Lecture 18: The SVD: Examples, Norms, Fundamental Subspaces, Compression

3.2.3. Example of the singular value decomposition.

The standard algorithm for computing the singular value decomposition differs a bit from the algorithm described in the last lecture. We know from our experiences with the normal equations for least squares problems that significant errors can be introduced when $\mathbf{A}^*\mathbf{A}$ is constructed. For practical SVD computations, one can sidestep this by using Householder transformations to create unitary matrices $\mathbf{U}$ and $\mathbf{V}$ such that $\mathbf{B} := \mathbf{U}\mathbf{A}\mathbf{V}^*$ is *bidiagonal*, i.e., $b_{jk} = 0$ unless $j = k$ or $j-1 = k$ One then applies specialized eigenvalue algorithms for computing the SVD of a bidiagonal matrix; see Trefethen & Bau (Lecture 31) for details.

While this approach has numerical advantages over the method used in our constructive proof of the SVD, it is still instructive to follow through that construction for a simple matrix, say

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}.$$

Step 1. First, form $\mathbf{A}^*\mathbf{A}$:

$$\mathbf{A}^*\mathbf{A} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

and compute its eigenvalues and (normalized) eigenvectors:

$$\lambda_1 = 3, \quad \mathbf{v}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \qquad \lambda_2 = 1, \quad \mathbf{v}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

Step 2. Set

$$\begin{aligned} \sigma_1 &= \|\mathbf{A}\mathbf{v}_1\|_2 = \sqrt{\lambda_1} = \sqrt{3}; \\ \sigma_2 &= \|\mathbf{A}\mathbf{v}_2\|_2 = \sqrt{\lambda_2} = 1. \end{aligned}$$

Step 3. Since $\sigma_1, \sigma_2 \neq 0$, we can immediately form $\mathbf{u}_1$ and $\mathbf{u}_2$:

$$\mathbf{u}_1 = \frac{1}{\sigma_1}\mathbf{A}\mathbf{v}_1 = \frac{1}{\sqrt{6}}\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}, \qquad \mathbf{u}_2 = \frac{1}{\sigma_2}\mathbf{A}\mathbf{v}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}.$$

The $1/\sigma_j$ scaling ensures that both $\mathbf{u}_1$ and $\mathbf{u}_2$ are unit vectors. We can verify that they are orthogonal:

$$\mathbf{u}_1^*\mathbf{u}_2 = \frac{1}{\sqrt{12}}\,[1 \;\; 1 \;\; 2] \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix} = 0.$$

Step 4. At this point, we have all the ingredients to build the reduced singular value decomposition:

$$\mathbf{A} = \widehat{\mathbf{U}}\widehat{\mathbf{\Sigma}}\mathbf{V}^* = \begin{bmatrix} 1/\sqrt{6} & -1/\sqrt{2} \\ 1/\sqrt{6} & 1/\sqrt{2} \\ 2/\sqrt{6} & 0 \end{bmatrix} \begin{bmatrix} \sqrt{3} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}.$$

The only additional information required to build the full SVD is the unit vector $\mathbf{u}_3$ that is orthogonal to $\mathbf{u}_1$ and $\mathbf{u}_2$. One can find such a vector by inspection:

$$\mathbf{u}_3 = \frac{1}{\sqrt{3}}\begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix}.$$

If you are naturally able to eyeball this orthogonal vector, there are any number of mechanical ways to compute $\mathbf{u}_3$, e.g., by finding a vector $\mathbf{u}_3 = [\alpha, \beta, \gamma]^T$ that satisfies the orthogonality conditions $\mathbf{u}_1^*\mathbf{u}_3 = \mathbf{u}_2^*\mathbf{u}_3 = 0$ and normalization $\mathbf{u}_3^*\mathbf{u}_3 = 1$, or using the Gram–Schmidt process. A related method is just to read $\mathbf{u}_3$ off as the third column of the $\mathbf{Q}$ factor in the full QR decomposition of $[\mathbf{u}_1\ \mathbf{u}_2]$. (Why is this so? Recall that the first n columns of the $\mathbf{Q}$ factor form a basis for the range of the factored matrix; the remaining $m-n$ columns are unit vectors that must be orthogonal to those previous columns, since $\mathbf{Q}$ is unitary.) For example:

```
>> u1 = [1;1;2]/sqrt(6);   u2 = [-1;1;0]/sqrt(2);
>> [Q,R] = qr([u1 u2])
Q =
   -0.4082    0.7071   -0.5774
   -0.4082   -0.7071   -0.5774
   -0.8165   -0.0000    0.5774
R =
   -1.0000         0
         0   -1.0000
         0         0
```

