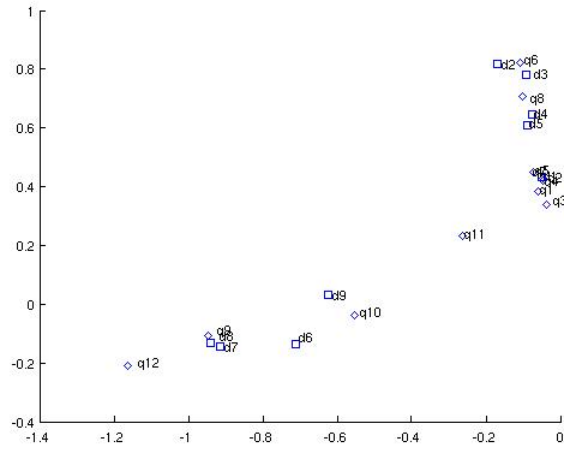


FIGURE 1. Rank 2 representation of the term-by-document matrix.

Do you think this low-rank approximation does a good job of “clustering” related concepts? In general terms, describe what the analogous scatterplot for the original full-rank “concept space” corresponding to the matrix  $D$  would look like (e.g. how many dimensions would such a scatterplot have?).

(g) Which query result (from (c) or (e)) was more relevant? Can you explain how one query method returned more relevant documents than the other?