EECS C145B / BioE C165: Image Processing and Reconstruction Tomography

Lecture 8

Jonathan S. Maltz

jon@eecs.berkeley.edu
http://muti.lbl.gov/145b
510-486-6744

Topics to be covered

1. Motivation
2. The meaning of matrix multiplication
3. Linear models
4. Solving overdetermined systems of equations using the least squares method
5. Review of the concepts of range and nullspace of a matrix.
6. The singular value decomposition (SVD)
7. Application of the SVD to deconvolution and image compression.

Motivation

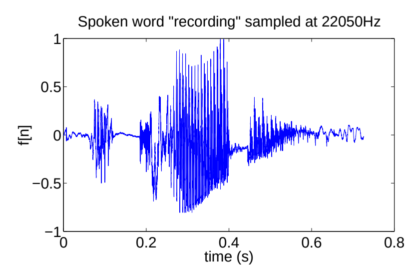
Eariler in the course, we looked at how convolution could be expressed as a matrix operation. In the cave echo example, we looked at an application of deconvolution, which involved inverting an equation such as:

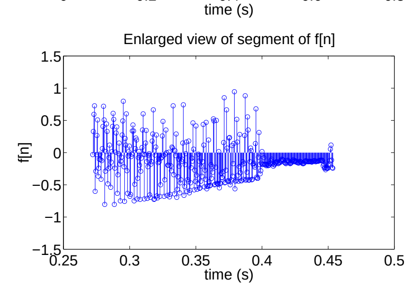
$$
\begin{bmatrix} g[0] \\ g[1] \\ g[2] \\ g[3] \\ g[4] \\ g[5] \\ g[6] \\ g[7] \\ g[8] \end{bmatrix}
=
\begin{bmatrix}
h[0] & 0 & 0 & 0 & 0 & 0 & 0 \\
h[1] & h[0] & 0 & 0 & 0 & 0 & 0 \\
h[2] & h[1] & h[0] & 0 & 0 & 0 & 0 \\
0 & h[2] & h[1] & h[0] & 0 & 0 & 0 \\
0 & 0 & h[2] & h[1] & h[0] & 0 & 0 \\
0 & 0 & 0 & h[2] & h[1] & h[0] & 0 \\
0 & 0 & 0 & 0 & h[2] & h[1] & h[0] \\
0 & 0 & 0 & 0 & 0 & h[2] & h[1] \\
0 & 0 & 0 & 0 & 0 & 0 & h[2]
\end{bmatrix}
\begin{bmatrix} f[0] \\ f[1] \\ f[2] \\ f[3] \\ f[4] \\ f[5] \\ f[6] \end{bmatrix}
$$

In general, it is not possible to find a unique inverse for $\mathbf{H}$, since this matrix is not always square.

Deconvolution example

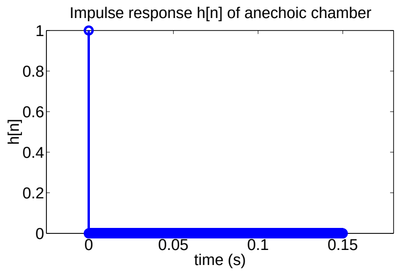
In the example, we convolved the spoken word "recording" ($f[n]$) with a "cave" impulse response.

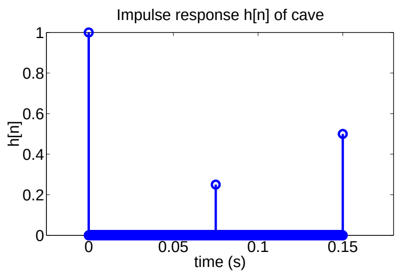




Deconvolution example

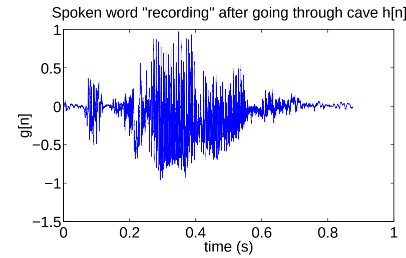
The "cave" impulse response $h[n]$ is shown in the lower figure:

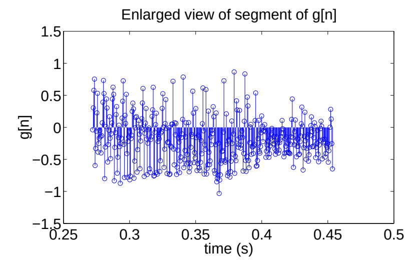




Deconvolution example

Convolution with the kernel gave $g[n]$:





Convolution and deconvolution procedure

- Inputs:

Original signal	$\mathbf{f}$	(3201×1)
Kernel	$\mathbf{h}$	(661×1)

- Make Toeplitz matrix $\mathbf{H}$ from kernel. $\mathbf{H}$ has size (3861×3201).
- Convolve to get (3861×1) echo signal:

$$\mathbf{g} = \mathbf{H}\mathbf{f}$$

- Perform deconvolution by finding some approximation to the inverse of $\mathbf{H}$:

$$\hat{\mathbf{f}} = \mathbf{H}^{\dagger}\mathbf{g}$$

  This gives an estimate $\hat{\mathbf{f}}$ of the original signal $\mathbf{f}$ that is optimal in some way.

- In this lecture, we will explore a very powerful method of finding an $\mathbf{H}^{\dagger}$ which will be useful for solving many inverse problems.

What is matrix multiplication?

Consider the equation:

$$\begin{aligned}
\mathbf{y} &= \mathbf{F}\boldsymbol{\theta} \\
&= \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \mathbf{f}_1 & \mathbf{f}_2 & \cdots & \mathbf{f}_N \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_N \end{bmatrix} \\
&= \theta_1\,\mathbf{f}_1 + \theta_2\,\mathbf{f}_2 + \ldots + \theta_N\,\mathbf{f}_N
\end{aligned}$$

Multiplying the vector $\boldsymbol{\theta}$ by the matrix $\mathbf{F}$ gives a vector y that is the sum of the scaled columns of $\mathbf{F}$. If the columns of $\mathbf{F}$ have dimension M (number of rows of $\mathbf{F}$), then $\mathbf{y}$ may be considered the linear combination of N vectors of dimension M.

What is matrix multiplication?