Note that the third vector in the `Q` matrix is simply $-\mathbf{u}_3$. (We could just as well replace $\mathbf{u}_3$ by $-\mathbf{u}_3$ without changing the SVD. Why?)

In conclusion, a full SVD of $\mathbf{A}$ is:

$$\mathbf{A} = \mathbf{U\Sigma V}^* = \begin{bmatrix} 1/\sqrt{6} & -1/\sqrt{2} & 1/\sqrt{3} \\ 1/\sqrt{6} & 1/\sqrt{2} & 1/\sqrt{3} \\ 2/\sqrt{6} & 0 & -1/\sqrt{3} \end{bmatrix} \begin{bmatrix} \sqrt{3} & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}.$$

3.2.4. Singular values and the matrix 2-norm.

In Lecture 2 we defined the induced matrix 2-norm

$$\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{Ax}\|_2,$$

but did not provide a simple formula for this norm in terms of the entries of $\mathbf{A}$, as we did for the induced matrix 1- and ∞-norms. With the SVD at hand, we can now derive such a formula.

Recall that the vector 2-norm (and hence the matrix 2-norm) is invariant to premultiplication by a unitary matrix, as proved in Lecture 2. Let $\mathbf{A} = \mathbf{U\Sigma V}^*$ be a singular value decomposition of $\mathbf{A}$. Thus

$$\|\mathbf{A}\|_2 = \|\mathbf{U\Sigma V}^*\|_2 = \|\mathbf{\Sigma V}^*\|_2.$$

The matrix 2-norm is also immune to a unitary matrix on the right:

$$\|\mathbf{\Sigma V}^*\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{\Sigma V}^*\mathbf{x}\|_2 = \max_{\|\mathbf{y}\|_2=1} \|\mathbf{\Sigma y}\|_2 = \|\mathbf{\Sigma}\|_2,$$

where we have set $\mathbf{y} = \mathbf{V}^*\mathbf{x}$ and noted that $\|\mathbf{y}\|_2 = \|\mathbf{V}^*\mathbf{x}\|_2 = \|\mathbf{x}\|_2$ since $\mathbf{V}^*$ is a unitary matrix. Let $p = \min\{m, n\}$. Then

$$\|\mathbf{\Sigma y}\|_2^2 = \sum_{j=1}^{p} \sigma_j^2 y_j^2,$$

which is maximized over $\|\mathbf{y}\|_2 = 1$ by $\mathbf{y} = [1, 0, \ldots, 0]^T$, giving

$$\|\mathbf{A}\|_2 = \|\mathbf{\Sigma}\|_2 = \sigma_1.$$

Thus the matrix 2-norm is simply the first singular value. The 2-norm is often the 'natural' norm to use in applications, but if the matrix $\mathbf{A}$ is large, its computation is costly ($O(mn^2)$ floating point operations). For quick estimates that only require $O(mn)$ operations and are accurate to a factor of $\sqrt{m}$ or $\sqrt{n}$, use the matrix 1- or ∞-norms.

The SVD has many other important uses. For example, if $\mathbf{A} \in \mathbb{C}^{n\times n}$ is invertible, we have $\mathbf{A}^{-1} = \mathbf{V\Sigma}^{-1}\mathbf{U}^*$, and so $\|\mathbf{A}^{-1}\|_2 = 1/\sigma_n$. This illustrates that a square matrix is singular if and only if $\sigma_n = 0$. We shall explore this in more depth later when we use the SVD to construct low-rank approximations to $\mathbf{A}$.

