1.7 Exercises

Exercise 1.1 (Best fit functions versus best least squares fit) *In many experiments one collects the value of a parameter at various instances of time. Let y_i be the value of the parameter y at time x_i. Suppose we wish to construct the best linear approximation to the data in the sense that we wish to minimize the mean square error. Here error is measured vertically rather than perpendicular to the line. Develop formulas for m and b to minimize the mean square error of the points $\{(x_i, y_i)\,|1 \leq i \leq n\}$ to the line $y = mx + b$.*

Exercise 1.2 *Given five observed parameters, height, weight, age, income, and blood pressure of n people, how would one find the best least squares fit subspace of the form*

$$a_1\,(\textit{height}) + a_2\,(\textit{weight}) + a_3\,(\textit{age}) + a_4\,(\textit{income}) + a_5\,(\textit{blood pressure}) = 0$$

Here $a_1, a_2, \ldots, a_5$ are the unknown parameters. If there is a good best fit 4-dimensional subspace, then one can think of the points as lying close to a 4-dimensional sheet rather than points lying in 5-dimensions. Why is it better to use the perpendicular distance to the subspace rather than vertical distance where vertical distance to the subspace is measured along the coordinate axis corresponding to one of the unknowns?

Exercise 1.3 *What is the best fit line for each of the following set of points?*

1. $\{(0,1)\,,(1,0)\}$
2. $\{(0,1)\,,(2,0)\}$
3. *The rows of the matrix*

$$\begin{pmatrix} 17 & 4 \\ -2 & 26 \\ 11 & 7 \end{pmatrix}$$

Solution: (1) and (2) are easy to do from scratch. (1) $y = x$ and (2) $y = 2x$. For (3), there is no simple method. We will describe a general method later and this can be applied. But the best fit line is $v_1 = \frac{1}{\sqrt{5}}\begin{pmatrix} 1 \\ 2 \end{pmatrix}$. **FIX** Convince yourself that this is correct. ∎

Exercise 1.4 *Let A be a square $n \times n$ matrix whose rows are orthonormal. Prove that the columns of A are orthonormal.*

Solution: Since the rows of A are orthonormal $AA^T = I$ and hence $A^TAA^T = A^T$. Since A^T is nonsingular it has an inverse $\left(A^T\right)^{-1}$. Thus $A^TAA^T\left(A^T\right)^{-1} = A^T\left(A^T\right)^{-1}$ implying that $A^TA = I$, i.e., the columns of A are orthonormal. ∎

Exercise 1.5 *Suppose A is a $n \times n$ matrix with block diagonal structure with k equal size blocks where all entries of the i^{th} block are a_i with $a_1 > a_2 > \cdots > a_k > 0$. Show that A has exactly k nonzero singular vectors $\mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_k}$ where $\mathbf{v_i}$ has the value $(\frac{k}{n})^{1/2}$ in the coordinates corresponding to the i^{th} block and 0 elsewhere. In other words, the singular vectors exactly identify the blocks of the diagonal. What happens if $a_1 = a_2 = \cdots = a_k$? In the case where the a_i are equal, what is the structure of the set of all possible singular vectors?*
Hint: *By symmetry, the top singular vector's components must be constant in each block.*

Exercise 1.6 *Prove that the left singular vectors of A are the right singular vectors of A^T.*

Solution: $A = UDV^T$, thus $A^T = VDU^T$. ∎

Exercise 1.7 *Interpret the right and left singular vectors for the document term matrix.*

Solution: The first right singular vector is a synthetic document that best matches the collection of documents. The first left singular vector is a synthetic word that best matches the collection of terms appearing in the documents. ∎

Exercise 1.8 *Verify that the sum of rank one matrices $\sum\limits_{i=1}^{r} \sigma_i \mathbf{u_i} \mathbf{v_i}^T$ can be written as UDV^T, where the $\mathbf{u_i}$ are the columns of U and $\mathbf{v_i}$ are the columns of V. To do this, first verify that for any two matrices P and Q, we have*

$$PQ = \sum_i \mathbf{p_i} \mathbf{q_i}^T$$

where $\mathbf{p_i}$ is the i^{th} column of P and $\mathbf{q_i}$ is the i^{th} column of Q.