Now consider:

$$
\begin{aligned}
\boldsymbol{\theta} &= \mathbf{F}^{-1}\mathbf{y} \\
&= \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \mathbf{g}_1 & \mathbf{g}_2 & \cdots & \mathbf{g}_M \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix} \\
&= y_1\,\mathbf{g}_1 + y_2\,\mathbf{g}_2 + \ldots + y_M\,\mathbf{g}_M
\end{aligned}
$$

We see that the solution $\boldsymbol{\theta}$ is a linear combination of the columns of $\mathbf{F}^{-1}$. In order for $\boldsymbol{\theta}$ to be unique, these vectors need to be linearly independent. This lecture is concerned with finding a good estimate of $\boldsymbol{\theta}$ when these vectors are not independent. This is the case when $\mathbf{F}$ is not square or when the determinant of a square $\mathbf{F}$ is zero.

Linear models

- A simple application of a linear model is fitting a straight line to a set of points.
- Suppose we place a variable voltage x across a $10\,\Omega$ resistor. When $x = 10\,\text{V}$, we measure a current of 1008 mA. When we increase the voltage to $20\,\text{V}$, we read 1987 mA.
- We know we can find the equation of a line:

$$q(\boldsymbol{\theta}, x) = \theta_1 x + \theta_2$$

  that fits these points perfectly. How do we know this? Because we have __ independent measurements and __ unknowns.
- $q(\boldsymbol{\theta}, x)$ is a linear model, because it is a linear function of all the elements of the parameter vector:

$$\boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$$

Linear models

- We can find the gradient θ_1 and intercept θ_2 of this line by solving a system of simultaneous linear equations for the parameters $\boldsymbol{\theta}$:

$$
\begin{aligned}
\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} &= \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \qquad \text{or} \\
\mathbf{y} &= \mathbf{F}\boldsymbol{\theta}
\end{aligned}
$$

  where $\mathbf{y}$ is the vector of current measurements.
- The solution is given by:

$$\boldsymbol{\theta} = \mathbf{F}^{-1}\mathbf{y}$$

- Under what conditions are we guaranteed that $\mathbf{F}^{-1}$ exists?

____

Linear models

- The numerical solution is:

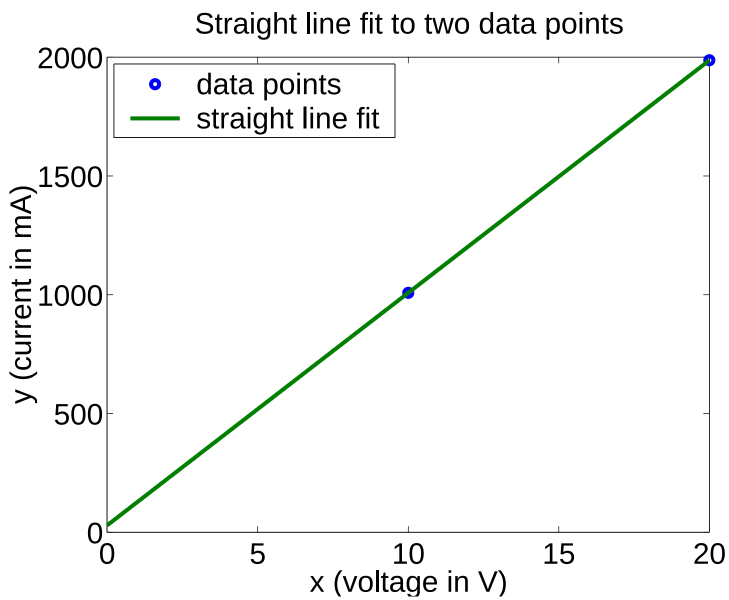
$$
\begin{aligned}
\boldsymbol{\theta} &= \begin{bmatrix} 10 & 1 \\ 20 & 1 \end{bmatrix}^{-1} \begin{bmatrix} 1008 \\ 1987 \end{bmatrix} \\
&= \begin{bmatrix} -0.1 & 0.1 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} 1008 \\ 1987 \end{bmatrix} \\
&= \begin{bmatrix} 97.9 \\ 29 \end{bmatrix}
\end{aligned}
$$

- Therefore, the current that flows through the resistor versus applied voltage may be expressed by the model:

$$q(\boldsymbol{\theta}, x) = 97.9\,x + 29 \quad \text{mA}$$

- Estimate the resistance of the resistor: ________________

Linear models



Linear models: More equations than unknowns

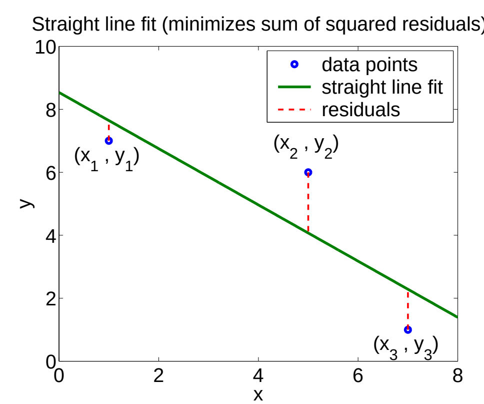
- When we measure physical quantities, such as photon counts in imaging, our measurements are **always contaminated** by **noise**.
- In general, the more measurements we make, the lower the effect of noise on the parameters we estimate.
- In imaging, our parameter vector most often consists of the pixel or voxel intensities. The quality of an image is therefore dependent on the number of measurements we make.
- For example, a digital camera image will look grainy if taken under low light conditions because:

Linear models: More equations than unknowns

- Our confidence in the accuracy of the resistance measurement would have been greater had we taken more than two measurements. However, the matrix **F** would then be non-square.
- When we have **more measurements than unknowns**, the **number of rows** M of **F** becomes **larger than the number of columns** N.
- We can no longer apply the standard matrix inverse to find $\boldsymbol{\theta}$.
- A system of equations that has a non-square **F** has **no unique solution**.
- We must choose **one** out of the infinite number of solutions that is optimal in some sense.

Linear models: Least-squares

- The least-squares (LS) method is the most commonly used technique for choosing a specific solution to linear equations defined by singular and non-square matrices.
- In LS, we **minimize the sum of the squared residuals**. A residual is the difference between the prediction of the model and the actual data at a specific value of x.



Linear models: Least-squares

- The previous slide showed a straight line fit to the points: $(1,7)$, $(5,6)$ and $(7,1)$.
- This problem involves using 3 measurements to determine 2 unknowns.
- The matrix equation defining the problem is:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} \qquad \text{or}$$

$$\mathbf{y} = \mathbf{F}\,\boldsymbol{\theta}$$

- This equation does not have a unique solution.
- The LS method will show us how to find a **pseudoinverse** $\mathbf{F}^+$ such that:

$$\boldsymbol{\theta}_{\mathrm{LS}} = \mathbf{F}^+\mathbf{y}$$

  where $\boldsymbol{\theta}_{\mathrm{LS}}$ is the unique solution that minimizes the sum of the squared residuals.