Like the 2-norm, the Frobenius norm,

$$\|\mathbf{A}\|_F = \Big(\sum_{j=1}^{m}\sum_{k=1}^{n} |a_{jk}|^2\Big)^{1/2}$$

is unitarily invariant. What are $\|\mathbf{A}\|_F$ and $\|\mathbf{A}^{-1}\|_F$ in terms of the singular values of $\mathbf{A}$?

3.2.5. The SVD and the four fundamental subspaces.

For simplicity, assume $m \ge n$. Then $\mathbf{A} = \mathbf{U\Sigma V}^*$ can be written as the linear combination of m-by-n outer product matrices:

$$\mathbf{A} = \mathbf{U\Sigma V}^* = \begin{matrix}[\sigma_1\mathbf{u}_1 & \sigma_2\mathbf{u}_2 & \cdots & \sigma_n\mathbf{u}_n] \\ & & & \end{matrix} \begin{bmatrix} \mathbf{v}_1^* \\ \mathbf{v}_2^* \\ \vdots \\ \mathbf{v}_n^* \end{bmatrix} = \sum_{j=1}^{n} \sigma_j \mathbf{u}_j \mathbf{v}_j^*.$$

Hence for any $\mathbf{x} \in \mathbb{C}^n$,

$$\mathbf{Ax} = [\sum_{j=1}^{n} \sigma_j \mathbf{u}_j \mathbf{v}_j^*]\mathbf{x} = \sum_{j=1}^{n} (\sigma_j \mathbf{v}_j^*\mathbf{x})\mathbf{u}_j,$$

since $\mathbf{v}_j^*\mathbf{x}$ is just a scalar. We see that $\mathbf{Ax}$ is a linear combination of the left singular vectors $\{\mathbf{u}_j\}$, and have nearly uncovered a basis for $\text{Ran}(\mathbf{A})$. The only catch is that $\mathbf{u}_j$ will not contribute to the above linear combination if $\sigma_j = 0$. If all the singular values are nonzero, set $r = n$; otherwise, define r such that $\sigma_r \neq 0$ but $\sigma_{r+1} = 0$. Then we have

$$\mathbf{Ax} = \sum_{j=1}^{r} (\sigma_j \mathbf{v}_j^*\mathbf{x})\mathbf{u}_j,$$

and so

$$\text{Ran}(\mathbf{A}) = \Big\{\sum_{j=1}^{r} \gamma_j \mathbf{u}_j : \gamma_1, \ldots, \gamma_r \in \mathbb{C}\Big\}.$$

Since the vectors $\mathbf{u}_1, \ldots, \mathbf{u}_r$ are orthogonal by construction, they are linearly independent, and thus give a basis for $\mathrm{Ran}(\mathbf{A})$:

$$\mathrm{Ran}(\mathbf{A}) = \mathrm{span}\{\mathbf{u}_1, \ldots, \mathbf{u}_r\}.$$

Moreover, r is the dimension of $\mathrm{Ran}(\mathbf{A})$, i.e., $\mathrm{rank}(\mathbf{A}) = r$.

Immediately we have a basis for $\mathrm{Ker}(\mathbf{A}^*)$, too: The Fundamental Theorem of Linear Algebra guarantees that $\mathrm{Ran}(\mathbf{A}) \oplus \mathrm{Ker}(\mathbf{A}^*) = \mathbb{C}^m$ and $\mathrm{Ran}(\mathbf{A}) \perp \mathrm{Ker}(\mathbf{A}^*)$. Together these facts, with the orthogonality of the left singular vectors, gives

$$\mathrm{Ker}(\mathbf{A}^*) = \mathrm{span}\{\mathbf{u}_{r+1}, \ldots, \mathbf{u}_m\}.$$