Exercise 1.9

1. *Show that the rank of A is r where r is the miminum i such that $\underset{\substack{\mathbf{v} \perp \mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v}_i \\ |\mathbf{v}|=1}}{\arg\max} |A\ \mathbf{v}| = 0$.*

2. *Show that $\left|\mathbf{u_1^T} A\right| = \max\limits_{|\mathbf{u}|=1} \left|\mathbf{u^T} A\right| = \sigma_1$.*

Hint: *Use SVD.*

Exercise 1.10 *If $\sigma_1, \sigma_2, \ldots, \sigma_r$ are the singular values of A and $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r$ are the corresponding right singular vectors, show that*

1. $A^T A = \sum\limits_{i=1}^{r} \sigma_i^2 \mathbf{v_i} \mathbf{v_i}^T$

2. $\mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_r$ *are eigenvectors of* A^TA.

3. *Assuming that the set of eigenvectors of a matrix is unique, conclude that the set of singular values of the matrix is unique.*

See the appendix for the definition of eigenvectors.

Exercise 1.11 *Let A be a matrix. Given an algorithm for finding*

$$\mathbf{v_1} = \arg\max_{|\mathbf{v}|=1} |A\mathbf{v}|$$

describe an algorithm to find the SVD of A.

Exercise 1.12 *Compute the singular valued decomposition of the matrix*

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

Exercise 1.13 *Write a program to implement the power method for computing the first singular vector of a matrix. Apply your program to the matrix*

$$A = \begin{pmatrix} 1 & 2 & 3 & \cdots & 9 & 10 \\ 2 & 3 & 4 & \cdots & 10 & 0 \\ \vdots & \vdots & \vdots & & & \vdots \\ 9 & 10 & 0 & \cdots & 0 & 0 \\ 10 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}$$

Exercise 1.14 *Modify the power method to find the first four singular vectors of a matrix A as follows. Randomly select four vectors and find an orthonormal basis for the space spanned by the four vectors. Then multiple each of the basis vectors times A and find a new orthonormal basis for the space spanned by the resulting four vectors. Apply your method to find the first four singular vectors of matrix A of Exercise 1.13*

Exercise 1.15 *Let A be a real valued matrix. Prove that* $B = AA^T$ *is positive definite.*

Exercise 1.16 *Prove that the eigenvalues of a symmetric real valued matrix are real.*

Exercise 1.17 *Suppose A is a square invertible matrix and the SVD of A is* $A = \sum_i \sigma_i u_i v_i^T$. *Prove that the inverse of A is* $\sum_i \frac{1}{\sigma_i} v_i u_i^T$.

Exercise 1.18 *Suppose A is square, but not necessarily invertible and has SVD* $A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$. *Let* $B = \sum_{i=1}^{r} \frac{1}{\sigma_i} v_i u_i^T$. *Show that* $B\mathbf{x} = \mathbf{x}$ *for all* $\mathbf{x}$ *in the span of the right singular vectors of A. For this reason B is sometimes called the pseudo inverse of A and can play the role of* A^{-1} *in many applications.*

Exercise 1.19

1. *For any matrix A, show that* $\sigma_k \le \frac{||A||_F}{\sqrt{k}}$.

2. *Prove that there exists a matrix B of rank at most k such that* $||A - B||_2 \le \frac{||A||_F}{\sqrt{k}}$.

3. *Can the 2-norm on the left hand side in (b) be replaced by Frobenius norm?*

Exercise 1.20 *Suppose an* $n \times d$ *matrix A is given and you are allowed to preprocess A. Then you are given a number of d-dimensional vectors* $\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_m}$ *and for each of these vectors you must find the vector* $A\mathbf{x_i}$ *approximately, in the sense that you must find a vector* $\mathbf{u_i}$ *satisfying* $|\mathbf{u_i} - A\mathbf{x_i}| \le \varepsilon ||A||_F |\mathbf{x_i}|$. *Here* $\varepsilon > 0$ *is a given error bound. Describe an algorithm that accomplishes this in time* $O\left(\frac{d+n}{\varepsilon^2}\right)$ *per* $\mathbf{x_i}$ *not counting the preprocessing time.*

Exercise 1.21 ***(Constrained Least Squares Problem using SVD)*** *Given A,* $\mathbf{b}$, *and m, use the SVD algorithm to find a vector* $\mathbf{x}$ *with* $|\mathbf{x}| < m$ *minimizing* $|A\mathbf{x} - \mathbf{b}|$. *This problem is a learning exercise for the advanced student. For hints/solution consult Golub and van Loan, Chapter 12.*

Exercise 1.22 (Document-Term Matrices): *Suppose we have a* $m \times n$ *document-term matrix where each row corresponds to a document where the rows have been normalized to length one. Define the "similarity" between two such documents by their dot product.*