Linear models: Least-squares

- To give the problem a practical context, we begin by solving this particular LS problem. Later, we will generalize the method to all systems of equations that have more independent measurements than unknowns.
- Our model is given by:

$$q(\boldsymbol{\theta}, x) = \theta_1\, x + \theta_2$$

- The residual for data point y_m at abscissa point x_m is then:

$$r_m = y_m - q(\boldsymbol{\theta}, x_m) = y_m - (\theta_1\, x_m + \theta_2)$$

- The sum of squared residuals is:

$$C = \sum_{m=1}^{M} r_m^2 = \sum_{m=1}^{M} \Big[y_m - (\theta_1\, x_m + \theta_2) \Big]^2$$

- The LS problem is defined by the expression:

$$\boldsymbol{\theta}_{\mathrm{LS}} = \arg\min_{\boldsymbol{\theta}} C$$

Linear models: Least-squares

- We know that C will be minimized with respect to $\boldsymbol{\theta}$ if the M equations:

$$\frac{dC}{d\boldsymbol{\theta}} = \mathbf{0}$$

  i.e., if both

$$\frac{\partial C}{\partial \theta_1} = 2\sum_{m=1}^{M} \Big[y_m - (\theta_1\, x_m + \theta_2) \Big](-x_m) = 0 \qquad \text{and}$$

$$\frac{\partial C}{\partial \theta_2} = 2\sum_{m=1}^{M} \Big[y_m - (\theta_1\, x_m + \theta_2) \Big](-1) = 0$$

  hold.

- For this problem, the respective equations are:

$$(\_\_\_ - \theta_1 - \theta_2)(\_\_\_) + (6 - 5\,\theta_1 - \theta_2)(\_\_\_) + (1 - 7\,\theta_1 - \theta_2)(-7) = 0$$

$$(\_\_\_ - \theta_1 - \theta_2)(\_\_\_) + (6 - 5\,\theta_1 - \theta_2)(-1) + (1 - 7\,\theta_1 - \theta_2)(\_\_\_) = 0$$

  which simplify to

$$75\,\theta_1 + \_\_\_\,\theta_2 = 44$$

$$\_\_\_\,\theta_1 + 3\,\theta_2 = 14$$

Linear models: Least-squares

- We now have a 2×2 system that we can solve by matrix inversion:

$$\begin{bmatrix} 75 & \_\_\_ \\ \_\_\_ & 3 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 44 \\ 14 \end{bmatrix}$$

  giving:

$$\boldsymbol{\theta}_{\mathrm{LS}} = \begin{bmatrix} -0.8929 \\ 8.5357 \end{bmatrix}$$

- The straight line equation that minimizes the sum of squared residuals is:

$$q(\boldsymbol{\theta}, x) = -0.8929x + 8.5357$$

Linear models: Generalizing linear models

We can generalize the linear model as follows:

- Let the matrix $\mathbf{F}$ have the general form:

$$\begin{bmatrix} f_1^1 & f_2^1 & \cdots & f_N^1 \\ f_1^2 & f_2^2 & \cdots & f_N^2 \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & f_N^{M-1} \\ f_1^M & \cdots & f_{N-1}^M & f_N^M \end{bmatrix}$$

where N is the dimension of $\boldsymbol{\theta}$ and M is the number of data points.

- $\mathbf{F}$ is an $(M \times N)$ matrix.

Linear models: Generalizing linear models

- When we fitted a straight line in the previous example, $\mathbf{F}$ had a very simple form:

$$\begin{aligned} f_1^m &= x_m && \text{for all } m \\ f_2^m &= 1 && \text{for all } m \end{aligned}$$

giving:

$$\mathbf{F} = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{bmatrix}$$

Linear models: Generalizing linear models

- In the generalized model, the model functions are:

$$q_m(\boldsymbol{\theta}, x_m) = \sum_{n=1}^{N} f_n^m(x_m)\,\theta_n \qquad m = 1, 2, \ldots, M$$

and can be expressed in compactly in matrix form as the equation:

$$\mathbf{q} = \mathbf{F}\boldsymbol{\theta}$$

- Each observation may be thus be modeled as an arbitrary linear combination of the parameters θ_n.
- The residual is:

$$r_m = \sum_{n=1}^{N} f_n^m(x_m)\,\theta_n - y_m$$

and the function to be minimized is:

$$C = \sum_{m=1}^{M} r_m^2 = \sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\,\theta_n - y_m \right]^2$$

Linear models: Generalizing linear models

- The N partial derivatives are set to zero to perform the minimization:

$$\frac{\partial C}{\partial \theta_k} = 2 \sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\,\theta_n - y_m \right] f_k^m(x_m) = 0, \qquad k = 1, 2, \ldots N$$

- Reversing the order of the summations and rearranging terms gives:

$$\sum_{n=1}^{N} \theta_n \sum_{m=1}^{M} f_n^m(x_m)\, f_k^m(x_m) = \sum_{m=1}^{M} y_m f_k^m(x_m), \qquad k = 1, 2, \ldots N$$

- It is possible to write out these M equations very compactly in matrix form as:

$$\mathbf{F}^T \mathbf{F}\, \boldsymbol{\theta}_{\mathrm{LS}} = \mathbf{F}^T \mathbf{y}$$

Linear models: Generalizing linear models

- Multiplying both sides by $(\mathbf{F}^T\mathbf{F})^{-1}$ gives:

$$\boldsymbol{\theta}_{\mathrm{LS}} = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T\mathbf{y}$$

- Since this solution is of the form:

$$\boldsymbol{\theta}_{\mathrm{LS}} = \mathbf{F}^{+}\mathbf{y}$$

we identify the **pseudoinverse** that gives the LS solution as:

$$\mathbf{F}^{+} = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T$$

This is called the Moore-Penrose pseudoinverse.

- This pseudoinverse exists as long as $\mathbf{F}^T\mathbf{F}$ can be inverted. When will this be the case?

- We will see later that the least-squares solution finds that solution **in the range of F** that gives:

$$\mathbf{q} = \mathbf{F}\,\boldsymbol{\theta}_{\mathrm{LS}}$$

such that **q** is **closest** to the data vector **y**.

- We will now review the concepts of **range** and **nullspace**.

Range and nullspace

- Consider an $N \times N$ matrix **F**. The set of simultaneous equations:

$$\mathbf{F}\,\boldsymbol{\theta} = \mathbf{y}$$

can be viewed as a mapping of the vector space $\boldsymbol{\theta}$ to the vector space **y**.