Applying the same arguments to $\mathbf{A}^*$ yields bases for the two remaining fundamental subspaces:

$$\mathrm{Ran}(\mathbf{A}^*) = \mathrm{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}, \qquad \mathrm{Ker}(\mathbf{A}) = \mathrm{span}\{\mathbf{v}_{r+1}, \ldots, \mathbf{v}_n\},$$

where $\mathrm{Ran}(\mathbf{A}^*) \oplus \mathrm{Ker}(\mathbf{A}) = \mathbb{C}^n$ and $\mathrm{Ran}(\mathbf{A}^*) \perp \mathrm{Ker}(\mathbf{A})$. Hence, the SVD is a beautiful tool for revealing the fundamental subspaces.

3.2.6. Low-rank matrix approximation.

One of the key applications of the singular value decomposition is the construction of *low-rank approximations* to a matrix. Recall that the SVD of $\mathbf{A}$ can be written as

$$\mathbf{A} = \sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^*,$$

where $r = \mathrm{rank}(\mathbf{A})$. We can approximate $\mathbf{A}$ by taking only a partial sum here:

$$\mathbf{A}_k = \sum_{j=1}^{k} \sigma_j \mathbf{u}_j \mathbf{v}_j^*$$

for $k \le r$. The linear independence of $\{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$ guarantees that $\mathrm{rank}(\mathbf{A}_k) = k$. But how well does this partial sum approximate $\mathbf{A}$? This question is answered by the following result, due variously to Schmidt, Mirsky, Eckart, and Young, that has wide-ranging consequences in applications.

Theorem. For all $1 \le k < \mathrm{rank}(\mathbf{A})$,

$$\min_{\mathrm{rank}(\mathbf{X})=k} \|\mathbf{A} - \mathbf{X}\| = \sigma_{k+1},$$

with the minimum attained by

$$\mathbf{A}_k = \sum_{j=1}^{k} \sigma_j \mathbf{u}_j \mathbf{v}_j^*.$$

Proof. [See, e.g., J. W. Demmel, *Applied Numerical Linear Algebra*, §3.2.3] Let $\mathbf{X} \in \mathbb{C}^{m\times n}$ be any rank-k matrix. The Fundamental Theorem of Linear Algebra guarantees that $\mathbb{C}^n = \mathrm{Ran}(\mathbf{X}^*) \oplus \mathrm{Ker}(\mathbf{X})$. Since $\mathrm{rank}(\mathbf{X}^*) = \mathrm{rank}(\mathbf{X}) = k$, we conclude that $\dim(\mathrm{Ker}(\mathbf{X})) = n - k$.

From the singular value decomposition

$$\mathbf{A} = \sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^*,$$

extract the vectors $\{\mathbf{v}_1,\ldots,\mathbf{v}_{k+1}\}$, which form a basis for a $k+1$ dimensional subspace of $\mathbb{C}^n$. Since $\mathrm{Ker}(\mathbf{X}) \subseteq \mathbb{C}^n$ has dimension $n-k$, it must be that the intersection

$$\mathrm{Ker}(\mathbf{X}) \cap \mathrm{span}\{\mathbf{v}_1,\ldots,\mathbf{v}_{k+1}\}$$

is nontrivial, i.e., is at least one-dimensional.[†] Let $\mathbf{z}$ be some unit vector in that intersection:

$$\mathbf{z} \in \mathrm{Ker}(\mathbf{X}) \cap \mathrm{span}\{\mathbf{v}_1,\ldots,\mathbf{v}_{k+1}\}, \qquad \|\mathbf{z}\|_2 = 1.$$

Expand $\mathbf{z} = \gamma_1\mathbf{v}_1 + \cdots + \gamma_{k+1}\mathbf{v}_{k+1}$, so that $\|\mathbf{z}\|_2 = 1$ implies

$$1 = \mathbf{z}^*\mathbf{z} = \Big(\sum_{j=1}^{k+1}\gamma_j\mathbf{v}_j\Big)^*\Big(\sum_{j=1}^{k+1}\gamma_j\mathbf{v}_j\Big) = \sum_{j=1}^{k+1}|\gamma_j|^2.$$