1. *Consider a "synthetic" document whose sum of squared similarities with all documents in the matrix is as high as possible. What is this synthetic document and how would you find it?*

2. *How does the synthetic document in (1) differ from the center of gravity?*

3. *Building on (1), given a positive integer k, find a set of k synthetic documents such that the sum of squares of the mk similarities between each document in the matrix and each synthetic document is maximized. To avoid the trivial solution of selecting k copies of the document in (1), require the k synthetic documents to be orthogonal to each other. Relate these synthetic documents to singular vectors.*

4. *Suppose that the documents can be partitioned into k subsets (often called clusters), where documents in the same cluster are similar and documents in different clusters are not very similar. Consider the computational problem of isolating the clusters. This is a hard problem in general. But assume that the terms can also be partitioned into k clusters so that for* $i \ne j$, *no term in the* i^{th} *cluster occurs in a document in the* j^{th} *cluster. If we knew the clusters and arranged the rows and columns in them to be contiguous, then the matrix would be a block-diagonal matrix. Of course*

the clusters are not known. By a "block" of the document-term matrix, we mean a submatrix with rows corresponding to the i^{th} cluster of documents and columns corresponding to the i^{th} cluster of terms . We can also partition any n vector into blocks. Show that any right singular vector of the matrix must have the property that each of its blocks is a right singular vector of the corresponding block of the document-term matrix.

5. *Suppose now that the singular values of all the blocks are distinct (also across blocks). Show how to solve the clustering problem.*

Hint: *(4) Use the fact that the right singular vectors must be eigenvectors of A^TA. Show that A^TA is also block-diagonal and use properties of eigenvectors.*

Solution: (1)
(2)
(3): It is obvious that A^TA is block diagonal. We claim that for any block-diagonal symmetric matrix B, each eigenvector must be composed of eigenvectors of blocks. To see this, just note that since for an eigenvector $\mathbf{v}$ of B, $B\mathbf{v}$ is $\lambda\mathbf{v}$ for a real λ, for a block B_i of B, $B_i\mathbf{v}$ is also λ times the corresponding block of $\mathbf{v}$.
(4): By the above, it is easy to see that each eigenvector of A^TA has nonzero entries in just one block.
(e) ∎

Exercise 1.23 *Generate a number of samples according to a mixture of 1-dimensional Gaussians. See what happens as the centers get closer. Alternatively, see what happens when the centers are fixed and the standard deviation is increased.*

Exercise 1.24 *Show that maximizing $\mathbf{x}^T\mathbf{u}\mathbf{u}^T(\mathbf{1}-\mathbf{x})$ subject to $x_i \in \{0,1\}$ is equivalent to partitioning the coordinates of $\mathbf{u}$ into two subsets where the sum of the elements in both subsets are equal.*

Solution: $\mathbf{x}^T\mathbf{u}\mathbf{u}^T(\mathbf{1}-\mathbf{x})$ can be written as the product of two scalars $\left(\mathbf{x}^T\mathbf{u}\right)\left(\mathbf{u}^T(\mathbf{1}-\mathbf{x})\right)$. The first scalar is the sum of the coordinates of $\mathbf{u}$ corresponding to the subset S and the second scalar is the sum of the complementary coordinates of $\mathbf{u}$. To maximize the product, one partitions the coordinates of $\mathbf{u}$ so that the two sums are as equally as possible. Given the subset determined by the maximization, check if $\mathbf{x}^T\mathbf{u} = \mathbf{u}^T(\mathbf{1}-\mathbf{x})$. ∎

Exercise 1.25 *Read in a photo and convert to a matrix. Perform a singular value decomposition of the matrix. Reconstruct the photo using only 10%, 25%, 50% of the singular values.*

1. *Print the reconstructed photo. How good is the quality of the reconstructed photo?*

2. *What percent of the Forbenius norm is captured in each case?*

Hint: *If you use Matlab, the command to read a photo is imread. The types of files that can be read are given by imformats. To print the file use imwrite. Print using jpeg format. To access the file afterwards you may need to add the file extension .jpg. The command imread will read the file in uint8 and you will need to convert to double for the SVD code. Afterwards you will need to convert back to uint8 to write the file. If the photo is a color photo you will get three matrices for the three colors used.*

Exercise 1.26 *Find a collection of something such as photgraphs, drawings, or charts and try the SVD compression technique on it. How well does the reconstruction work?*

