Chapter 11

The Singular Value Decomposition

One singular sensation...
A Chorus Line, lyrics by Edward Kleban

In the last chapter, we studied special matrices—a Haar basis matrix, a Fourier matrix, a circulant matrix—such that multiplying by such a matrix was fast. Moreover, storage for such a matrix is very cheap. The Haar and Fourier matrices can be represented implicitly, by procedures, and a circulant matrix can be represented by storing just its first row, for the other rows can be derived from that one row.

11.1 Approximation of a matrix by a low-rank matrix

11.1.1 The benefits of low-rank matrices

A low-rank matrix has the same benefits. Consider a matrix whose rank is one. All the rows lie in a one-dimensional space. Let $\{\boldsymbol{v}\}$ be a basis for that space. Every row of the matrix is some scalar multiple of $\boldsymbol{v}$. Let $\boldsymbol{u}$ be the vector whose entries are these scalar multiples. Then the matrix can be written as $\boldsymbol{u}\boldsymbol{v}^T$. Such a representation requires small storage—just $m + n$ numbers have to be stored for a rank-one $m \times n$ matrix. Moreover, to multiply the matrix $\boldsymbol{u}\boldsymbol{v}^T$ by a vector $\boldsymbol{w}$, we use the equation

$$\left(\begin{bmatrix} \\ \boldsymbol{u} \\ \\ \end{bmatrix} \begin{bmatrix} & \boldsymbol{v}^T & \end{bmatrix}\right) \begin{bmatrix} \\ \boldsymbol{w} \\ \\ \end{bmatrix} = \begin{bmatrix} \\ \boldsymbol{u} \\ \\ \end{bmatrix} \left(\begin{bmatrix} & \boldsymbol{v}^T & \end{bmatrix} \begin{bmatrix} \\ \boldsymbol{w} \\ \\ \end{bmatrix}\right)$$

which shows that the matrix-vector product can be computed by computing two dot-products.

Even if a matrix has rank more than one, if the rank of a matrix is small, we get some of the

same benefits. A rank-two matrix, for example, can be written as

$$\left[\begin{array}{c|c} \boldsymbol{u}_1 & \boldsymbol{u}_2 \end{array}\right] \left[\begin{array}{c} \boldsymbol{v}_1^T \\ \hline \boldsymbol{v}_2^T \end{array}\right]$$

so it can be stored compactly and can be multiplied by a vector quickly.

Unfortunately, most matrices derived from observed data do not have low rank. Fortunately, sometimes a low-rank approximation to a matrix will do nearly as well as the the matrix itself; sometimes even better! In this chapter, we will learn about how to find the best rank-k approximation to a given matrix, the rank-k matrix that is closest to the given matrix. There are a variety of applications, including two analytical methods, one called *principal components analysis* (PCA) and the other called *latent semantic indexing*.

11.1.2 Matrix norm

In order to define the problem of finding the rank-k matrix closest to a given matrix, we need to define a distance for matrices. For vectors, distance is given by the norm, which is in turn defined by the inner product. For vectors over $\mathbb{R}$, we defined inner product to be dot-product. We saw in Chapter 10 that the inner product for complex numbers was somewhat different. For this chapter, we will leave aside complex numbers and return to vectors and matrices over $\mathbb{R}$. Our inner product, therefore, is once again just dot-product, and so the norm of a vector is simply the square root of the sum of the squares of its entries. But how can we define the norm of a matrix?

Perhaps the most natural matrix norm arises from interpreting a matrix A as a vector. An $m \times n$ matrix is represented by an mn-vector, i.e. the vector has one entry for each entry of the matrix. The norm of a vector is the square root of the sum of the entries, and so that is how we measure the norm of a matrix A. This norm is called the *Frobenius* norm:

$$||A||_F = \sqrt{\sum_i \sum_j A[i,j]^2}$$

Lemma 11.1.1: The square of the Frobenius norm of A equals the sum of the squares of the rows of A.

Proof

Suppose A is an $m \times n$ matrix. Write A in terms of its rows: $A = \left[\begin{array}{c} \boldsymbol{a}_1 \\ \hline \vdots \\ \hline \boldsymbol{a}_m \end{array}\right]$. For each row label i,

$$\|\boldsymbol{a}_i\|^2 = \boldsymbol{a}_i[1]^2 + \boldsymbol{a}_i[2]^2 + \cdots + \boldsymbol{a}_i[n]^2 \tag{11.1}$$

We use this equation to substitute in the definition of Frobenius norm:

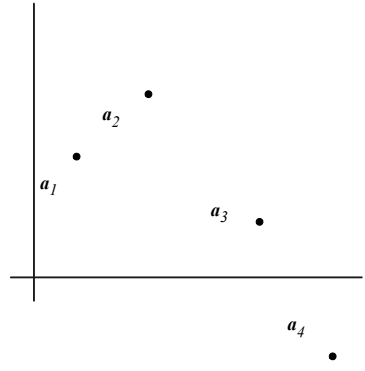
$$\begin{aligned}\|A\|_F^2 &= \left(A[1,1]^2 + A[1,2]^2 + \cdots + A[1,n]^2\right) + \cdots + \left(A[m,1]^2 + A[m,2]^2 + \cdots + A[m,n]^2\right) \\ &= \|\boldsymbol{a}_1\|^2 + \cdots \|\boldsymbol{a}_m\|^2\end{aligned}$$

□

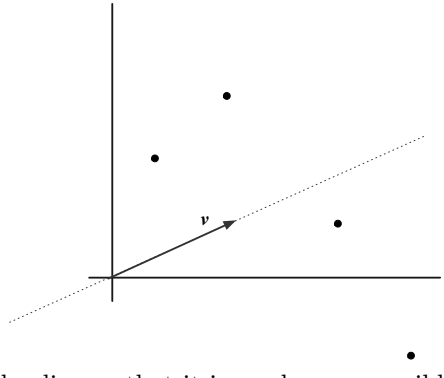
The analogous statement holds for columns as well.

11.2 The *trolley-line-location* problem

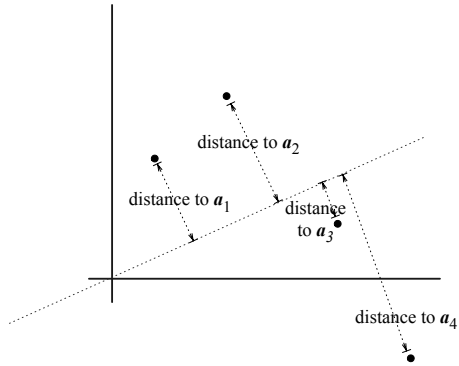
We start with a problem that is in a sense the opposite of the fire-engine problem. I call it the *trolley-line-location* problem. Given the locations of m houses, specified as vectors $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$,



we must choose where to locate a trolley-line. The trolley-line is required to go through downtown (which we represent as the origin) and is required to be a straight line.



The goal is to locate the trolley-line so that it is as close as possible to the m houses.



So far the problem is not fully specified. If there is only one house (i.e. one vector $\boldsymbol{a}_1$) then the solution is obvious: build a trolley-line along the line going through the origin and $\boldsymbol{a}_1$. In this case, the distance from the one house to the trolley-line is zero. If there are many vectors $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$, how should we measure the distance from these vectors to the trolley-line? Each vector $\boldsymbol{a}_i$ has its own distance d_i from the trolley-line—how should we combine the numbers $[d_1, \ldots, d_m]$ to get a single number to minimize? As in least squares, we minimize the norm of the vector $[d_1, \ldots, d_m]$. This is equivalent to minimizing the square of the norm of this vector, i.e. $d_1^2 + \cdots + d_m^2$.

And in what form should the output line be specified? By a unit-norm vector $\boldsymbol{v}$. The line of the trolley-line is then Span $\{\boldsymbol{v}\}$.

Computational Problem 11.2.1: *Trolley-Line-location problem:*

- *input:* vectors $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$
- *output:* a unit vector $\boldsymbol{v}$ that minimizes

$$(\text{distance from } \boldsymbol{a}_1 \text{ to Span } \{\boldsymbol{v}\})^2 + \cdots + (\text{distance from } \boldsymbol{a}_m \text{ to Span } \{\boldsymbol{v}\})^2 \qquad (11.2)$$

11.2.1 Solution to the trolley-line-location problem

For each vector $\boldsymbol{a}_i$, write $\boldsymbol{a}_i = \boldsymbol{a}_i^{\parallel \boldsymbol{v}} + \boldsymbol{a}_i^{\perp \boldsymbol{v}}$ where $\boldsymbol{a}_i^{\parallel \boldsymbol{v}}$ is the projection of $\boldsymbol{a}_i$ along $\boldsymbol{v}$ and $\boldsymbol{a}_i^{\perp \boldsymbol{v}}$ is the projection orthogonal to $\boldsymbol{v}$. Then

$$\begin{aligned} \boldsymbol{a}_1^{\perp \boldsymbol{v}} &= \boldsymbol{a}_1 - \boldsymbol{a}_1^{\parallel \boldsymbol{v}} \\ &\vdots \\ \boldsymbol{a}_m^{\perp \boldsymbol{v}} &= \boldsymbol{a}_m - \boldsymbol{a}_m^{\parallel \boldsymbol{v}} \end{aligned}$$