Since $\mathbf{z} \in \mathrm{Ker}(\mathbf{X})$, we have

$$\|\mathbf{A}-\mathbf{X}\|_2 \;\ge\; \|(\mathbf{A}-\mathbf{X})\mathbf{z}\|_2 \;=\; \|\mathbf{A}\mathbf{z}\|_2 \;=\; \Big\|\sum_{j=1}^{k+1}\sigma_j\mathbf{u}_j\mathbf{v}_j^*\mathbf{z}\Big\|_2 \;=\; \Big\|\sum_{j=1}^{k+1}\sigma_j\gamma_j\mathbf{u}_j\Big\|_2.$$

Since $\sigma_{k+1} \le \sigma_k \le \cdots \le \sigma_1$ and the $\mathbf{u}_j$ vectors are orthogonal,

$$\Big\|\sum_{j=1}^{k+1}\sigma_j\gamma_j\mathbf{u}_j\Big\|_2 \ge \sigma_{k+1}\Big\|\sum_{j=1}^{k+1}\gamma_j\mathbf{u}_j\Big\|_2.$$

But notice that

$$\Big\|\sum_{j=1}^{k+1}\gamma_j\mathbf{u}_j\Big\|_2^2 = \Big(\sum_{j=1}^{k+1}\gamma_j\mathbf{u}_j\Big)^*\Big(\sum_{j=1}^{k+1}\gamma_j\mathbf{u}_j\Big) = \sum_{j=1}^{k+1}|\gamma_j|^2 = 1,$$

where the last equality was derived above from the fact that $\|\mathbf{z}\|_2 = 1$. In conclusion,

$$\|\mathbf{A}-\mathbf{X}\|_2 \;\ge\; \sigma_{k+1}\Big\|\sum_{j=1}^{k+1}\gamma_j\mathbf{u}_j\Big\|_2 \;=\; \sigma_{k+1}$$

for any rank-k matrix $\mathbf{X}$.

All that remains is to show that this bound is attained by $\mathbf{A}_k$, the kth partial sum of the singular value decomposition. We have

$$\mathbf{A}-\mathbf{A}_k \;=\; \sum_{j=1}^{r}\sigma_j\mathbf{u}_j\mathbf{v}_j - \sum_{j=1}^{k}\sigma_j\mathbf{u}_j\mathbf{v}_j \;=\; \sum_{j=k+1}^{r}\sigma_j\mathbf{u}_j\mathbf{v}_j.$$

But this last expression is essentially a singular value decomposition for $\mathbf{A}-\mathbf{X}$, with largest singular value σ_{k+1}. Hence $\|\mathbf{A}-\mathbf{A}_k\|_2 = \sigma_{k+1}$ as claimed, and we see that $\mathbf{A}_k$ is a best rank-k approximation to $\mathbf{A}$ in the two-norm. ■

Notice that we do not claim that the best rank-k approximation given in the theorem is *unique*. Can you think of how you might find other rank-k matrices $\widehat{\mathbf{A}}_k$ such that $\|\mathbf{A}-\widehat{\mathbf{A}}_k\|_2 = \|\mathbf{A}-\mathbf{A}_k\|_2$?

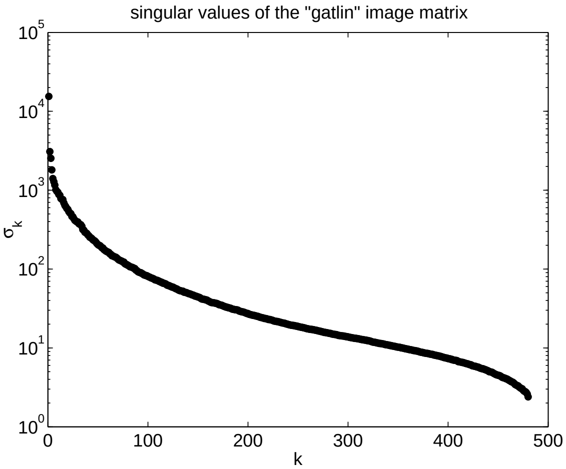
[†] Otherwise, $\mathrm{Ker}(\mathbf{X}) \oplus \mathrm{span}\{\mathbf{v}_1,\ldots,\mathbf{v}_{k+1}\}$ would be an $n+1$ dimensional subspace of $\mathbb{C}^n$: impossible!