Exercise 1.27 *Create a set of 100,* 100×100 *matrices of random numbers between 0 and 1 such that each entry is highly correlated with the adjacency entries. Find the SVD of A. What fraction of the Frobenius norm of A is captured by the top 100 singular vectors? How many singular vectors are required to capture 95% of the Frobenius norm?*

Exercise 1.28 *Create a* 100×100 *matrix A of random numbers between 0 and 1 such that each entry is highly correlated with the adjacency entries and find the first 100 vectors for a single basis that is reasonably good for all 100 matrices. How does one do this? What fraction of the Frobenius norm of a new matrix is captured by the basis?*

Solution: If $\mathbf{v_1}, \mathbf{v_2}, \cdots, \mathbf{v_{100}}$ is the basis, then $A = A\mathbf{v_1}\mathbf{v_1}^T + A\mathbf{v_2}\mathbf{v_2}^T + \cdots$. ∎

Exercise 1.29 *Show that the running time for the maximum cut algorithm in Section* **??** *can be carried out in time* $O(n^3 + poly(n)k^k)$*, where poly is some polynomial.*

Exercise 1.30 *Let* $x_1, x_2, \ldots, x_n$ *be n points in d-dimensional space and let X be the* $n \times d$ *matrix whose rows are the n points. Suppose we know only the matrix D of pairwise distances between points and not the coordinates of the points themselves. The* x_{ij} *are not unique since any translation, rotation, or reflection of the coordinate system leaves the distances invariant. Fix the origin of the coordinate system so that the centroid of the set of points is at the origin.*

1. *Show that the elements of* X^TX *are given by*

$$x_i^T x_j = -\frac{1}{2}\left[d_{ij}^2 - \frac{1}{n}\sum_{j=1}^{n} d_{ij}^2 - \frac{1}{n}\sum_{i=1}^{n} d_{ij}^2 + \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}^2\right].$$

2. *Describe an algorithm for determining the matrix X whose rows are the* x_i.

Solution: (1) Since the centroid of the set of points is at the origin of the coordinate axes, $\sum_{i=1}^{n} x_{ij} = 0$. Write

$$d_{ij}^2 = (x_i - x_j)^T(x_i - x_j) = x_i^T x_i + x_j^T x_j - 2x_i^T x_j \tag{1.3}$$

Then

$$\frac{1}{n}\sum_{i=1}^{n} d_{ij}^2 = \frac{1}{n}\sum_{i=1}^{n} x_i^T x_i + x_j^T x_j \tag{1.4}$$

Since $\frac{1}{n}\sum_{i=1}^{n} x_j^T x_j = x_j^T x_j$ and $\frac{1}{n}\left(\sum_{i=1}^{n} x_i^T\right) x_j = 0$.
Similarly

$$\frac{1}{n}\sum_{j=1}^{n} d_{ij}^2 = \frac{1}{n}\sum_{j=1}^{n} x_j^T x_j + x_i^T x_i \tag{1.5}$$

Summing (1.4) over j gives

$$\frac{1}{n}\sum_{j=1}^{n}\sum_{i=1}^{n} d_{ij}^2 = \sum_{i=1}^{n} x_i^T x_i + \sum_{j=1}^{n} x_j^T x_j = 2\sum_{i=1}^{n} x_i^T x_i \tag{1.6}$$

Rearranging (1.3) and substituting for $x_i^T x_i$ and $x_j^T x_j$ from (1.3) and (1.4) yields

$$x_i^T x_j = -\frac{1}{2}\left(d_{ij}^2 - x_i^T x_i - x_j^T x_j\right) = -\frac{1}{2}\left(d_{ij}^2 - \frac{1}{n}\sum_{j=1}^{n} d_{ij}^2 - \frac{1}{n}\sum_{i=1}^{n} d_{ij}^2 + \frac{2}{n}\sum_{i=1}^{n} x_i^T x_i\right)$$

Finally substituting (1.6) yields

$$x_i^T x_j = -\frac{1}{2}\left(d_{ij}^2 - x_i^T x_i - x_j^T x_j\right) = -\frac{1}{2}\left(d_{ij}^2 - \frac{1}{n}\sum_{j=1}^{n} d_{ij}^2 - \frac{1}{n}\sum_{i=1}^{n} d_{ij}^2 + \frac{1}{n^2}\sum_{j=1}^{n}\sum_{i=1}^{n} d_{ij}^2\right)$$