By the Pythagorean Theorem,

$$\begin{aligned} \|\boldsymbol{a}_1^{\perp \boldsymbol{v}}\|^2 &= \|\boldsymbol{a}_1\|^2 - \|\boldsymbol{a}_1^{\parallel \boldsymbol{v}}\|^2 \\ &\vdots \\ \|\boldsymbol{a}_m^{\perp \boldsymbol{v}}\|^2 &= \|\boldsymbol{a}_m\|^2 - \|\boldsymbol{a}_m^{\parallel \boldsymbol{v}}\|^2 \end{aligned}$$

Since the distance from $\boldsymbol{a}_i$ to Span $\{\boldsymbol{v}\}$ is $\|\boldsymbol{a}_i^{\perp \boldsymbol{v}}\|$, we have

$$\begin{aligned}(\text{distance from } \boldsymbol{a}_1 \text{ to Span } \{\boldsymbol{v}\})^2 &= \|\boldsymbol{a}_1\|^2 - \|\boldsymbol{a}_1^{\|\boldsymbol{v}}\|^2 \\ &\vdots \\ (\text{distance from } \boldsymbol{a}_m \text{ to Span } \{\boldsymbol{v}\})^2 &= \|\boldsymbol{a}_m\|^2 - \|\boldsymbol{a}_m^{\|\boldsymbol{v}}\|^2\end{aligned}$$

Adding vertically, we obtain

$$\begin{aligned}\textstyle\sum_i (\text{distance from } \boldsymbol{a}_i \text{ to Span } \{\boldsymbol{v}\})^2 &= \|\boldsymbol{a}_1\|^2 + \cdots + \|\boldsymbol{a}_m\|^2 - \left(\|\boldsymbol{a}_1^{\|\boldsymbol{v}}\|^2 + \cdots + \|\boldsymbol{a}_m^{\|\boldsymbol{v}}\|^2\right) \\ &= \|A\|_F^2 - \left(\|\boldsymbol{a}_1^{\|\boldsymbol{v}}\|^2 + \cdots + \|\boldsymbol{a}_m^{\|\boldsymbol{v}}\|^2\right)\end{aligned}$$

where A is the matrix whose rows are $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$, by Lemma 11.1.1.

Using the fact that $\boldsymbol{a}_i^{\|\boldsymbol{v}} = \langle \boldsymbol{a}_i, \boldsymbol{v} \rangle \ \boldsymbol{v}$ because $\boldsymbol{v}$ is a norm-one vector, we have $\|\boldsymbol{a}_i^{\|\boldsymbol{v}}\|^2 = \langle \boldsymbol{a}_i, \boldsymbol{v} \rangle^2$, so

$$\textstyle\sum_i (\text{distance from } \boldsymbol{a}_i \text{ to Span } \{\boldsymbol{v}\})^2 = ||A||_F^2 - \left(\langle \boldsymbol{a}_1, \boldsymbol{v} \rangle^2 + \langle \boldsymbol{a}_2, \boldsymbol{v} \rangle^2 + \cdots + \langle \boldsymbol{a}_m, \boldsymbol{v} \rangle^2\right) \tag{11.3}$$

Next, we show that $\left(\langle \boldsymbol{a}_1, \boldsymbol{v} \rangle^2 + \langle \boldsymbol{a}_2, \boldsymbol{v} \rangle^2 + \cdots + \langle \boldsymbol{a}_m, \boldsymbol{v} \rangle^2\right)$ can be replaced by $\|A\boldsymbol{v}\|^2$. By our dot-product interpretation of matrix-vector multiplication,

$$\left[\begin{array}{c} \boldsymbol{a}_1 \\ \hline \vdots \\ \hline \boldsymbol{a}_m \end{array}\right] \left[\begin{array}{c} \\ \boldsymbol{v} \\ \\ \end{array}\right] = \left[\begin{array}{c} \langle \boldsymbol{a}_1, \boldsymbol{v} \rangle \\ \vdots \\ \langle \boldsymbol{a}_m, \boldsymbol{v} \rangle \end{array}\right] \tag{11.4}$$

so

$$\|A\boldsymbol{v}\|^2 = \left(\langle \boldsymbol{a}_1, \boldsymbol{v} \rangle^2 + \langle \boldsymbol{a}_2, \boldsymbol{v} \rangle^2 + \cdots + \langle \boldsymbol{a}_m, \boldsymbol{v} \rangle^2\right)$$

Substituting into Equation 11.3, we obtain

$$\textstyle\sum_i (\text{distance from } \boldsymbol{a}_i \text{ to Span } \{\boldsymbol{v}\})^2 = ||A||_F^2 - \|A\boldsymbol{v}\|^2 \tag{11.5}$$

Therefore the best vector $\boldsymbol{v}$ is a unit vector that maximizes $||A\boldsymbol{v}||^2$ (equivalently, maximizes $||A\boldsymbol{v}||$). We now know a solution, at least in principle, for the trolley-line-location problem, Computational Problem 11.2.1:

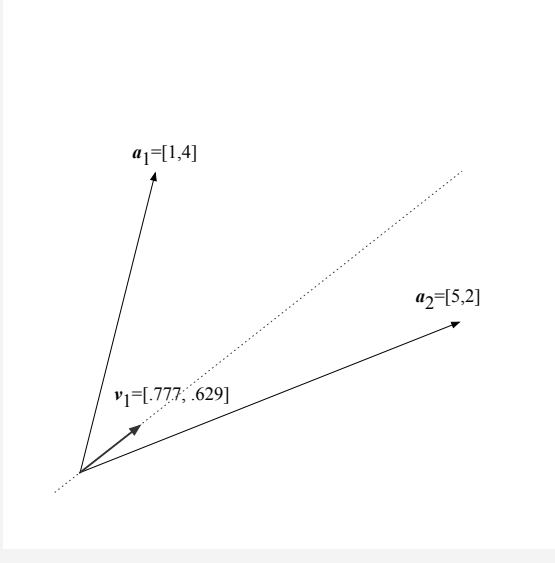
```
def trolley_line_location(A):
    Given a matrix A, find the vector v_1
    minimizing Σ_i (distance from row i of A to Span {v_1})^2
    v_1 = arg max{||Av|| : ||v|| = 1}
    σ_1 = ||Av_1||
    return v_1
```

The arg max notation means the thing (in this case the norm-one vector $\boldsymbol{v}$) that results in the largest value of $\|A\boldsymbol{v}\|$.

So far, this is a solution only in *principle* since we have not specified how to actually compute $\boldsymbol{v}_1$. In Chapter 12, we will describe a method for approximating $\boldsymbol{v}_1$.

Definition 11.2.2: We refer to σ_1 as the *first singular value* of A, and we refer to $\boldsymbol{v}_1$ as the *first right singular vector*.

Example 11.2.3: Let $A = \begin{bmatrix} 1 & 4 \\ 5 & 2 \end{bmatrix}$, so $\boldsymbol{a}_1 = [1, 4]$ and $\boldsymbol{a}_2 = [5, 2]$. In this case, a unit vector maximizing $||A\boldsymbol{v}||$ is $\boldsymbol{v}_1 \approx \begin{bmatrix} 0.78 \\ 0.63 \end{bmatrix}$. We use σ_1 to denote $||A\boldsymbol{v}_1||$, which is about 6.1:



We have proved the following theorem, which states that `trolley_line_location`(A) finds the closest vector space.

Theorem 11.2.4: Let A be an $m \times n$ matrix over $\mathbb{R}$ with rows $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$. Let $\boldsymbol{v}_1$ be the first right singular vector of A. Then Span $\{\boldsymbol{v}_1\}$ is the one-dimensional vector space $\mathcal{V}$ that minimizes

$$(\text{distance from } \boldsymbol{a}_1 \text{ to } \mathcal{V})^2 + \cdots + (\text{distance from } \boldsymbol{a}_m \text{ to } \mathcal{V})^2$$

How close is the closest vector space to the rows of A?

Lemma 11.2.5: The minimum sum of squared distances is $||A||_F^2 - \sigma_1^2$.

Proof

According to Equation 11.5, the squared distance is $\sum_i ||\boldsymbol{a}_i||^2 - \sum_i ||\boldsymbol{a}_i^{\parallel \boldsymbol{v}}||^2$. By Lemma 11.1.1, the first sum is $||A||_F^2$. The second sum is the square of the quantity $||A\boldsymbol{v}_1||$, a quantity we have named σ_1. □

Example 11.2.6: Continuing with Example 11.2.3 (Page 534), we calculate the sum of squared distances.

First we find the projection of $\boldsymbol{a}_1$ orthogonal to $\boldsymbol{v}_1$:

$$\begin{aligned} \boldsymbol{a}_1 - \langle \boldsymbol{a}_1, \boldsymbol{v}_1 \rangle \, \boldsymbol{v}_1 &\approx [1,4] - (1 \cdot 0.78 + 4 \cdot 0.63)[0.78, 0.63] \\ &\approx [1,4] - 3.3\,[0.78, 0.63] \\ &\approx [-1.6, 1.9] \end{aligned}$$

The norm of this vector, about 2.5, is the distance from $\boldsymbol{a}_1$ to Span $\{\boldsymbol{v}_1\}$.

Next we find the projection of $\boldsymbol{a}_2$ orthogonal to $\boldsymbol{v}_1$:

$$\begin{aligned} \boldsymbol{a}_2 - \langle \boldsymbol{a}_1, \boldsymbol{v}_1 \rangle \, \boldsymbol{v}_1 &\approx [5,2] - (5 \cdot 0.78 + 2 \cdot 0.63)[0.78, 0.63] \\ &\approx [5,2] - 5.1\,[0.78, 0.63] \\ &\approx [1, -1.2] \end{aligned}$$

The norm of this vector, about 1.6, is the distance from $\boldsymbol{a}_2$ to Span $\{\boldsymbol{v}_1\}$.