Application: image compression. As an illustration of the utility of low-rank matrix approximations, consider the compression of digital images. On a computer, an image is simply a matrix denoting pixel colors. For example, a grayscale image can be represented as a matrix whose entries are integers between 0 and 255 (for 256 shades of gray), denoting the shade of each pixel. Typically, such matrices can be well-approximated by low-rank matrices. Instead of storing the mn entries of the matrix $\mathbf{A}$, one need only store the $k(m+n)+k$ numbers that make up the various σ_j, $\mathbf{u}_j$, and $\mathbf{v}_j$ values in the sum

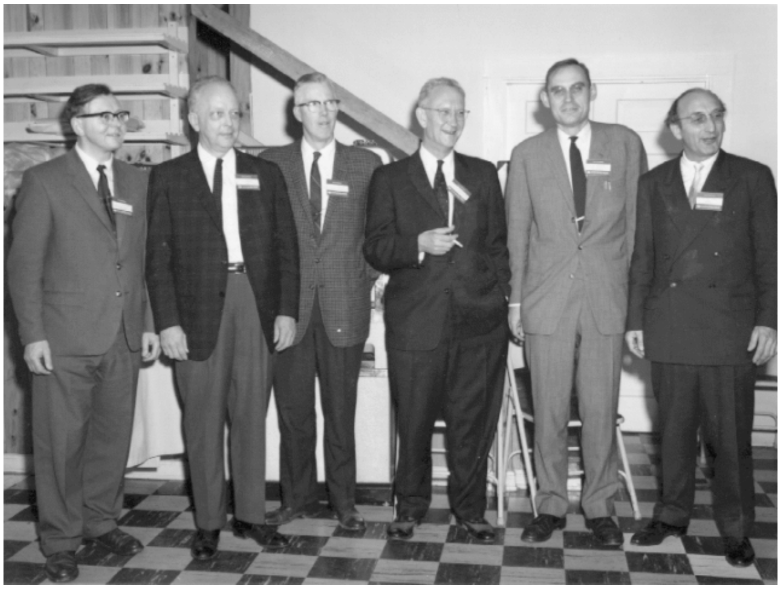
$$\mathbf{A}_k = \sum_{j=1}^{k} \sigma_j \mathbf{u}_j \mathbf{v}_j^*.$$

When $k \ll \min(m,n)$, this can make for a significant improvement (though modern image compression protocols use more sophisticated approaches).

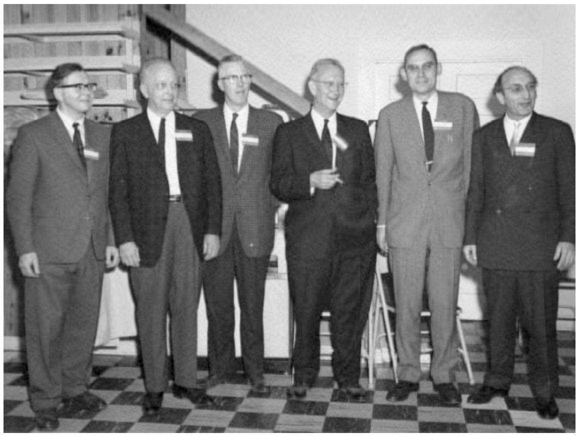
Next we show the singular values for one image matrix, a photograph of many of the patriarchs of modern matrix computations taken at the 1964 Gatlinburg Conference on Numerical Algebra: from left to right, we have Jim Wilkinson, Wallace Givens, George Forsythe, Alston Householder, Peter Henrici, and Fritz Bauer. The matrix is of dimension 480-by-640, reflecting the fact that the picture is wider than it is tall. Though the singular values are large, $\sigma_{480} > 1$, there is a relative difference of four orders of magnitude between the smallest and largest singular value. If all the singular values were roughly the same, we would not expect accurate low-rank approximations.)