- It is very important for us to be able to establish if a matrix **F** is capable of mapping a vector $\boldsymbol{\theta}$ onto a known vector **y**.
- In a practical context, we need to know whether it is possible to find a unique parameter vector $\boldsymbol{\theta}$ that when operated upon by our linear model matrix **F**, can reproduce the measured data **y**.

Range and nullspace

- We will demonstrate this by example. We define the matrix:

$$\mathbf{F} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$$

- We can see that its rows are independent, so it is non-singular. We can check this by calculating the determinant of **F** as: $(1 \times 1) - (-1 \times 1) = 2 \neq 0$
- We use this matrix to map the vector $\boldsymbol{\theta}$ onto another vector **y** in $\Re^2$. We know **y**:

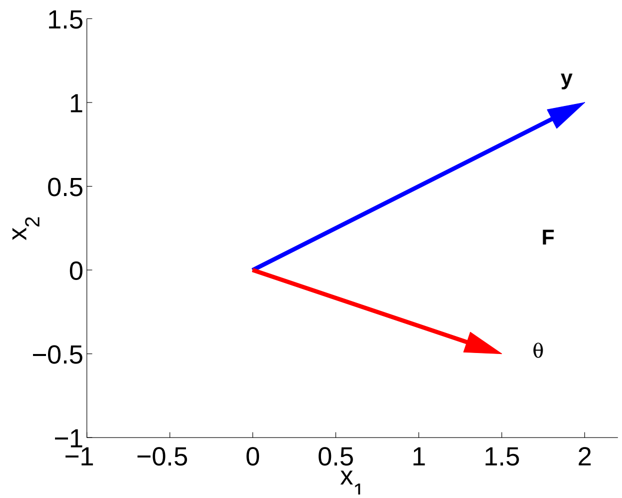
$$\mathbf{y} = \mathbf{F}\,\boldsymbol{\theta} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

and find $\boldsymbol{\theta}$ by inverting **F**:

$$\boldsymbol{\theta} = \begin{bmatrix} 1.5 \\ -0.5 \end{bmatrix}$$

Range and nullspace

The matrix **F** maps vector $\boldsymbol{\theta}$ to another vector **y**. We say that **y** is in the **range** of **F**.



Range and nullspace

- Now we'll define the matrix:

$$\mathbf{F} = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}$$

- We can see that its rows are **dependent**, so it is **singular.** To verify this, we calculate the determinant as $(1 \times -1) - (-1 \times 1) = 0$.
- We use this matrix to try to map the vector $\boldsymbol{\theta}$ onto another vector $\mathbf{y}$ in $\Re^2$:

$$\mathbf{y} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

- However, we cannot find $\boldsymbol{\theta}$ by inverting $\mathbf{F}$ because $\mathbf{F}^{-1}$ is not defined.
- Singular matrices have a **nullspace**. This means that some non-zero vector $\boldsymbol{\theta}$ exists such that:

$$\mathbf{F}\,\boldsymbol{\theta} = \mathbf{0}$$

- Find an example of a vector in the nullspace of $\mathbf{F}$: ________________.
- How can we determine whether $\mathbf{y}$ is in the range of $\mathbf{F}$?

Range and nullspace: The SVD

- Any $M \times N$ matrix can be decomposed into a product of three matrices that possess special and useful properties.
- This decomposition is called the singular value decomposition (SVD). The proof that the SVD always exists is beyond the scope of this course. We will also not discuss the numerical methods involved in finding the SVD.
- The SVD decomposition is given by:

$$\mathbf{F} = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

where:

- The N columns of $\mathbf{U}$, the $\mathbf{u}_n$ are orthonormal:

$$\mathbf{u}_k^T \mathbf{u}_l = \delta(k-l)$$

- Similarly for the N columns of $\mathbf{V}$, $\mathbf{v}_n$:

$$\mathbf{v}_k^T \mathbf{v}_l = \delta(k-l)$$

Range and nullspace: The SVD

- When $M \geq N$, the matrix $\mathbf{S}$ is $N \times N$ and diagonal and contains the N **singular values** of $\mathbf{F}$.
- The $M \times N$ matrix $\mathbf{U}$ contains the N **left singular vectors**.
- The $N \times N$ matrix $\mathbf{V}$ contains the N **right singular vectors**.
- The **columns** of $\mathbf{U}$ for which the corresponding singular values are **non-zero** span the **range** of $\mathbf{F}$.
- The **columns** of $\mathbf{V}$ for which the corresponding singular values are **zero** span the **nullspace** of $\mathbf{F}$.

The SVD

- As an example, we take the SVD of the non-singular matrix $\mathbf{F}$:

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \\ &= \mathbf{U}\mathbf{S}\mathbf{V}^T \\ &= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{aligned}$$

- What can we say about the dimension of the range of $\mathbf{F}$?

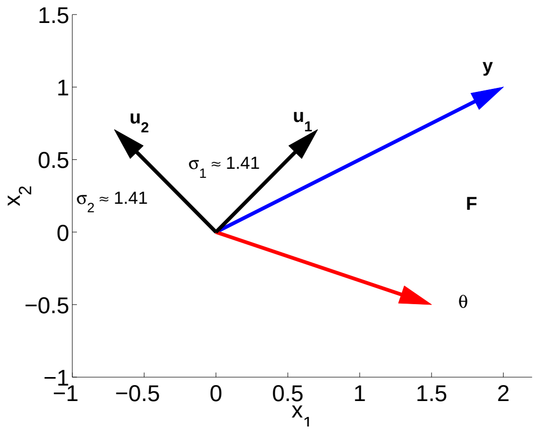
____

- What can we say about the dimension of the nullspace of $\mathbf{F}$?

____

The SVD

The two left singular vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ have corresponding non-zero singular values, and so define an orthonormal **basis** for the **range** of $\mathbf{F}$. Thus, **all** vectors in $\Re^2$ that can be represented as a linear combination of these vectors are in the range of $\mathbf{F}$. Since this basis spans $\Re^2$, every vector in $\Re^2$ is in the range of $\mathbf{F}$.



The SVD

We now show how the vector $\mathbf{y}$ can be expressed in terms of the orthonormal basis defined by $\mathbf{u}_1$ and $\mathbf{u}_2$.

- Currently, $\mathbf{y}$ is expressed in terms of the Cartesian basis defined by $\mathbf{e}_1$ and $\mathbf{e}_2$. These are the unit vectors along the x_1 and x_2 axes:

$$\mathbf{y} = 2\,\mathbf{e}_1 + 1\,\mathbf{e}_2$$

We wish to reexpress it in terms of u_1 and u_2:

$$\mathbf{y} = k_1\mathbf{u}_1 + k_2\mathbf{u}_2$$

We find k_1 by taking the dot product of $\mathbf{y}$ and $\mathbf{u}_1$:

$$k_1 = \mathbf{u}_1^T\mathbf{y} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = 2.121$$

- This is the **projection** of $\mathbf{y}$ onto $\mathbf{u}_1$.