Thus the sum of squared distances is about $2.5^2 + 1.6^2$, which is about 8.7.

According to Lemma 11.2.5, the sum of squared distances should be $||A||_F^2 - \sigma_1^2$. The squared Frobenius of A is $1^2 + 4^2 + 5^2 + 2^2 = 46$, and the first singular value is about 6.1, so $||A||_F^2 - \sigma_1^2$ is about 8.7. Lemma 11.2.5 is correct in this example!

Warning: We are measuring the error by distance to the subspace. The norm of a vector treats every entry equally. For this technique to be relevant, the units for the entries need to be appropriate.

Example 11.2.7: Let $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{100}$ be the voting records for US Senators (same data as you used in the politics lab). These are 46-vectors with ± 1 entries.

We find the unit-norm vector $\boldsymbol{v}$ that minimizes least-squares distance from $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{100}$ to Span $\{\boldsymbol{v}\}$, and we plot the projection along $\boldsymbol{v}$ of each of these vectors:

The results are not so meaningful. Moderates and conservatives have very similar projections:

Snowe	0.106605199	moderate Republican from Maine
Lincoln	0.106694552	moderate Republican from Rhode Island
Collins	0.107039376	moderate Republican from Maine
Crapo	0.107259689	conservative moderate Republican from Idaho
Vitter	0.108031374	conservative moderate Republican from Louisiana

There is that one outlier, way off to the left. That's Russ Feingold.
We'll later return to this data and try again....

11.2.2 Rank-one approximation to a matrix

Building on our solution to the trolley-line-location problem, we will obtain a solution to another computational problem: *finding the best rank-one approximation to a given matrix*. In finding the best k-sparse approximation to a vector (Chapter 10), "best" meant "closest to the original vector", where distance between vectors is measured in the usual way, by the norm. Here we would like to similarly measure the distance between the original matrix and its approximation. For that, we need a norm for matrices.

11.2.3 The best rank-one approximation

We are now in a position to define the problem *rank-one approximation*.

Computational Problem 11.2.8: *Rank-one approximation:*

- *input:* a nonzero matrix A
- *output:* the rank-one matrix $\tilde{A}$ that is closest to A according to Frobenius norm

Equivalently, the goal is to find the rank-one matrix $\tilde{A}$ that minimizes $||A - \tilde{A}||_F$:

$$\tilde{A} = \arg\min\{||A - B||_F \ : \ B \text{ has rank one}\}$$

Suppose we have some rank-one matrix $\tilde{A}$. How close is it to A? Let's look at the squared distance between A and $\tilde{A}$. By Lemma 11.1.1,

$$\|A - \tilde{A}\|_F^2 = \|\text{row 1 of } \ A - \tilde{A}\|^2 + \cdots + \|\text{row } m \text{ of } \ A - \tilde{A}\|^2 \tag{11.6}$$

This tells us that, in order to minimize the distance to A, we should choose each row of $\tilde{A}$ to be as close as possible to the corresponding row of A. On the other hand, we require that $\tilde{A}$ have rank one. That is, we require that, for some vector $\boldsymbol{v}$, each row of $\tilde{A}$ lies in Span $\{\boldsymbol{v}\}$. To minimize the distance to A, therefore, once $\boldsymbol{v}$ has been chosen, we should choose $\tilde{A}$ thus:

$$\tilde{A} = \begin{bmatrix} \text{vector in Span } \{\boldsymbol{v}\} \text{ closest to } \boldsymbol{a}_1 \\ \hline \vdots \\ \hline \text{vector in Span } \{\boldsymbol{v}\} \text{ closest to } \boldsymbol{a}_m \end{bmatrix} \tag{11.7}$$

Accordingly, for $i = 1, \ldots, m$,

$$\|\text{row } i \text{ of } \; A - \tilde{A}\|_F = \text{distance from } \boldsymbol{a}_i \text{ to Span } \{\boldsymbol{v}\}$$

Combining with Equation 11.6 tells us that, once we have chosen $\boldsymbol{v}$, the best approximation $\tilde{A}$ satisfies

$$\|A - \tilde{A}\|^2 = (\text{distance from } \boldsymbol{a}_1 \text{ to Span } \{\boldsymbol{v}\})^2 + \cdots + (\text{distance from } \boldsymbol{a}_m \text{ to Span } \{\boldsymbol{v}\})^2$$

Theorem 11.2.4 tells us that, to minimize the sum of squared distances to Span $\{\boldsymbol{v}\}$, we should choose $\boldsymbol{v}$ to be $\boldsymbol{v}_1$, the first right singular value. By Lemma 11.2.5, the sum of squared distances is then $||A||_F^2 - \sigma_1^2$. We therefore obtain

Theorem 11.2.9: The rank-one matrix $\tilde{A}$ that minimizes $\|A - \tilde{A}\|_F$ is

$$\tilde{A} = \begin{bmatrix} \text{vector in Span } \{\boldsymbol{v}_1\} \text{ closest to } \boldsymbol{a}_1 \\ \hline \vdots \\ \hline \text{vector in Span } \{\boldsymbol{v}_1\} \text{ closest to } \boldsymbol{a}_m \end{bmatrix} \tag{11.8}$$

and, for this choice, $\|A - \tilde{A}\|_F^2 = \|A\|_F^2 - \sigma_1^2$.

11.2.4 An expression for the best rank-one approximation

Equation 11.8 specifies $\tilde{A}$ but there is a slicker way of writing it. The vector in Span $\{\boldsymbol{v}_1\}$ closest to $\boldsymbol{a}_i$ is $\boldsymbol{a}_i^{\|\boldsymbol{v}_1}$, the projection of $\boldsymbol{a}_i$ onto Span $\{\boldsymbol{v}_1\}$. Using the formula $\boldsymbol{a}_i^{\|\boldsymbol{v}_1} = \langle \boldsymbol{a}_i, \boldsymbol{v}_1 \rangle \boldsymbol{v}_1$, we obtain

$$\tilde{A} = \begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{v}_1 \rangle \boldsymbol{v}_1^T \\ \hline \vdots \\ \hline \langle \boldsymbol{a}_m, \boldsymbol{v}_1 \rangle \boldsymbol{v}_1^T \end{bmatrix}$$

Using the linear-combinations interpretation of vector-matrix multiplication, we can write this as an outer product of two vectors:

$$\tilde{A} = \begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{v}_1 \rangle \\ \vdots \\ \langle \boldsymbol{a}_m, \boldsymbol{v}_1 \rangle \end{bmatrix} \begin{bmatrix} & \boldsymbol{v}_1^T & \end{bmatrix} \tag{11.9}$$

By Equation 11.4, the first vector in the outer product can be written as $A\boldsymbol{v}_1$. Substituting into Equation 11.9, we obtain

$$\tilde{A} = \begin{bmatrix} \\ A\boldsymbol{v}_1 \\ \\ \end{bmatrix} \begin{bmatrix} & \boldsymbol{v}_1^T & \end{bmatrix} \tag{11.10}$$

We have defined σ_1 to be the norm $\|A\boldsymbol{v}_1\|$. We define $\boldsymbol{u}_1$ to be the norm-one vector such that $\sigma_1 \boldsymbol{u}_1 = A\boldsymbol{v}_1$. Then we can rewrite Equation 11.10 as

$$\tilde{A} = \sigma_1 \begin{bmatrix} \\ \boldsymbol{u}_1 \\ \\ \end{bmatrix} \begin{bmatrix} & \boldsymbol{v}_1^T & \end{bmatrix} \tag{11.11}$$

Definition 11.2.10: The *first left singular vector* of A is defined to be the vector $\boldsymbol{u}_1$ such that $\sigma_1 \boldsymbol{u}_1 = A\boldsymbol{v}_1$, where σ_1 and $\boldsymbol{v}_1$ are, respectively, the first singular value and the first right singular vector.

Theorem 11.2.11: The best rank-one approximation to A is $\sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1^T$ where σ_1 is the first singular value, $\boldsymbol{u}_1$ is the first left singular vector, and $\boldsymbol{v}_1$ is the first right singular vector of A.

Example 11.2.12: We saw in Example 11.2.3 (Page 534) that, for the matrix $A = \begin{bmatrix} 1 & 4 \\ 5 & 2 \end{bmatrix}$, the first right singular vector is $\boldsymbol{v}_1 \approx \begin{bmatrix} 0.78 \\ 0.63 \end{bmatrix}$ and the first singular value σ_1 is about 6.1. The first left singular vector is $\boldsymbol{u}_1 \approx \begin{bmatrix} 0.54 \\ 0.84 \end{bmatrix}$, meaning $\sigma_1 \boldsymbol{u}_1 = A\boldsymbol{v}_1$.

We then have

$$\begin{aligned} \tilde{A} &= \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1^T \\ &\approx 6.1 \begin{bmatrix} 0.54 \\ 0.84 \end{bmatrix} \begin{bmatrix} 0.78 & 0.63 \end{bmatrix} \\ &\approx \begin{bmatrix} 2.6 & 2.1 \\ 4.0 & 3.2 \end{bmatrix} \end{aligned}$$

Then

$$\begin{aligned} A - \tilde{A} &\approx \begin{bmatrix} 1 & 4 \\ 5 & 2 \end{bmatrix} - \begin{bmatrix} 2.6 & 2.1 \\ 4.0 & 3.2 \end{bmatrix} \\ &\approx \begin{bmatrix} -1.56 & 1.93 \\ 1.00 & -1.23 \end{bmatrix} \end{aligned}$$

so the squared Frobenius norm of $A - \tilde{A}$ is

$$1.56^2 + 1.93^2 + 1^2 + 1.23^2 \approx 8.7$$

Does this agree with Theorem 11.2.9? That theorem states that $||A - \tilde{A}||_F^2 = ||A||_F^2 - \sigma_1^2$, which we calculated to be about 8.7 in Example 11.2.6 (Page 535).

