1. (40 points) Tinker with the MATLAB function `svd` to gain intuition into how the singular value decomposition works. Then answer the following questions:

- For the term-document (terms are rows, documents are columns) matrix \( A \) given in http://people.cs.vt.edu/~ramakris/Courses/CS6604/data/7a.txt, compute the SVD and create an approximation to \( A \) of rank 2. Then visualize the term vectors and document vectors in the 2D space and make interesting observations.

- When we create an approximation using the SVD, the new matrix _______ contain negative entries. Fill in the blanks with either ‘will’ or ‘might’ and give a scientific explanation why this happens.

- We claimed in class that SVD can be viewed through a maximum likelihood lens, assuming independent observations and normal error. What exactly does this mean in an information retrieval context? Which ‘things’ are supposed to be independent, and what exactly is normally distributed?

- We know that information retrieval using SVD reduces to the vector space model (VSM) when we do not ‘drop’ any terms in the singular value decomposition. Thus, VSM is a simple specialization of using the SVD. Is the generalized vector-space model (GVSM) also a specialization of the SVD? In this approach, the similarity between two document vectors \( d_1 \) and \( d_2 \) (which are column vectors of matrix \( A \)) is captured as \( d_1^T A A^T d_2 \). As this expression reveals, GVSM captures term-term correlations and uses them to prescribe similarities between documents.

2. (60 points) In this question, you will take the mushroom dataset from the UCI Machine Learning Repository (http://www.ics.uci.edu/~mlearn/MLSummary.html) and conduct the Koller-Sahami feature selection strategy. First, separate out the dataset into two parts: a training set and a test set, with the training portion containing 2/3 of the mushrooms (make sure percentages of edible and poisonous ones are retained). Apply the feature selection strategy systematically, removing one feature at a time. At each step, evaluate the performance of a Naive Bayes classifier constructed from the training set on the test set data. Track the error (or accuracy) as more and more features are removed. Prepare a set of observations. What do you learn from this exercise?