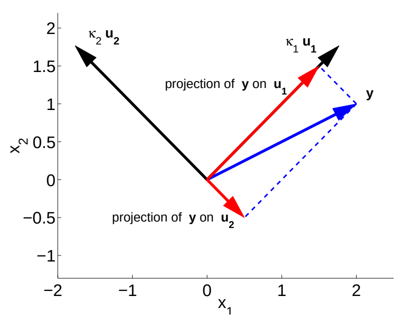
The SVD

- Similarly for k_2:

$$k_2 = \mathbf{u}_2^T\mathbf{y} = \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = -0.707$$

So, $\mathbf{y}$ can be fully expressed as a linear combination of the orthonormal basis for the range of $\mathbf{F}$ as:

$$\mathbf{y} = 2.121\mathbf{u}_1 - 0.707\mathbf{u}_2$$



The SVD

- Now we decompose the singular matrix:

$$\begin{aligned}
\mathbf{F} &= \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \\
&= \mathbf{USV}^T \\
&= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}
\end{aligned}$$

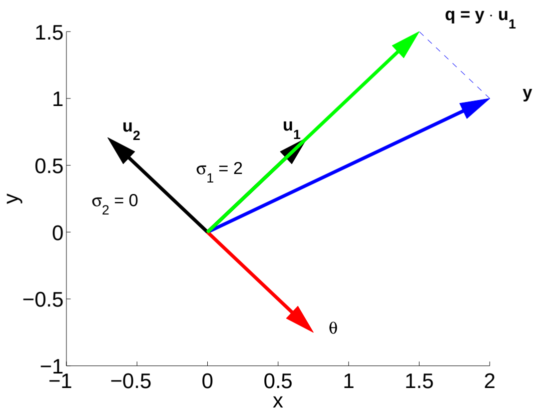
- What can we say about the dimension of the range of $\mathbf{F}$?

- What can we say about the dimension of the nullspace of $\mathbf{F}$?

- Is the vector $\mathbf{y} = [2 \;\; 1]^T$ in the range of $\mathbf{F}$?

The SVD

The first left singular vector $\mathbf{u}_1$ has a corresponding non-zero singular value. However $\mathbf{u}_2$ does not and so is **not** a basis vector for the range of $\mathbf{F}$. Consequently, instead of having a range that is the whole plane in $\Re^2$, the range of $\mathbf{F}$ is only the line defined by $\mathbf{u}_1$. Since $\mathbf{y}$ is not a multiple of $\mathbf{u}_1$, it is not in the range of $\mathbf{F}$. We can see this also because $\|\mathbf{y}-\mathbf{q}\| \neq 0$.



The SVD

- When $\mathbf{F}$ was non-singular, the SVD gave us the ability to express $\mathbf{y}$ in terms of the basis vectors of the range of $\mathbf{F}$:

$$\mathbf{y} = \left[\sum_{n=1}^{N} \mathbf{y}\cdot\mathbf{u}_n\right]\mathbf{u}_n$$

- When $\mathbf{F}$ is singular or non-square, we can find the vector $\mathbf{q}$ in the range of $\mathbf{F}$ that is **closest** to $\mathbf{y}$ using:

$$\mathbf{q} = \left[\sum_{n=1}^{N'} \mathbf{y}\cdot\mathbf{u}_n\right]\mathbf{u}_n$$

  where N' is the number of non-zero singular values.

- If the distance between $\mathbf{y}$ and $\mathbf{q}$:

$$\mathbf{r} = \|\mathbf{y}-\mathbf{q}\|$$

  is zero, then $\mathbf{y}$ is in the range of $\mathbf{F}$.

The SVD

We now show how the vector $\mathbf{q}$ can be expressed in terms of the orthonormal basis defined by $\mathbf{u}_1$ by finding k_1 in:

$$\mathbf{q} = k_1\mathbf{u}_1$$

Taking the dot product of $\mathbf{y}$ and $\mathbf{u}_1$:

$$k_1 = \mathbf{u}_1^T\mathbf{y} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}\begin{bmatrix} 2 \\ 1 \end{bmatrix} = 2.121$$

Thus:

$$\mathbf{q} = 2.121\mathbf{u}_1 + 0\,\mathbf{u}_2$$

The length of the difference vector $\mathbf{r}$ is:

$$\begin{aligned} \|\mathbf{r}\| &= \sqrt{(\mathbf{y}-\mathbf{q})^T(\mathbf{y}-\mathbf{q})} = \sqrt{\begin{bmatrix} (2-2.121) & (1+0) \end{bmatrix}\begin{bmatrix} 2-2.121 \\ 1+0 \end{bmatrix}} \\ &= 1.007 \neq 0 \end{aligned}$$

So $\mathbf{y}$ is **not** in the range of $\mathbf{F}$.

The SVD

- It can be shown that the orthogonal projection of $\mathbf{y}$ onto $\mathbf{u}_1$:

$$\mathbf{q} = \mathbf{y}\cdot\mathbf{u}_1$$

  is related to the LS solution by:

$$\mathbf{q} = \mathbf{F}\,\boldsymbol{\theta}_{\text{LS}}$$

- We can get a valuable geometric interpretation of "minimizing the square of the residual" this from this example.
- $\mathbf{q}$ is that vector along $\mathbf{u}_1$ **closest** to the data vector $\mathbf{y}$, and $\boldsymbol{\theta}_{\text{LS}}$ is the parameter vector that gives it to us.

The SVD

- The dotted line in the diagram is the vector:

$$\mathbf{r} = \mathbf{y} - \mathbf{q}$$

  and this is the **residual vector**, or the difference between the linear model and the data. The LS solution minimizes the squared norm of this residual:

$$\min_{\boldsymbol{\theta}} \|\mathbf{r}\|^2$$

  which is identical to minimizing the norm of the residual

$$\min_{\boldsymbol{\theta}} \|\mathbf{r}\|$$

  which is what the SVD gives us as a solution.

- Why are these two minimizations equivalent?

SVD and least squares equivalence

To minimize the sum of squared residuals in the least squares formulation, we minimized:

$$C = \sum_{m=1}^{M} r_m^2 = \sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\, \theta_n - y_m \right]^2$$

We know that $\boldsymbol{\theta}_{\mathrm{LS}}$ is the solution that minimizes C.