11.2.5 The closest one-dimensional affine space

When we defined the trolley-line-location problem in Section 11.2, we stipulated that the trolley-line go through the origin. This was necessary in order that the trolley-line-location problem correspond to finding the closest one-dimensional *vector space*. A one-dimensional vector space is a line through the origin. Recall from Chapter 3 that an arbitrary line (one not necessarily passing through the origin) is an *affine* space.

We can adapt the trolley-line-location techniques to solve this problem as well. Given points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$, we choose a point $\bar{\boldsymbol{a}}$ and translate each of the input points by subtracting $\bar{\boldsymbol{a}}$:

$$\boldsymbol{a}_1 - \bar{\boldsymbol{a}}, \ldots, \boldsymbol{a}_m - \bar{\boldsymbol{a}}$$

We find the one-dimensional vector space closest to these translated points, and then translate that vector space by adding back $\bar{\boldsymbol{a}}$.

Whether the procedure we have just described correctly finds the *closest* affine space depends on how $\bar{\boldsymbol{a}}$ is chosen. The best choice of $\bar{\boldsymbol{a}}$, quite intuitively, is the *centroid* of the input points, the vector

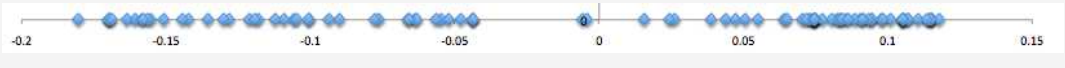
$$\bar{\boldsymbol{a}} = \frac{1}{m}(\boldsymbol{a}_1 + \cdots + \boldsymbol{a}_m)$$

We omit the proof.

Finding the centroid of given points and then translating those points by subtracting off the centroid is called *centering* the points.

Example 11.2.13: We revisit Example 11.2.7 (Page 535), in which $\boldsymbol{a}_1, \ldots, a_{100}$ were the voting records for US Senators. This time, we *center* the data, and only then find the closest one-dimensional vector space Span $\{\boldsymbol{v}_1\}$.

Now projection along $\boldsymbol{v}$ gives a better spread:



Only three of the senators to the left of the origin are Republican:

```
>>> {r for r in senators if is_neg[r] and is_Repub[r]}
{'Collins', 'Snowe', 'Chafee'}
```

and these are perhaps the most moderate Republicans in the Senate at that time. Similarly, only three of the senators to the right of the origin are Democrat.

11.3 Closest dimension-k vector space

The generalization of the trolley-line-location problem to higher dimensions is this:

Computational Problem 11.3.1: *closest low-dimensional subspace:*

- *input:* Vectors $\boldsymbol{a}_1, \ldots \boldsymbol{a}_m$ and positive integer k
- *output:* basis for the k-dimensional vector space V_k that minimizes

$$\sum_i (\text{distance from } \boldsymbol{a}_i \text{ to } V_k)^2$$

The trolley-line-location problem is merely the special case in which $k = 1$. In this special case, we seek the basis for a one-dimensional vector space. The solution, embodied in `trolley_line_location`(A), is a basis consisting of the unit-norm vector $\boldsymbol{v}$ that maximizes $||A\boldsymbol{v}||$ where A is the matrix whose rows are $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$.

11.3.1 A *Gedanken* algorithm to find the singular values and vectors

There is a natural generalization of this algorithm in which an *orthonormal* basis is sought. In the i^{th} iteration, the vector $\boldsymbol{v}$ selected is the one that maximizes $||A\boldsymbol{v}||$ subject to being orthogonal to all previously selected vectors:

- Let $\boldsymbol{v}_1$ be the norm-one vector $\boldsymbol{v}$ maximizing $||A\boldsymbol{v}||$,
- let $\boldsymbol{v}_2$ be the norm-one vector $\boldsymbol{v}$ orthogonal to $\boldsymbol{v}_1$ that maximizes $||A\boldsymbol{v}||$,
- let $\boldsymbol{v}_3$ be the norm-one vector $\boldsymbol{v}$ orthogonal to $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ that maximizes $||A\boldsymbol{v}||$,

and so on.

Here is the same algorithm in pseudocode:

Given an $m \times n$ matrix A, find vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\text{rank } A}$ such that, for $k = 1, 2, \ldots, \text{rank } (A)$, the k-dimensional subspace $\mathcal{V}$ that minimizes $\sum_i (\text{distance from row } i \text{ of } A \text{ to } \mathcal{V}_k)^2$ is Span $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$

```
def find_right_singular_vectors(A):
  for i = 1, 2, ...
    v_i = arg max{||Av|| : ||v|| = 1, v is orthogonal to v_1, v_2, ... v_{i-1}}
    σ_i = ||Av_i||
  until Av = 0 for every vector v orthogonal to v_1, ..., v_i
  let r be the final value of the loop variable i.
  return [v_1, v_2, ..., v_r]
```

Like the procedure `trolley_line_location(A)`, so far this procedure is not fully specified since we have not said how to compute each arg max. Indeed, this is not by any stretch the best algorithm for computing these vectors, but it is very helpful indeed to think about. It is a *Gedanken* algorithm.

Definition 11.3.2: The vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_r$ are the *right singular vectors* of A, and the corresponding real numbers $\sigma_1, \sigma_2, \ldots, \sigma_r$ are the *singular values* of A.

11.3.2 Properties of the singular values and right singular vectors

The following property is rather obvious.

Proposition 11.3.3: The right singular vectors are orthonormal.

Proof

In iteration i, $\boldsymbol{v}_i$ is chosen from among vectors that have norm one and are orthogonal to $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{i-1}$. □

Example 11.3.4: We revisit the matrix $A = \begin{bmatrix} 1 & 4 \\ 5 & 2 \end{bmatrix}$ of Examples 11.2.3, 11.2.6 and 11.2.12. We saw that the first right singular vector is $\boldsymbol{v}_1 \approx \begin{bmatrix} 0.78 \\ 0.63 \end{bmatrix}$ and the first singular value σ_1 is about 6.1. The second right singular vector must therefore be chosen among the vectors orthogonal to $\begin{bmatrix} 0.78 \\ 0.63 \end{bmatrix}$. It turns out to be $\begin{bmatrix} 0.63 \\ -0.78 \end{bmatrix}$. The corresponding singular value is $\sigma_2 \approx 2.9$.

The vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ vectors are obviously orthogonal. Notice that σ_2 is smaller than σ_1. It cannot be greater since the second maximization is over a smaller set of candidate solutions.

Since the vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are orthogonal and nonzero, we know they are linearly independent, and therefore that they span $\mathbb{R}^2$.

Here's another nearly obvious property.

Proposition 11.3.5: The singular values are nonnegative and in descending order.

Proof

Since each singular value is the norm of a vector, it is nonnegative. For each $i > 1$, the set of vectors from which $\boldsymbol{v}_i$ is chosen is a subset of the set of vectors from which $\boldsymbol{v}_{i-1}$ is chosen, so the maximum achieved in iteration i is no greater than the maximum achieved in iteration $i-1$. This shows $\sigma_i \leq \sigma_{i-1}$. □

Now for something not at all obvious—a rather surprising fact that is at the heart of the notion of Singular Value Decomposition.

Lemma 11.3.6: Every row of A is in the span of the right singular vectors.

Proof

Let $\mathcal{V} = \text{Span}\,\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r\}$. Let $\mathcal{V}^o$ be the annihilator of $\mathcal{V}$, and recall that $\mathcal{V}^o$ consists of all vectors orthogonal to $\mathcal{V}$. By the loop termination condition, for any vector $\boldsymbol{v}$ in $\mathcal{V}^o$, the product $A\boldsymbol{v}$ is the zero vector, so the rows of A are orthogonal to $\boldsymbol{v}$. The annihilator of the annihilator $(\mathcal{V}^o)^*$ consists of all vectors orthogonal to $\mathcal{V}^o$, so the rows of A are in $(\mathcal{V}^o)^*$. Theorem 6.5.15, the Annihilator Theorem, states that $(\mathcal{V}^o)^o$ equals $\mathcal{V}$. This shows that the rows of A are in $\mathcal{V}$. □

11.3.3 The singular value decomposition

Lemma 11.3.6 tells us that each row $\boldsymbol{a}_i$ of A is a linear combination of the right singular vectors:

$$\boldsymbol{a}_i = \sigma_{i1}\,\boldsymbol{v}_1 + \cdots + \sigma_{ir}\,\boldsymbol{v}_r$$

Since $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$ are orthonormal, the j^{th} summand $\sigma_{ij}\,\boldsymbol{v}_j$ is the projection of $\boldsymbol{a}_i$ along the j^{th} right singular vector $\boldsymbol{v}_j$, and the coefficient σ_{ij} is just the inner product of $\boldsymbol{a}_i$ and $\boldsymbol{v}_j$:

$$\boldsymbol{a}_i = \langle \boldsymbol{a}_i, \boldsymbol{v}_1 \rangle\ \boldsymbol{v}_1 + \cdots + \langle \boldsymbol{a}_i, \boldsymbol{v}_r \rangle\ \boldsymbol{v}_r$$

