Example 121. What is the pseudoinverse A^+ of $A = \begin{bmatrix} 4 & 0 \\ 0 & 3 \\ 0 & 0 \end{bmatrix}$? Compute A^+A and AA^+.

Solution. The pseudoinverse is $A^+ = \begin{bmatrix} 1/4 & 0 & 0 \\ 0 & 1/3 & 0 \end{bmatrix}$.

The products are $A^+A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ and $AA^+ = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

Comment. Note how the pseudoinverse tries to behave like the regular inverse. But since A has only 2 columns, AA^+ can have rank at most 2, so it cannot be the full 3×3 identity.

Review 122.

- If A has SVD $A = U\Sigma V^T$, then its pseudoinverse is $A^+ = V\Sigma^+U^T$.
- The system $A\boldsymbol{x} = \boldsymbol{b}$ has "optimal" solution $\boldsymbol{x} = A^+\boldsymbol{b}$.

  Here, "optimal" means that $\boldsymbol{x}$ is the smallest least squares solution. In particular:

  - If $A\boldsymbol{x} = \boldsymbol{b}$ has a unique solution, then $\boldsymbol{x} = A^+\boldsymbol{b}$ is that solution.
  - If $A\boldsymbol{x} = \boldsymbol{b}$ has a unique least squares solution, then $\boldsymbol{x} = A^+\boldsymbol{b}$ is that least squares solution.
  - If $A\boldsymbol{x} = \boldsymbol{b}$ has many (possibly least squares) solutions, then $\boldsymbol{x} = A^+\boldsymbol{b}$ is one of these, namely the one with smallest norm.

    We haven't yet seen this case in action. The next examples illustrate the simplest case.

Example 123. Find the smallest solution to $A\boldsymbol{x} = [\,6\,]$ with $A = [\,2\ \ 0\ \ 0\,]$ in two ways: directly (because the equation is so simple) and using the pseudoinverse of $A = [\,2\ \ 0\ \ 0\,]$.

Solution. (direct) The general solution is $\boldsymbol{x} = \begin{bmatrix} 3 \\ s_1 \\ s_2 \end{bmatrix}$. Obviously, the smallest among these is $\begin{bmatrix} 3 \\ 0 \\ 0 \end{bmatrix}$.

Solution. $A^TA = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}[\,2\ \ 0\ \ 0\,] = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ has 4-eigenvector $\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ and 0-eigenvectors $\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$.

$\boldsymbol{u}_1 = \frac{1}{\sigma_1}A\boldsymbol{v}_1 = \frac{1}{2}[\,2\ \ 0\ \ 0\,]\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = 1$

Hence, $A = U\Sigma V^T$ with $U = [\,1\,]$, $\Sigma = [\,2\ \ 0\ \ 0\,]$, $V = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

[Dang! We could have realized from the beginning that A is already diagonal!]

$A^+ = V\Sigma^+U^T = \begin{bmatrix} 1/2 \\ 0 \\ 0 \end{bmatrix}$, $\boldsymbol{x} = A^+[\,6\,] = \begin{bmatrix} 3 \\ 0 \\ 0 \end{bmatrix}$.

Example 124.

(a) Find the pseudoinverse of $A = [\,1\ \ 2\ \ 3\,]$.

(b) Find the smallest solution to $x_1 + 2x_2 + 3x_3 = 6$.

As before, smallest solutions means the solution $\boldsymbol{x}$ such that $\|\boldsymbol{x}\|$ is as small as possible. One obvious solution is $[\ 1, 1, 1\ ]^T$, but is it the smallest?

Solution.

(a) $A^T A = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} [\ 1\ \ 2\ \ 3\ ] = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{bmatrix}$ has 14-eigenvector $\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$ and 0-eigenvectors $\begin{bmatrix} -2 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -3 \\ 0 \\ 1 \end{bmatrix}$.

$\boldsymbol{u}_1 = \frac{1}{\sigma_1} A \boldsymbol{v}_1 = \frac{1}{\sqrt{14}} [\ 1\ \ 2\ \ 3\ ] \frac{1}{\sqrt{14}} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = 1$

Hence, $A = U \Sigma V^T$ with $U = [\ 1\ ]$, $\Sigma = [\ \sqrt{14}\ \ 0\ \ 0\ ]$, $V = \begin{bmatrix} 1/\sqrt{14} & * & * \\ 2/\sqrt{14} & * & * \\ 3/\sqrt{14} & * & * \end{bmatrix}$.

$A^+ = V \Sigma^+ U^T = \begin{bmatrix} 1/\sqrt{14} & * & * \\ 2/\sqrt{14} & * & * \\ 3/\sqrt{14} & * & * \end{bmatrix} [\ 1/\sqrt{14}\ \ 0\ \ 0\ ] [\ 1\ ] = \frac{1}{14} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

Comment. No surprise on U. The only options for U are $U = [\ 1\ ]$ and $U = [\ -1\ ]$.

Homework. Complete the SVD of A. That is, find an option for the two missing columns of V, so that V is an orthogonal matrix. In other words, find an orthonormal basis for the orthogonal complement of $\boldsymbol{v}_1$.

(b) We are solving $A\boldsymbol{x} = [\ 6\ ]$ with $A = [\ 1\ \ 2\ \ 3\ ]$ as in the previous example.

We conclude that the smallest solution is $\boldsymbol{x} = A^+ [\ 6\ ] = \frac{3}{7} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$.

Compare. $\left\| \frac{3}{7} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \right\| = \frac{3}{7}\sqrt{14} \approx 1.604$ is indeed smaller than, say, $\left\| \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\| = \sqrt{3} \approx 1.732$.