To minimize the norm of the residual, we would minimize:

$$C' = \sqrt{\sum_{m=1}^{M} r_m^2} = \sqrt{\sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\, \theta_n - y_m \right]^2}$$

If we can show that $\boldsymbol{\theta}_{\mathrm{LS}}$ is the unique minimizer of C', then the two minimizations are equivalent.

SVD and least squares equivalence

Taking the derivative with respect to θ_k gives

$$\frac{\partial C'}{\partial \theta_k} = \frac{1}{2} \left[\sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\, \theta_n - y_m \right]^2 \right]^{-\frac{1}{2}} \times$$

$$2 \sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\, \theta_n - y_m \right] f_k^m(x_m) = 0, \qquad k = 1, 2, \ldots N$$

Now recall:

$$\frac{\partial C}{\partial \theta_k} = 2 \sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\, \theta_n - y_m \right] f_k^m(x_m) = 0, \qquad k = 1, 2, \ldots N$$

SVD and least squares equivalence

So

$$\frac{\partial C'}{\partial \theta_k} = \frac{1}{2} \left[\sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\, \theta_n - y_m \right]^2 \right]^{-\frac{1}{2}} \frac{\partial C}{\partial \theta_k}$$

The first factor of cannot be zero unless:

$$\sum_{m=1}^{M} \left[\sum_{n=1}^{N} f_n^m(x_m)\, \theta_n - y_m \right]^2$$

is infinite. In this case, C' would not be minimized. Therefore, the same unique $\boldsymbol{\theta}_{\mathrm{LS}}$ minimizes C and C' $\blacksquare$

The SVD of an overdetermined non-square matrix

- We define the matrix:

$$\mathbf{F} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix}$$

- We can see than $\mathbf{F}$ describes a system that has more equations (3) than unknowns (2).
- We can use the SVD to determine the range of $\mathbf{F}$.
- We will try to determine whether the vector:

$$\mathbf{y} = \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}$$

is in the range of $\mathbf{F}$ (in other words whether a $\boldsymbol{\theta}$ exists such that $\mathbf{F}\boldsymbol{\theta} = \mathbf{y}$).

The SVD of an overdetermined non-square matrix

- We decompose $\mathbf{F}$ using the SVD:

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} 1 & -1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix} \\ &= \mathbf{USV}^T \\ &= \begin{bmatrix} 0.1690 & -0.9487 \\ 0.5071 & 0.3162 \\ 0.8452 & 0.0000 \end{bmatrix} \begin{bmatrix} 2.6458 & 0 \\ 0 & 1.4142 \end{bmatrix} \begin{bmatrix} 0.8944 & 0.4472 \\ -0.4472 & 0.8944 \end{bmatrix} \end{aligned}$$

- Even though $\mathbf{F}$ is a mapping from 2 to 3 dimensions, we see it has only two left singular vectors spanning its range. Therefore, it maps the entire 2D plane into a 2D plane inside a 3D space.
- Consequently, not every vector in $\Re^3$ is in the range of $\mathbf{F}$.

The SVD of a overdetermined non-square matrix

- Is

$$\mathbf{y} = \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}$$

in the range of $\mathbf{F}$?

- We calculate $\mathbf{q}$ as:

$$\begin{aligned} \mathbf{q} &= \sum_{n=1}^{2} \left[\mathbf{y} \cdot \mathbf{u}_n\right] \mathbf{u}_n \\ &= 0.8452\, \mathbf{u}_1 - 1.5811\, \mathbf{u}_2 \\ &= 0.8452 \begin{bmatrix} 0.1690 \\ 0.5071 \\ 0.8452 \end{bmatrix} - 1.5811 \begin{bmatrix} -0.9487 \\ 0.3162 \\ 0.0000 \end{bmatrix} \\ &= \begin{bmatrix} 1.6429 \\ -0.0714 \\ 0.7143 \end{bmatrix} \end{aligned}$$

The SVD of a overdetermined non-square matrix

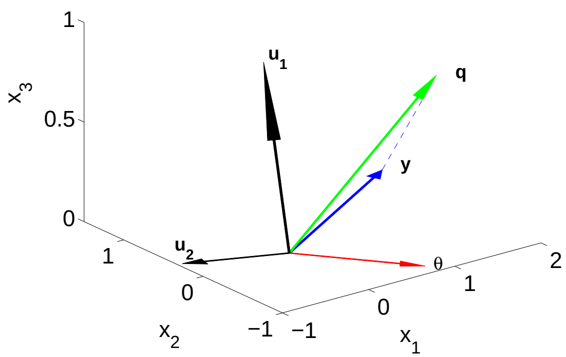
- We now calculate the residual vector $\mathbf{r}$:

$$\begin{aligned} \|\mathbf{r}\| &= \sqrt{\begin{bmatrix} (2-1.6429) & (1+0.0714) & (0-0.7143) \end{bmatrix} \begin{bmatrix} (2-1.6429) \\ (1+0.0714) \\ (0-0.7143) \end{bmatrix}} \\ &= 1.3363 \neq 0 \end{aligned}$$

So $\mathbf{y}$ is **not** in the range of $\mathbf{F}$, but we have found the closest solution in the range of $\mathbf{F}$, which is $\mathbf{q}$.

The SVD of a overdetermined non-square matrix

Both left singular vectors have corresponding non-zero singular values. However, since the range of $\mathbf{F}$ is two dimensional, all vectors in the 2D real plane get mapped to another 2D plane in $\Re^3$. Since $\mathbf{y}$ is not in this plane, it is not in the range of $\mathbf{F}$. From this view, we can see that $\mathbf{q}$ **is** a vector in the plane defined by $\mathbf{u}_1$ and $\mathbf{u}_2$. It is the orthogonal projection of $\mathbf{y}$ onto this plane.



The SVD of an row rank-deficient non-square matrix

- We define the matrix:

$$\mathbf{F} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \\ 1 & -1 \end{bmatrix}$$

- We can see than $\mathbf{F}$ describes a system that has more equations (3) than unknowns (2).
- We see that row 1 is the same as row 3. This type of matrix is termed **row rank-deficient** because not all the rows are independent.

The SVD of an row rank-deficient non-square matrix

- We can use the SVD to find the range of $\mathbf{F}$.
- We will try to determine whether the vector:

$$\mathbf{y} = \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}$$

is in the range of $\mathbf{F}$ (in other words whether a $\boldsymbol{\theta}$ exists such that $\mathbf{F}\boldsymbol{\theta} = \mathbf{y}$).