Note that is D is the matrix of pairwise squared distances, then $\frac{1}{n}\sum_{k=1}^{n} d_{ij}^2$, $\frac{1}{n}\sum_{i=1}^{n} d_{ij}^2$, and $\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}^2$ are the averages of the square of the elements of the i^{th} row, the square of the elements of the j^{th} column and all squared distances respectively.
(2) Having constructed $X^T X$ we can use an eigenvalue decomposition to determine the coordinate matrix X. Clearly $X^T X$ is symmetric and if the distances come from a set of n points in a d-dimensional space $X^T X$ will be positive definite and of rank d. Thus we can decompose $X^T X$ as$X^T X = V^T \sigma V$ where the first d eigenvalues are positive and the remainder are zero. Since the $X^T X = V^T \sigma^{\frac{1}{2}} \sigma^{\frac{1}{2}} V$ and thus the coordinates are given by $X = V^T \sigma^{\frac{1}{2}}$

Exercise 1.31

1. *Consider the pairwise distance matrix for twenty US cities given below. Use the algorithm of Exercise 2 to place the cities on a map of the US.*

2. *Suppose you had airline distances for 50 cities around the world. Could you use these distances to construct a world map?*

	B	B	C	D	D	H	L	M	M	M
	O	U	H	A	E	O	A	E	I	I
	S	F	I	L	N	U		M	A	M
Boston	-	400	851	1551	1769	1605	2596	1137	1255	1123
Buffalo	400	-	454	1198	1370	1286	2198	803	1181	731
Chicago	851	454	-	803	920	940	1745	482	1188	355
Dallas	1551	1198	803	-	663	225	1240	420	1111	862
Denver	1769	1370	920	663	-	879	831	879	1726	700
Houston	1605	1286	940	225	879	-	1374	484	968	1056
Los Angeles	2596	2198	1745	1240	831	1374	-	1603	2339	1524
Memphis	1137	803	482	420	879	484	1603	-	872	699
Miami	1255	1181	1188	1111	1726	968	2339	872	-	1511
Minneapolis	1123	731	355	862	700	1056	1524	699	1511	-
New York	188	292	713	1374	1631	1420	2451	957	1092	1018
Omaha	1282	883	432	586	488	794	1315	529	1397	290
Philadelphia	271	279	666	1299	1579	1341	2394	881	1019	985
Phoenix	2300	1906	1453	887	586	1017	357	1263	1982	1280
Pittsburgh	483	178	410	1070	1320	1137	2136	660	1010	743
Saint Louis	1038	662	262	547	796	679	1589	240	1061	466
Salt Lake City	2099	1699	1260	999	371	1200	579	1250	2089	987
San Francisco	2699	2300	1858	1483	949	1645	347	1802	2594	1584
Seattle	2493	2117	1737	1681	1021	1891	959	1867	2734	1395
Washington D.C.	393	292	597	1185	1494	1220	2300	765	923	934

	N	O	P	P	P	S	S	S	S	D
	Y	M	H	H	I	t	L	F	E	C
		A	I	O	T	L	C		A	
Boston	188	1282	271	2300	483	1038	2099	2699	2493	393
Buffalo	292	883	279	1906	178	662	1699	2300	2117	292
Chicago	713	432	666	1453	410	262	1260	1858	1737	597
Dallas	1374	586	1299	887	1070	547	999	1483	1681	1185
Denver	1631	488	1579	586	1320	796	371	949	1021	1494
Houston	1420	794	1341	1017	1137	679	1200	1645	1891	1220
Los Angeles	2451	1315	2394	357	2136	1589	579	347	959	2300
Memphis	957	529	881	1263	660	240	1250	1802	1867	765
Miami	1092	1397	1019	1982	1010	1061	2089	2594	2734	923
Minneapolis	1018	290	985	1280	743	466	987	1584	1395	934
New York	-	1144	83	2145	317	875	1972	2571	2408	205
Omaha	1144	-	1094	1036	836	354	833	1429	1369	1014
Philadelphia	83	1094	-	2083	259	811	1925	2523	2380	123
Phoenix	2145	1036	2083	-	1828	1272	504	653	1114	1963
Pittsburgh	317	836	259	1828	-	559	1668	2264	2138	192
Saint Louis	875	354	811	1272	559	-	1162	1744	1724	712
Salt Lake City	1972	833	1925	504	1668	1162	-	600	701	1848
San Francisco	2571	1429	2523	653	2264	1744	600	-	678	2442
Seattle	2408	1369	2380	1114	2138	1724	701	678	-	2329
Washington D.C.	250	1014	123	1983	192	712	1848	2442	2329	-

References

Hubs and Authorities
Golub and Van Loan
Clustering Gaussians