Using the dot-product definition of vector-matrix multiplication, we write this as

$$\boldsymbol{a}_i = \begin{bmatrix} \langle \boldsymbol{a}_i, \boldsymbol{v}_1 \rangle & \cdots & \langle \boldsymbol{a}_i, \boldsymbol{v}_r \rangle \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1^T \\ \hline \vdots \\ \hline \boldsymbol{v}_r^T \end{bmatrix}$$

Combining all these equations and using the vector-matrix definition of matrix-matrix multiplication, we can express A as a matrix-matrix product:

$$\begin{bmatrix} \boldsymbol{a}_1^T \\ \hline \boldsymbol{a}_2^T \\ \hline \vdots \\ \hline \boldsymbol{a}_m^T \end{bmatrix} = \begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{v}_1 \rangle & \cdots & \langle \boldsymbol{a}_1, \boldsymbol{v}_r \rangle \\ \langle \boldsymbol{a}_2, \boldsymbol{v}_1 \rangle & \cdots & \langle \boldsymbol{a}_2, \boldsymbol{v}_r \rangle \\ & \vdots & \\ \langle \boldsymbol{a}_m, \boldsymbol{v}_1 \rangle & \cdots & \langle \boldsymbol{a}_m, \boldsymbol{v}_r \rangle \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1^T \\ \hline \vdots \\ \hline \boldsymbol{v}_r^T \end{bmatrix}$$

We can further simplify this equation. The j^{th} column of the first matrix on the right-hand side is

$$\begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{v}_j \rangle \\ \langle \boldsymbol{a}_2, \boldsymbol{v}_j \rangle \\ \vdots \\ \langle \boldsymbol{a}_m, \boldsymbol{v}_j \rangle \end{bmatrix}$$

which is, by the dot-product definition of linear combinations, simply $A\boldsymbol{v}_j$. It is convenient to have a name for these vectors.

Definition 11.3.7: The vectors $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_r$ such that $\sigma_j \boldsymbol{u}_j = A\boldsymbol{v}_j$ are the *left singular vectors* of A.

Proposition 11.3.8: The left singular vectors are orthonormal.

(The proof is given in Section 11.3.10.)

Using the definition of left singular vectors, we substitute $\sigma_j \boldsymbol{u}_j$ for $A\boldsymbol{v}_j$, resulting in the equation

$$\left[\begin{array}{c|c|c} & & \\ & A & \\ & & \end{array}\right] = \left[\begin{array}{c|c|c} & & \\ \sigma_1\boldsymbol{u}_1 & \cdots & \sigma_r\boldsymbol{u}_r \\ & & \end{array}\right] \left[\begin{array}{c} \boldsymbol{v}_1^T \\ \hline \vdots \\ \hline \boldsymbol{v}_r^T \end{array}\right]$$

Finally, we separate out $\sigma_1, \ldots, \sigma_r$ into a diagonal matrix, obtaining the equation

$$\left[\begin{array}{c|c|c} & & \\ & A & \\ & & \end{array}\right] = \left[\begin{array}{c|c|c} & & \\ \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_r \\ & & \end{array}\right] \left[\begin{array}{ccc} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{array}\right] \left[\begin{array}{c} \boldsymbol{v}_1^T \\ \hline \vdots \\ \hline \boldsymbol{v}_r^T \end{array}\right] \tag{11.12}$$

Definition 11.3.9: The *singular value decomposition* of a matrix A is a factorization of A as $A = U\Sigma V^T$ in which the matrices U, Σ, and V have three properties:

Property S1: Σ is a diagonal matrix whose entries $\sigma_1, \ldots, \sigma_r$ are positive and in descending order.

Property S2: V is a column-orthogonal matrix.

Property S3: U is a column-orthogonal matrix.

(Sometimes this is called the *reduced* singular value decomposition.)
We have established the following theorem.

Theorem 11.3.10: Every matrix A over $\mathbb{R}$ has a singular value decomposition.

Proof

We have derived Equation 11.12, which shows the factorization of A into the product of

matrices U, Σ, and V. Property S1 follows from Proposition 11.3.5. Property S2 follows from Proposition 11.3.3. Property S3 follows from Proposition 11.3.8. □

The procedure `def find_right_singular_vectors`(A) is not the most efficient way to find a singular value decomposition of A. The best algorithms are beyond the scope of this book, but we provide a module `svd` with a procedure `factor(A)` that, given a Mat A, returns a triple (U, Σ, V) such that $A = U\Sigma * V^T$.

It is worth noting that the singular value decomposition has a nice symmetry under transposition. By the properties of the transpose of a matrix product (Proposition 4.11.14),

$$\begin{aligned} A^T &= (U\Sigma V^T)^T \\ &= V\Sigma^T U^T \\ &= V\Sigma U^T \end{aligned}$$

because the transpose of Σ is Σ itself.

We see that the the SVD of A^T can be obtained from the SVD of A^T just by swapping U and V.

As we will see, the SVD is important as both a mathematical concept and a computational tool. One of the people who helped develop good algorithms for computing the SVD was Gene Golub, whose license plate reflected his interest in the topic: It read "PROF SVD".

11.3.4 Using right singular vectors to find the closest k-dimensional space

Now we show how to use the right singular vectors to address Computational Problem 11.3.1. First we state how good a solution they provide.

Lemma 11.3.11: Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ be an orthonormal vector basis for a vector space $\mathcal{V}$. Then

$$(\text{distance from } \boldsymbol{a}_1 \text{ to } \mathcal{V})^2 + \cdots + (\text{distance from } \boldsymbol{a}_m \text{ to } \mathcal{V})^2$$

is $\|A\|_F^2 - \|A\boldsymbol{v}_1\|^2 - \|A\boldsymbol{v}_2\|^2 - \cdots - \|A\boldsymbol{v}_k\|^2$

Proof

The argument is the same as that given in Section 11.2.1. For each vector $\boldsymbol{a}_i$, write $\boldsymbol{a}_i = \boldsymbol{a}_i^{\parallel\mathcal{V}} + \boldsymbol{a}_i^{\perp\mathcal{V}}$. By the Pythagorean Theorem, $\|\boldsymbol{a}_1^{\perp\mathcal{V}}\|^2 = \|\boldsymbol{a}_1\|^2 - \|\boldsymbol{a}_1^{\parallel\mathcal{V}}\|^2$. Therefore the sum of squared distances is

$$\left(\|\boldsymbol{a}_1\|^2 - \|\boldsymbol{a}_1^{\parallel\mathcal{V}}\|^2\right) + \cdots + \left(\|\boldsymbol{a}_m\|^2 - \|\boldsymbol{a}_m^{\parallel\mathcal{V}}\|^2\right)$$

which equals

$$\left(\|\boldsymbol{a}_1\|^2 + \cdots + \|\boldsymbol{a}_m\|^2\right) + \left(\|\boldsymbol{a}_1^{\parallel\mathcal{V}}\|^2 + \cdots + \|\boldsymbol{a}_m^{\parallel\mathcal{V}}\|^2\right)$$

The first sum $\|\boldsymbol{a}_1\|^2 + \cdots + \|\boldsymbol{a}_m\|^2$ equals $\|A\|_F^2$. As for the second sum,

$$\begin{aligned} &\|\boldsymbol{a}_1^{\|\mathcal{V}}\|^2 + \cdots + \|\boldsymbol{a}_m^{\|\mathcal{V}}\|^2 \\ &= \left(\|\boldsymbol{a}_1^{\|\boldsymbol{v}_1}\|^2 + \cdots + \|\boldsymbol{a}_1^{\|\boldsymbol{v}_k}\|^2\right) + \cdots + \left(\|\boldsymbol{a}_m^{\|\boldsymbol{v}_1}\|^2 + \cdots + \|\boldsymbol{a}_m^{\|\boldsymbol{v}_k}\|^2\right) \\ &= \left(\langle \boldsymbol{a}_1, \boldsymbol{v}_1\rangle^2 + \cdots + \langle \boldsymbol{a}_1, \boldsymbol{v}_k\rangle^2\right) + \cdots + \left(\langle \boldsymbol{a}_m, \boldsymbol{v}_1\rangle^2 + \cdots + \langle \boldsymbol{a}_m, \boldsymbol{v}_k\rangle^2\right) \end{aligned}$$

Reorganizing all these squared inner products, we get

$$\begin{aligned} &\left(\langle \boldsymbol{a}_1, \boldsymbol{v}_1\rangle^2 + \langle \boldsymbol{a}_2, \boldsymbol{v}_1\rangle^2 + \cdots + \langle \boldsymbol{a}_m, \boldsymbol{v}_1\rangle^2\right) + \cdots + \left(\langle \boldsymbol{a}_1, \boldsymbol{v}_k\rangle^2 + \langle \boldsymbol{a}_2, \boldsymbol{v}_k\rangle^2 + \cdots + \langle \boldsymbol{a}_m, \boldsymbol{v}_k\rangle^2\right) \\ &\quad = \|A\boldsymbol{v}_1\|^2 + \cdots + \|A\boldsymbol{v}_k\|^2 \end{aligned}$$

□

The next theorem says that the span of the first k right singular vectors is the best solution.