The SVD of an row rank-deficient non-square matrix

- We decompose $\mathbf{F}$ using the SVD:

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} 1 & -1 \\ 1 & 1 \\ 1 & -1 \end{bmatrix} \\ &= \mathbf{USV}^T \\ &= \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ 0 & -1 \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & \sqrt{2} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \end{aligned}$$

- Even though $\mathbf{F}$ is a mapping from 2 to 3 dimensions, we see it has only two singular vectors spanning its range. Therefore, it maps the entire 2D plane into a 2D plane inside a 3D space.
- Consequently, not every vector in $\Re^3$ is in the range of $\mathbf{F}$.

The SVD of an row rank-deficient non-square matrix

- Is

$$\mathbf{y} = \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}$$

in the range of $\mathbf{F}$?

- We calculate $\mathbf{q}$ as:

$$\begin{aligned} \mathbf{q} &= \left[\sum_{n=1}^{2} \mathbf{y} \cdot \mathbf{u}_n\right] \mathbf{u}_n \\ &= \sqrt{2}\mathbf{u}_1 - 1\mathbf{u}_2 \\ &= \sqrt{2}\begin{bmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{1}{\sqrt{2}} \end{bmatrix} - 1\begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \end{aligned}$$

The SVD of an row rank-deficient non-square matrix

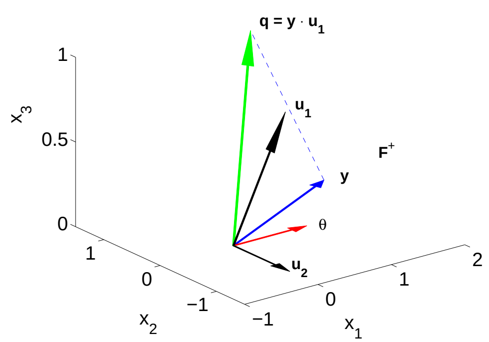
- And

$$\begin{aligned} \|\mathbf{r}\| &= \sqrt{\begin{bmatrix} (2-1) & (1-1) & (0-1) \end{bmatrix} \begin{bmatrix} (2-1) \\ (1-1) \\ (0-1) \end{bmatrix}} \\ &= \sqrt{2} \neq 0 \end{aligned}$$

So $\mathbf{y}$ is **not** in the range of $\mathbf{F}$, but we have found the closest solution in the range of $\mathbf{F}$, which is $\mathbf{q}$.

The SVD of an row rank-deficient non-square matrix

Both left singular vectors have corresponding non-zero singular values. However, since the range of $\mathbf{F}$ is two dimensional all vectors in the 2D real plane get mapped to another 2D plane in $\Re^3$. Since $\mathbf{y}$ is not in this plane, it is not in the range of $\mathbf{F}$. $\mathbf{q}$ **is** a vector in the plane defined by $\mathbf{u}_1$ and $\mathbf{u}_2$. It is the orthogonal projection of $\mathbf{y}$ onto this plane.



The SVD of a matrix with more columns than rows

- We define the matrix:

$$\mathbf{F} = \begin{bmatrix} 1 & -1 & 2 \\ 1 & 1 & 2 \end{bmatrix}$$

- We can see than $\mathbf{F}$ describes a system that has more unknowns (3) than equations (2).

The SVD of a matrix with more columns than rows

- We can use the SVD to find the range of $\mathbf{F}$.
- We will try to determine whether the vector:

$$\mathbf{y} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

is in the range of $\mathbf{F}$ (in other words whether a $\boldsymbol{\theta}$ exists such that $\mathbf{F}\boldsymbol{\theta} = \mathbf{y}$).

The SVD of a matrix with more columns than rows

- We decompose $\mathbf{F}$ using the SVD:

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} 1 & -1 & 2 \\ 1 & 1 & 2 \end{bmatrix} \\ &= \mathbf{USV}^T \\ &= \begin{bmatrix} 0.89 & 0.45 \\ 0.45 & -0.90 \end{bmatrix} \begin{bmatrix} 2.65 & 0 & 0 \\ 0 & 1.41 & 0 \end{bmatrix} \begin{bmatrix} 0.51 & -0.17 & 0.85 \\ -0.32 & -0.95 & 0.00 \\ -0.80 & 0.27 & 0.53 \end{bmatrix} \end{aligned}$$

- In this SVD, we see that $\mathbf{S}$ is no longer $N \times N$.
- $\mathbf{F}$ is a mapping from 3 to 2 dimensions, and has two singular vectors spanning its range. Therefore, it maps the entire 3D plane into the 2D plane $\Re^2$.
- **Every** vector in $\Re^2$ is in the range of $\mathbf{F}$, since it can be represented in terms of $\mathbf{u}_1$ and $\mathbf{u}_2$, which span $\Re^2$.

The SVD of an underdetermined non-square matrix

- To verify this, we ask: Is

$$\mathbf{y} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

in the range of $\mathbf{F}$?

- We calculate $\mathbf{q}$ as:

$$\begin{aligned} \mathbf{q} &= \left[\sum_{n=1}^{2} \mathbf{y} \cdot \mathbf{u}_n \right] \mathbf{u}_n \\ &= 3.5777\mathbf{u}_1 - 0.4472\mathbf{u}_2 \\ &= 3.5777 \begin{bmatrix} 0.8944 \\ 0.4472 \end{bmatrix} - 0.4472 \begin{bmatrix} 0.4472 \\ -0.8944 \end{bmatrix} \\ &= \begin{bmatrix} 3 \\ 2 \end{bmatrix} \end{aligned}$$

The SVD of an underdeterminednon-square matrix

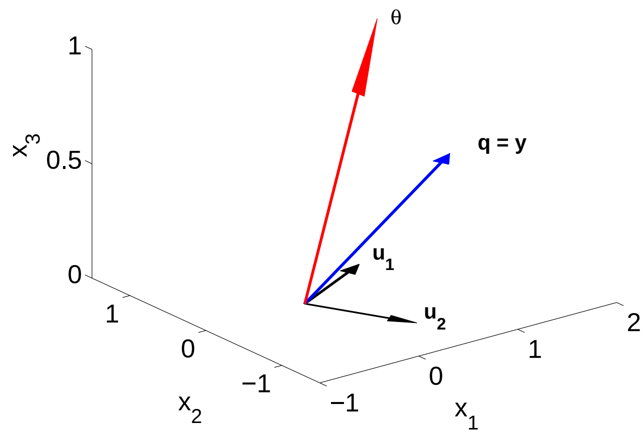
- And

$$\begin{aligned} \|\mathbf{r}\| &= \sqrt{\begin{bmatrix} (3-3) & (2-2) \end{bmatrix} \begin{bmatrix} (3-3) \\ (2-2) \end{bmatrix}} \\ &= 0 \end{aligned}$$

So $\mathbf{y}$ is in the range of $\mathbf{F}$. This will be true for any $\mathbf{y}$.