Geometric picture. The equation $x_1 + 2x_2 + 3x_3 = 6$ describes a plane (not through the origin), and we are asking for the point on that plane which is closest to the origin. That's a typical question in Calculus III. Try and use this geometric picture to solve the problem. Then compare with our earlier answer.

Example 125. (homework) Find the pseudoinverse of $A = [\ a_1\ \ a_2\ \ a_3\ ]$.

Solution. Going through the previous example, we see that the answer will be $A^+ = \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|^2}$ with $\boldsymbol{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$.

Theorem 126. (matrix approximation lemma) Suppose A is a $m \times n$ matrix, and we want to approximate A using a matrix B of rank s (smaller than the rank of A).

Then, the best such approximation is $B = U_s \Sigma_s V_s^T$, where Σ_s is the $s \times s$ diagonal matrix with entries $\sigma_1, \sigma_2, \ldots, \sigma_s$ and U_s, V_s are obtained from the corresponding matrices in the SVD $A = U \Sigma V^T$ by only taking the first s columns.

Comment. This approximation will be good if the omitted singular values $\sigma_{s+1}, \sigma_{s+2}, \ldots, \sigma_r$ are all "small".

In other words. Here is another common way to say the same thing:

- Observe that $A = U \Sigma V^T$ is equivalent to $A = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$.
- Each matrix $\boldsymbol{u}_i \boldsymbol{v}_i^T$ has rank 1.
- The best rank s approximation to A is $B = \sum_{i=1}^{s} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$.

Example 127. (image compression) Let us load a 341x512 grayscale photo and store it as a matrix A. Each entry of the matrix is a value between 0 (black) and 1 (white).

The beautiful picture is taken from: http://www.southalabama.edu/departments/publicrelations/brand/photography.html

[The same approach works with color pictures. These are often represented by three matrices: one for the red component of the pixel, one for the green and for the blue component (RGB color scheme).]

```
Sage] import pylab
Sage] A = matrix(pylab.imread('/home/armin/photo.png'))
Sage] A.dimensions()
```

$(341, 512)$

```
Sage] A[0,0]
```

0.137254908681

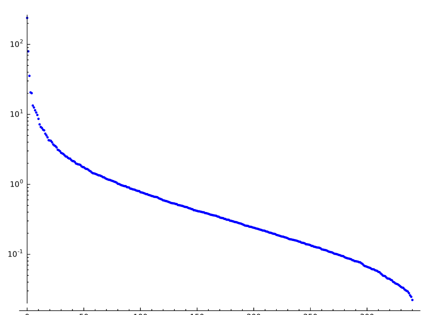
```
Sage] matrix_plot(A, cmap='gray')
```

Next, we compute the SVD of A. Despite the size of A that takes the computer only a fraction of a second:

```
Sage] U,S,V = A.SVD()
Sage] S.diagonal()[:6]
```

$[238.443435709, 79.4429775448, 35.4540786319, 20.5662302846, 20.0697710337, 13.3421216529]$

```
Sage] list_plot(S.diagonal(), scale='semilogy')
```



As we can see, the magnitude of the singular values drops off quickly. We get a good approximation to A (our original photo) by computing a best rank s approximation to A by computing $U_s \Sigma_s V_s$ where Σ_s is the $s \times s$ diagonal matrix with entries $\sigma_1, \sigma_2, \ldots, \sigma_s$ and U_s, V_s are obtained from the corresponding matrices in the SVD $A = U\Sigma V^T$ by only taking the first s columns.

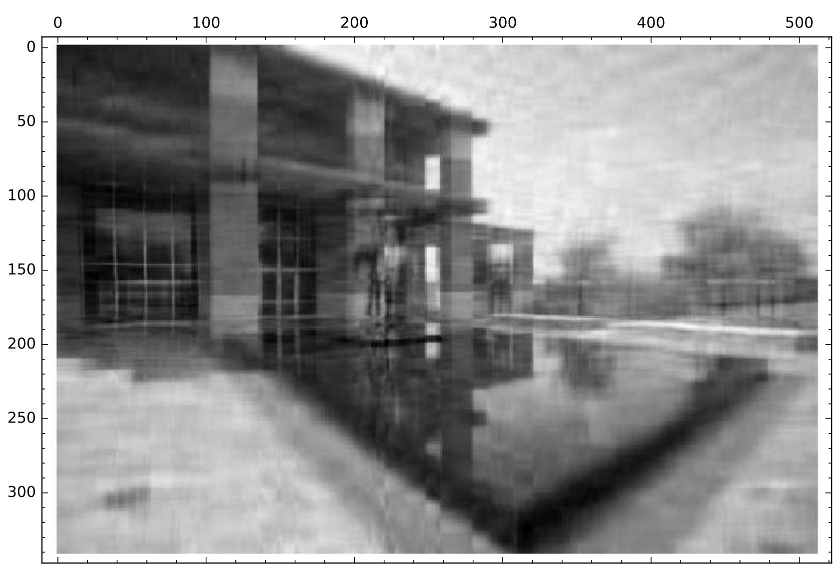
```
Sage] def A_approx(s):
          U0 = U.matrix_from_columns([0..s-1])
          S0 = diagonal_matrix(S.diagonal()[:s])
          V0 = V.matrix_from_columns([0..s-1])
          return U0*S0*V0.transpose()
```

Taking only 100 of the 341 singular values, we get an approximation, which is almost as good as the original:

```
Sage] matrix_plot(A_approx(100), cmap='gray')
```

But notice the development of artifacts. Taking only 20 of the 341 singular values, a lot is lost:

```
Sage] matrix_plot(A_approx(20), cmap='gray')
```



Comment. Image compression is just one (nice visual) example of the power of SVD. A variation of this approach can, for instance, also be used for image denoising. Much more generally, the SVD is able to extract the most important features of any sort of data!