Theorem 11.3.12: Let A be an $m \times n$ matrix, and let $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$ be its rows. Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$ be its right singular vectors, and let $\sigma_1, \ldots, \sigma_r$ be its singular values. For any positive integer $k \leq r$, Span $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ is the k-dimensional vector space $\mathcal{V}$ that minimizes

$$(\text{distance from } \boldsymbol{a}_1 \text{ to } \mathcal{V})^2 + \cdots + (\text{distance from } \boldsymbol{a}_m \text{ to } \mathcal{V})^2$$

and the minimum sum of squared distances is $\|A\|_F^2 - \sigma_1^2 - \sigma_2^2 - \cdots - \sigma_k^2$.

Proof

By Lemma 11.3.11, the sum of squared distances for the space $\mathcal{V} = \text{Span } \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ is

$$\|A\|_F^2 - \sigma_1^2 - \sigma_2^2 - \cdots - \sigma_k^2 \tag{11.13}$$

To prove that this is the minimum, we need to show that any other k-dimensional vector space $\mathcal{W}$ leads to a sum of squares that is no smaller.

Any k-dimensional vector space $\mathcal{W}$ has an orthonormal basis. Let $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_k$ be such a basis. Plugging these vectors into Lemma 11.3.11, we get that the sum of squared distances from $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$ to $\mathcal{W}$ is

$$\|A\|_F^2 - \|A\boldsymbol{w}_1\|^2 - \|A\boldsymbol{w}_2\|^2 - \cdots - \|A\boldsymbol{w}_k\|^2 \tag{11.14}$$

In order to show that $\mathcal{V}$ is the closest, we need to show that the quantity in 11.14 is no less than the quantity in 11.13. This requires that we show that $\|A\boldsymbol{w}_1\|^2 + \cdots + \|A\boldsymbol{w}_k\|^2 \leq \sigma_1^2 + \cdots + \sigma_k^2$. Let W be the matrix with columns $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_k$. Then $\|AW\|_F^2 = \|A\boldsymbol{w}_1\|^2 + \cdots + \|A\boldsymbol{w}_k\|^2$ by the column analogue of Lemma 11.1.1. We must therefore show that $\|AW\|_F^2 \leq \sigma_1^2 + \cdots + \sigma_k^2$.

By Theorem 11.3.10, A can be factored as $A = U\Sigma V^T$ where the columns of V are $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$, and where U and V are column-orthogonal and Σ is the diagonal matrix with

diagonal elements $\sigma_1, \ldots, \sigma_r$. By substitution, $\|AW\|_F^2 = \|U\Sigma V^T W\|_F^2$. Since U is column-orthogonal, multiplication by U preserves norms, so $\|U\Sigma V^T W\|_F^2 = \|\Sigma V^T W\|_F^2$.

Let X denote the matrix $V^T W$. The proof makes use of two diferent interpretations of X, in terms of columns and in terms of rows.

First, let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ denote the columns of X. For $j = 1, \ldots, k$, by the matrix-vector interpretation of matrix-matrix multiplication, $\boldsymbol{x}_j = V^T \boldsymbol{w}_j$. By the dot-product interpretation of matrix-vector multiplication, $\boldsymbol{x}_j = [\boldsymbol{v}_1 \cdot \boldsymbol{w}_j, \ldots, \boldsymbol{v}_r \cdot \boldsymbol{w}_j]$, which is the coordinate representation in terms of $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$ of the projection of $\boldsymbol{w}_j$ onto Span $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r\}$. Therefore the projection itself is $V\boldsymbol{x}_j$. The projection of a norm-one vector onto a space has norm at most one, so $\|V\boldsymbol{x}_j\| \leq 1$. Since V is a column-orthogonal matrix, $\|V\boldsymbol{x}_j\| = \|\boldsymbol{x}_j\|$, so $\boldsymbol{x}_j$ has norm at most one. This shows that $\|X\|_F^2 \leq k$.

Second, let $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_r$ denote the rows of X. For $i = 1, \ldots, r$, by the vector-matrix interpretation of matrix-matrix multiplication, $\boldsymbol{y}_i = \boldsymbol{v}_i^T W$. By the dot-product interpretation of vector-matrix multiplication, $\boldsymbol{y}_i = [\boldsymbol{v}_i \cdot \boldsymbol{w}_1, \ldots, \boldsymbol{v}_i \cdot \boldsymbol{w}_k]$, which is the coordinate representation in terms of $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_r$ of the projection of $\boldsymbol{v}_i$ onto $\mathcal{W}$. Using the same argument as before, since $\boldsymbol{v}_i$ has norm one, the coordinate representation has norm at most one. This shows that each row $\boldsymbol{y}_i$ of X has norm at most one.

Now we consider ΣX. Since Σ is a diagonal matrix with diagonal elements $\sigma_1, \ldots, \sigma_r$, it follows that row i of ΣX is σ_i times row i of X, which is $\sigma_i \boldsymbol{y}_i$. Therefore the squared Frobenius norm of ΣX is $\sigma_1^2 \|\boldsymbol{y}_1\|^2 + \cdots \sigma_r^2 \|\boldsymbol{y}_r\|^2$. How big can that quantity be?

Imagine you have k dollars to spend on r products. Product i gives you value σ_i^2 per dollar you spend on it. Your goal is to maximize the total value you receive. Since $\sigma_1 \geq \cdots \geq \sigma_r$, it makes sense to spend as much as you can on product 1, then spend as much of your remaining money on product 2, and so on. You are not allowed to spend more than one dollar on each product. What do you do? You spend one dollar on product 1, one dollar on product 2, ..., one dollar on product k, and zero dollars on the remaining products. The total value you receive is then $\sigma_1^2 + \cdots + \sigma_r^2$.

Now we formally justify this intuition. Our goal is to show that $\sigma_1^2 \|\boldsymbol{y}_1\|^2 + \cdots + \sigma_r^2 \|\boldsymbol{y}_r\|^2 \leq \sigma_1^2 + \cdots + \sigma_k^2$. We have shown that $\|\boldsymbol{y}_i\|^2 \leq 1$ for $i = 1, \ldots, k$. Since $\|X\|_F^2 \leq k$, we also know that $\|\boldsymbol{y}_1\|^2 + \cdots \|\boldsymbol{y}_r\|^2 \leq k$.

Define $\beta_i = \begin{cases} \sigma_i^2 - \sigma_r^2 & \text{if } i \leq r \\ 0 & \text{otherwise} \end{cases}$

Then $\sigma_i^2 \leq \beta_i + \sigma_r^2$ for $i = 1, \ldots, r$ (using the fact that $\sigma_1, \ldots, \sigma_r$ are in nonincreasing order).

Therefore

$$\begin{aligned}
\sigma_1^2 \|\boldsymbol{y}_1\|^2 + \cdots + \sigma_r^2 \|\boldsymbol{y}_r\|^2 &\leq (\beta_i + \sigma_r^2)\|\boldsymbol{y}_1\|^2 + \cdots + (\beta_r + \sigma_r^2)\|\boldsymbol{y}_r\|^2 \\
&= \left(\beta_1 \|\boldsymbol{y}_1\|^2 + \cdots + \beta_r \|\boldsymbol{y}_r\|^2\right) + \left(\sigma_r^2 \|\boldsymbol{y}_1\|^2 + \cdots + \sigma_r^2 \|\boldsymbol{y}_r\|^2\right) \\
&\leq (\beta_1 + \cdots + \beta_r) + \sigma_r^2 \left(\|\boldsymbol{y}_1\|^2 + \cdots + \|\boldsymbol{y}_r\|^2\right) \\
&\leq \left(\sigma_1^2 + \cdots + \sigma_k^2 - k\sigma_k^2\right) + \sigma_k^2 k \\
&= \sigma_1^2 + \cdots + \sigma_k^2
\end{aligned}$$

This completes the proof. □

11.3.5 Best rank-k approximation to A

We saw in Section 11.2.4 that the best rank-one approximation to A is $\sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1^T$. Now we generalize that formula:

Theorem 11.3.13: For $k \leq$ rank A, the best rank-at-most-k approximation to A is

$$\tilde{A} = \sigma_1 \, \boldsymbol{u}_1 \boldsymbol{v}_1^T + \cdots + \sigma_k \, \boldsymbol{u}_k \boldsymbol{v}_k^T \tag{11.15}$$

for which $\|A - \tilde{A}\|_F^2 = \|A\|_F^2 - \sigma_1^2 - \sigma_2^2 - \cdots - \sigma_k^2$.

Proof

The proof is a straightforward generalization of the argument in Section 11.2.2. Let $\tilde{A}$ be a rank-at-most-k approximation to A. By Lemma 11.1.1,

$$\|A - \tilde{A}\|_F^2 = \|\text{row 1 of } \; A - \tilde{A}\|^2 + \cdots + \|\text{row } m \text{ of } \; A - \tilde{A}\|^2 \tag{11.16}$$

For $\tilde{A}$ to have rank at most k, there must be some vector space $\mathcal{V}$ of dimension k such that every row of $\tilde{A}$ lies in $\mathcal{V}$. Once $\mathcal{V}$ has been chosen, Equation 11.16 tells us that the best choice of $\tilde{A}$ is

$$\tilde{A} = \begin{bmatrix} \text{vector in } \mathcal{V} \text{ closest to } \boldsymbol{a}_1 \\ \hline \vdots \\ \hline \text{vector in } \mathcal{V} \text{ closest to } \boldsymbol{a}_m \end{bmatrix} \tag{11.17}$$

and, for this choice,

$$\|A - \tilde{A}\|^2 = (\text{distance from } \boldsymbol{a}_1 \text{ to } \mathcal{V})^2 + \cdots + (\text{distance from } \boldsymbol{a}_m \text{ to } \mathcal{V})^2$$

Theorem 11.3.12 tells us that, to minimize the sum of squared distances to $\mathcal{V}$, we should choose $\mathcal{V}$ to be the span of the first k right singular vectors, and that the sum of squared distances is then $\|A\|_F^2 - \sigma_1^2 - \sigma_2^2 - \cdots - \sigma_k^2$.