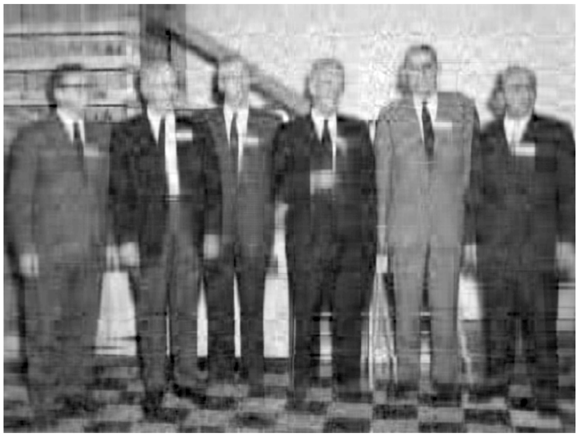
true image (rank 480)



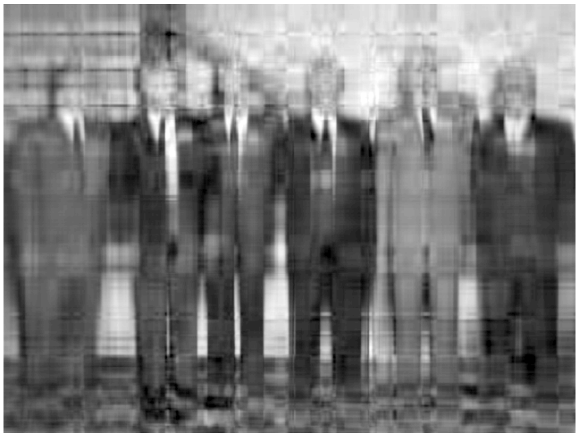
best rank−100 approximation



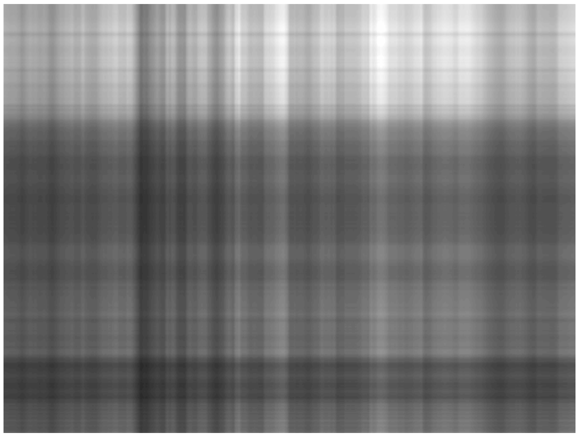
best rank−25 approximation



best rank−10 approximation



best rank−1 approximation



Below is a sample of that MATLAB code that generated these images, so you can experiment with this example further if you like. For further examples and more theory, see J. W. Demmel, *Applied Numerical Linear Algebra*, §3.2.3.

```
load gatlin                       % load the "gatlin" image data, built-in to MATLAB
[U,S,V] = svd(X);                 % "gatlin" stores the image as the variable "X"

figure(1),clf                     % plot the singular values
semilogy(diag(S),'b.','markersize',20)
set(gca,'fontsize',16)
title('singular values of the "gatlin" image matrix')
xlabel('k'), ylabel('\sigma_k')

figure(2),clf                     % plot the original image
image(X), colormap(map)           % image: MATLAB command to display a matrix as image
axis equal, axis off
title('true image (rank 480)','fontsize',16)

figure(3),clf                     % plot the optimal rank-k approximation
k = 100;
Xk = U(:,1:k)*S(1:k,1:k)*V(:,1:k)';
image(Xk), colormap(map)
axis equal, axis off
title(sprintf('best rank-%d approximation',k),'fontsize',16)
```