For $i = 1, \ldots, m$, the vector in $\mathcal{V}$ closest to $\boldsymbol{a}_i$ is the projection of $\boldsymbol{a}_i$ onto $\mathcal{V}$, and

$$\begin{array}{rcccc} \text{projection of } \boldsymbol{a}_i \text{ onto } \mathcal{V} & = & \text{projection of } \boldsymbol{a}_i \text{ along } \boldsymbol{v}_1 + & \cdots & + \text{projection of } \boldsymbol{a}_i \text{ along } \boldsymbol{v}_m \\ & = & \langle \boldsymbol{a}_i, \boldsymbol{v}_1 \rangle \, \boldsymbol{v}_1 + & \cdots & + \langle \boldsymbol{a}_i, \boldsymbol{v}_k \rangle \, \boldsymbol{v}_k \end{array}$$

Substituting into Equation 11.17 and using the definition of addition of matrices gives us

$$\tilde{A} = \begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{v}_1 \rangle \, \boldsymbol{v}_1 \\ \hline \vdots \\ \hline \langle \boldsymbol{a}_m, \boldsymbol{v}_1 \rangle \, \boldsymbol{v}_1 \end{bmatrix} + \cdots + \begin{bmatrix} \langle \boldsymbol{a}_1, \boldsymbol{v}_k \rangle \, \boldsymbol{v}_k \\ \hline \vdots \\ \hline \langle \boldsymbol{a}_m, \boldsymbol{v}_k \rangle \, \boldsymbol{v}_k \end{bmatrix}$$

$$= \sigma_1 \begin{bmatrix} \\ \boldsymbol{u}_1 \\ \\ \end{bmatrix} \begin{bmatrix} \quad \boldsymbol{v}_1 \quad \end{bmatrix} + \cdots + \sigma_k \begin{bmatrix} \\ \boldsymbol{u}_k \\ \\ \end{bmatrix} \begin{bmatrix} \quad \boldsymbol{v}_k \quad \end{bmatrix}$$

□

11.3.6 Matrix form for best rank-k approximation

Equation 11.15 gives the best rank-k approximation to A as the sum of k rank-one matrices. By using the definitions of matrix-matrix and matrix-vector multiplication, one can show that Equation 11.15 can be rewritten as

$$\tilde{A} = \left[\begin{array}{c|c|c} & & \\ \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_k \\ & & \end{array}\right] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1^T \\ \hline \vdots \\ \hline \boldsymbol{v}_k^T \end{bmatrix}$$

In view of the resemblance to the singular value decomposition of A, namely $A = U\Sigma V^T$, we write

$$\tilde{A} = \tilde{U}\tilde{\Sigma}\tilde{V}^T$$

where $\tilde{U}$ consists of the first k columns of U, $\tilde{V}$ consists of the first k columns of V, and $\tilde{\Sigma}$ is the diagonal matrix whose diagonal elements are the first k diagonal elements of Σ.

11.3.7 Number of nonzero singular values is rank A

It follows from Lemma 11.3.6 that the number r of right singular vectors produced by algorithm `find_right_singular_vectors`(A) is at least the rank of A.

Let $k = \text{rank } A$. For this value of k, the best rank-k approximation to A is A itself. This shows that any subsequent singular values $\sigma_{1+\text{rank } A}, \sigma_{2+\text{rank } A}, \ldots$ must be zero. Therefore, in the algorithm `find_right_singular_vectors`(A), after rank A iterations, $A\boldsymbol{v} = \boldsymbol{0}$ for every vector $\boldsymbol{v}$ orthogonal to $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\text{rank } A}$. Thus the number r of iterations is exactly rank A.

Let's reconsider the SVD of A:

$$\begin{bmatrix} & A & \end{bmatrix} = \underbrace{\left[\begin{array}{c|c|c} & & \\ \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_r \\ & & \end{array}\right]}_{U} \underbrace{\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix}}_{\Sigma} \underbrace{\left[\begin{array}{c} \boldsymbol{v}_1^T \\ \hline \vdots \\ \hline \boldsymbol{v}_r^T \end{array}\right]}_{V^T}$$

By the vector-matrix definition of matrix-matrix multiplication, each row of A is the corresponding row of $U\Sigma$ times the matrix V^T. Therefore, by the linear-combinations definition of vector-matrix multiplication, each row of A is a linear combination of the rows of V^T. On the other hand, the rows of V^T are mutually orthogonal and nonzero, so linearly independent (Proposition 9.5.1), and there are rank A of them, so the dimension of their span is exactly rank A. Thus, by the Dimension Principle (Lemma 6.2.14), Row A equals Row V^T.

A similar argument shows that Col A equals Col U. Each column of A is U times a column of ΣV^T, and $\dim \text{Col } A = \text{rank } A = \dim \text{Col } U$, so Col A = Col U.

We summarize our findings:

Proposition 11.3.14: In the singular value decomposition $U\Sigma V^T$ of A, Col U = Col A and Row V^T = Row A.

11.3.8 Numerical rank

In fact, computing or even defining the rank of a matrix with floating-point entries is not a trivial matter. Maybe the columns of A are linearly dependent but due to floating-point error when you run `orthogonalize` on the columns you get all nonzero vectors. Or maybe the matrix you have represented in your computer is only an approximation to some "true" matrix whose entries cannot be represented exactly by floating-point numbers. The rank of the true matrix might differ from that of the represented matrix. As a practical matter, we need some useful definition of rank, and here is what is used: the *numerical rank* of a matrix is defined to the number of singular values you get before you get a singular value that is tiny.

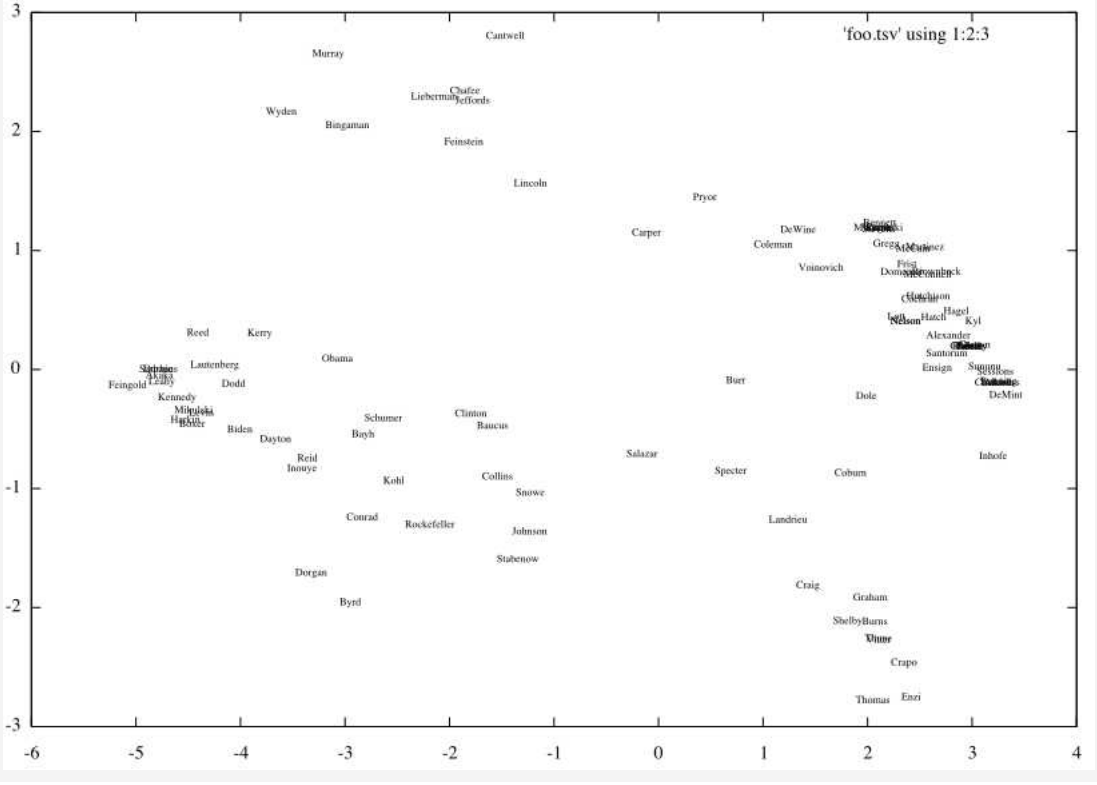
11.3.9 Closest k-dimensional affine space

To find not the closest k-dimensional vector space but the closest k-dimensional affine space, we can use the centering technique described in Section 11.2.5: find the centroid $\bar{\boldsymbol{a}}$ of the input points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$, and subtract it from each of the input points. Then find a basis $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ for the k-dimensional vector space closest to $\boldsymbol{a}_1 - \bar{\boldsymbol{a}}, \ldots, \boldsymbol{a}_m - \bar{\boldsymbol{a}}$. The k-dimensional affine space closest to the original points $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_m$ is

$$\{\bar{\boldsymbol{a}} + \boldsymbol{v} \ : \ \boldsymbol{v} \in \text{Span } \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}\}$$

The proof is omitted.

Example 11.3.15: Returning to the US Senate voting data, in Examples 11.2.7 and 11.2.13, we plotted the senators' voting records on the number line, based on their projection onto the closest one-dimensional vector. Now we can find the closest 2-dimensional affine space, and project their voting records onto these, and use the coordinates to plot the senators.



11.3.10 Proof that U is column-orthogonal

In the next proof, we use the *Cauchy-Schwartz inequality*: for vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, $|\boldsymbol{a} \cdot \boldsymbol{b}| \leq \|\boldsymbol{a}\| \, \|\boldsymbol{b}\|$. The proof is as follows: Write $\boldsymbol{b} = \boldsymbol{b}^{||\boldsymbol{a}} + \boldsymbol{b}^{\perp a}$. By the Pythagorean Theorem, $\|\boldsymbol{b}\|^2 = \|\boldsymbol{b}^{||\boldsymbol{a}}\|^2 + \|\boldsymbol{b}^{\perp a}\|^2$, so $\|\boldsymbol{b}\|^2 \geq \|\boldsymbol{b}^{||\boldsymbol{a}}\|^2 = \|\frac{\boldsymbol{b}\cdot\boldsymbol{a}}{\boldsymbol{a}\cdot\boldsymbol{a}}\boldsymbol{a}\|^2 = \left(\frac{\boldsymbol{b}\cdot\boldsymbol{a}}{\|\boldsymbol{a}\|^2}\right)^2 \|\boldsymbol{a}\|^2 = \frac{(\boldsymbol{b}\cdot\boldsymbol{a})^2}{\|\boldsymbol{a}\|^2}$, so $\|\boldsymbol{b}\|^2\|\boldsymbol{a}\|^2 \geq (\boldsymbol{b}\cdot\boldsymbol{a})^2$, which proves the inequality.

Property S3 of the singular value decomposition states that the matrix U of left singular vectors is column-orthogonal. We now prove that property.

The left singular vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_r$ have norm one by construction. We need to show that they are mutually orthogonal. We prove by induction on k that, for $i = 1, 2, \ldots, k$, the vector $\boldsymbol{u}_i$ is orthogonal to $\boldsymbol{u}_{i+1}, \ldots, \boldsymbol{u}_r$. Setting $k = r$ proves the desired result.

By definition of the singular vectors and values,

$$AV = \left[\begin{array}{c|c|c|c|c|c|c} \sigma_1\boldsymbol{u}_1 & \cdots & \sigma_{k-1}\boldsymbol{u}_{k-1} & \sigma_k\boldsymbol{u}_k & \sigma_{k+1}\boldsymbol{u}_{k+1} & \cdots & \sigma_r\boldsymbol{u}_r \end{array}\right]$$

By the inductive hypothesis, $\boldsymbol{u}_k$ is orthogonal to $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-1}$. Since $\boldsymbol{u}_k$ has norm one, $\boldsymbol{u}_k \cdot \sigma_k\boldsymbol{u}_k = \sigma_k$. Let

$$\begin{aligned} \beta_{k+1} &= \boldsymbol{u}_k \cdot \boldsymbol{u}_{k+1} \\ \beta_{k+2} &= \boldsymbol{u}_k \cdot \boldsymbol{u}_{k+2} \\ &\vdots \\ \beta_r &= \boldsymbol{u}_k \cdot \boldsymbol{u}_r \end{aligned}$$

Then

$$\boldsymbol{u}_k^T AV = \left[\begin{array}{ccccccc} 0 & \cdots & 0 & \sigma_k & \beta_{k+1} & \cdots & \beta_r \end{array}\right] \tag{11.18}$$

Our goal is to show that $\beta_{k+1}, \ldots, \beta_r$ are all zero, for this would show that $\boldsymbol{u}_k$ is orthogonal to $\boldsymbol{u}_{k+1}, \ldots, \boldsymbol{u}_r$.

Let $\boldsymbol{w} = \left[\begin{array}{ccccccc} 0 & \cdots & 0 & \sigma_k & \beta_{k+1} & \cdots & \beta_r \end{array}\right]$. Then $\|\boldsymbol{w}\|^2 = \sigma_k^2 + \beta_{k+1}^2 + \cdots + \beta_r^2$. Since V is column-orthogonal, $\|V\boldsymbol{w}\|^2 = \|\boldsymbol{w}\|^2$, so

$$\|V\boldsymbol{w}\|^2 = \sigma_k^2 + \beta_{k+1}^2 + \cdots + \beta_r^2 \tag{11.19}$$

Furthermore, since the first $k-1$ entries of $\boldsymbol{w}$ are zero, the vector $V\boldsymbol{w}$ is a linear combination of the remaining $r-(k-1)$ columns of V. Since the columns of V are mutually orthogonal, $V\boldsymbol{w}$ is orthogonal to $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{k-1}$. Let $\boldsymbol{v} = V\boldsymbol{w}/\|V\boldsymbol{w}\|$. Then $\boldsymbol{v}$ has norm one and is orthogonal to $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{k-1}$. We will show that if $\beta_{k+1}, \ldots, \beta_r$ are not all zero then $\|A\boldsymbol{v}\| > \sigma_k$, so $\boldsymbol{v}_k$ was not the unit-norm vector maximizing $\|A\boldsymbol{v}\|$ among vectors orthogonal to $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{k-1}$, a contradiction.

By Equation 11.18, $(\boldsymbol{u}_k^T AV) \cdot \boldsymbol{w} = \sigma_k^2 + \beta_{k+1}^2 + \cdots + \beta_r^2$. By the Cauchy-Schwartz Inequality, $|\boldsymbol{u}_k \cdot (AV\boldsymbol{w})| \leq \|\boldsymbol{u}_k\| \, \|AV\boldsymbol{w}\|$, so, since $\|\boldsymbol{u}_k\| = 1$, we infer $\|AV\boldsymbol{w}\| \geq \sigma_k^2 + \beta_{k+1}^2 + \cdots + \beta_r^2$. Combining this inequality with Equation 11.19, we obtain

$$\frac{\|AV\boldsymbol{w}\|}{\|V\boldsymbol{w}\|} \geq \frac{\sigma_k^2 + \beta_{k+1}^2 + \cdots + \beta_r^2}{\sqrt{\sigma_k^2 + \beta_{k+1}^2 + \cdots + \beta_r^2}} = \sqrt{\sigma_k^2 + \beta_{k+1}^2 + \cdots + \beta_r^2}$$

which is greater than σ_k if $\beta_{k+1}, \ldots, \beta_r$ are not all zero. This completes the induction step, and the proof.

11.4 Using the singular value decomposition

The singular value decomposition has emerged as a crucial tool in linear-algebra computation.

11.4.1 Using SVD to do least squares

In Section 9.8.5, we learned that the QR factorization of a matrix A can be used in solving the *least squares* problem, finding the vector $\hat{\boldsymbol{x}}$ that minimizes $\|A\boldsymbol{x} - \boldsymbol{b}\|$. However, that algorithm is applicable only if A's columns are linearly independent. Here we see that the singular value decomposition provides another method to solve least squares, a method that does not depend on A having linearly independent columns.

```
def SVD_solve(A):
  U, Σ, V = svd.factor(A)
  return VΣ^{-1}U^T b
```

Note that this algorithm seems to require multiplication by the inverse of a matrix, but the matrix is diagonal (with nonzero diagonal entries $\sigma_1, \ldots, \sigma_{\text{rank } A}$), so multiplication by its inverse amounts to applying the function $f([y_1, y_2, \ldots, y_r]) = [\sigma_1^{-1} y_1, \sigma_2^{-1} y_2, \ldots, \sigma_r^{-1} y_r]$.

To show this algorithm returns the correct solution, let $\hat{\boldsymbol{x}} = V\Sigma^{-1}U^T\boldsymbol{b}$ be the vector returned. Multiplying on the left by V^T, we get the equation

$$V^T\hat{\boldsymbol{x}} = \Sigma^{-1}U^T\boldsymbol{b}$$

Multiplying on the left by Σ, we get

$$\Sigma V^T\hat{\boldsymbol{x}} = U^T\boldsymbol{b}$$

Multiplying on the left by by U, we get

$$U\Sigma V^T\hat{\boldsymbol{x}} = UU^T\boldsymbol{b}$$

By substitution, we get

$$A\hat{\boldsymbol{x}} = UU^T\boldsymbol{b}$$

This equation should be familiar; it is similar to the equation that justified use of `QR_solve`(A) in solving a least-squares problem. By Lemma 9.8.3, $UU^T\boldsymbol{b}$ is the projection $\boldsymbol{b}^{||\text{Col } U}$ of $\boldsymbol{b}$ onto Col U. By Proposition 11.3.14, Col U = Col A, so $UU^T\boldsymbol{b}$ is the projection of $\boldsymbol{b}$ onto Col A. The Generalized Fire Engine Lemma shows therefore that $\hat{\boldsymbol{x}}$ is the correct least-squares solution.

11.5 PCA

We return for a time to the problem of analyzing data. The lab assignment deals with eigenfaces, which is the application to images of the idea of *principal component analysis*. (In this case, the images are of faces.) I'll therefore use this as an example.

Each image consists of about 32k pixels, so can be represented as a vector in $\mathbb{R}^{32000}$. Here is our first attempt at a crazy hypothesis.

> *Crazy hypothesis, version 1:* The set of face images lies in a ten-dimensional vector subspace.