The SVD of a matrix with more columns than rows

Both left singular vectors have corresponding non-zero singular values. Therefore, the entire x_1-x_2 plane is spanned by the range of $\mathbf{F}$ and $\mathbf{F}$ can map any 3D vector $\boldsymbol{\theta}$ onto this plane.



Finding the least squares solution using the SVD

- Up to now, we have been finding the closest vector $\mathbf{q}$ to $\mathbf{y}$ by projecting $\mathbf{y}$ onto the columns of $\mathbf{U}$.
- Generally, we wish to find the solution $\boldsymbol{\theta}_{\mathrm{LS}}$, as this is the vector of unknowns. Once we have this solution, it is easy to find $\mathbf{q}$ as:

$$\mathbf{q} = \mathbf{F}\boldsymbol{\theta}_{\mathrm{LS}}$$

- Recall that the SVD of $\mathbf{F}$ is:

$$\mathbf{F} = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

- What happens if we try to take the inverse of $\mathbf{F}$?

Finding the least squares solution using the SVD

- Taking the inverse of both sides gives:

$$\begin{aligned} \mathbf{F}^{-1} &= (\mathbf{V}^T)^{-1}\mathbf{S}^{-1}\mathbf{U}^{-1} \\ &= \mathbf{V}\mathbf{S}^{-1}\mathbf{U}^T \end{aligned}$$

  Recall for a unitary matrix $\mathbf{U}^T = \mathbf{U}^{-1}$. Up to now, we have used the "economy-sized" SVD where the dimension of $\mathbf{U}$ is $M \times N$. To be able to find $\mathbf{U}^{-1}$, we must use the $M \times M$ version of $\mathbf{U}$ that may contain extra singular vectors that are not in the range of $\mathbf{F}$.

- Now, for these inverses to exist, all these matrices must be square, **and** all of the singular values on the diagonal of $\mathbf{S}$ must be non-zero.
- If any singular value is zero, $\mathbf{S}^{-1}$ will have an infinite element on its diagonal. This is how the the SVD shows us the singularities that cause the inverse of $\mathbf{F}$ to "blow-up".

Finding the least squares solution using the SVD

- Now, the SVD also tells us that if a singular value **is zero**, we **must not** try to find a solution that has a projection along the associated column of $\mathbf{U}$. This can be accomplished by setting to zero all infinite values of $\mathbf{S}^{-1}$. Then we have:

$$\mathbf{F}^{\dagger} = \mathbf{V}\mathbf{W}^{-1}\mathbf{U}^T$$

  where $\mathbf{W}^{-1}$ is the modified version of $\mathbf{S}^{-1}$.

- We will now prove that this pseudoinverse is the same as the Moore-Penrose pseudoinverse we derived eariler.

Finding the least squares solution using the SVD

We can prove that:

$$\boldsymbol{\theta}_{\text{SVD}} = \mathbf{F}^{\dagger}\mathbf{y}$$

is equal to:

$$\boldsymbol{\theta}_{\text{LS}} = \mathbf{F}^{+}\mathbf{y}$$

by showing that

$$\mathbf{q} = \mathbf{F}\,\boldsymbol{\theta}_{\text{SVD}}$$

minimizes the sum of squared residuals.
The following proof shows that we cannot add any vector in the range of $\mathbf{F}$ to the residual and get a shorter vector than the residual. If this is so then $\mathbf{q}$ is the closest vector to $\mathbf{y}$ in the range of $\mathbf{F}$.

Proof: The residual vector is defined as:

$$\mathbf{r} = (\mathbf{y} - \mathbf{q}) = (\mathbf{y} - \mathbf{F}\,\boldsymbol{\theta}_{\text{SVD}})$$

Let us modify $\boldsymbol{\theta}_{\text{SVD}}$ by adding some arbitrary $\boldsymbol{\theta}'$. Then $\mathbf{q}' = \mathbf{F}\,\boldsymbol{\theta}'$ is a vector in the range of $\mathbf{F}$.

Finding the least squares solution using the SVD

We then have:

$$\begin{aligned}
&\|\mathbf{y} + \mathbf{q}' - \mathbf{F}\boldsymbol{\theta}_{\text{SVD}}\| \\
&= \|\mathbf{y} + \mathbf{q}' - \mathbf{F}(\mathbf{F}^{\dagger}\mathbf{y})\| \\
&= \|\mathbf{y} + \mathbf{q}' - (\mathbf{U}\,\mathbf{S}\,\mathbf{V}^T)(\mathbf{V}\,\mathbf{W}^{-1}\,\mathbf{U}^T)\,\mathbf{y}\| \\
&= \|\mathbf{q}' - (\mathbf{U}\,\mathbf{S}\,\mathbf{W}^{-1}\mathbf{U}^T - \mathbf{I})\,\mathbf{y}\| \\
&= \left\|\mathbf{q}' - \mathbf{U}(\mathbf{S}\,\mathbf{W}^{-1}\,\mathbf{U}^T - \mathbf{U}^T\,\mathbf{y})\right]\right\| \\
&= \left\|\mathbf{U}\left[\mathbf{U}^T\mathbf{q}' - (\mathbf{S}\,\mathbf{W}^{-1} - \mathbf{I})\mathbf{U}^T\,\mathbf{y}\right]\right\| \\
&= \left\|\mathbf{U}^T\mathbf{q}' - (\mathbf{S}\,\mathbf{W}^{-1} - \mathbf{I})\mathbf{U}^T\,\mathbf{y}\right\|
\end{aligned}$$

(the final equality holds because, for an orthonormal matrix $\mathbf{U}$, $\|\mathbf{U}\mathbf{g}\| = \|\mathbf{g}\|$)
Let w_n be the nth diagonal element of $\mathbf{W}^{-1}$.

- For all n for which $w_n = 0$, the vector $\mathbf{u}_n$ is not part of the range of $\mathbf{F}$. Since $\mathbf{q}'$ is in the range, $\mathbf{U}^T\mathbf{q}' = \mathbf{0}$ for all of these values of n.

Finding the least squares solution using the SVD

- For example, when $N = 3$ and there are $N' = 2$ non-zero singular values:

$$\mathbf{U}^T\,\mathbf{q}' = \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \mathbf{u}_3^T \end{bmatrix} \mathbf{q}' = \begin{bmatrix} \mathbf{u}_1^T\mathbf{q}' \\ \mathbf{u}_2^T\mathbf{q}' \\ \mathbf{u}_3^T\mathbf{q}' \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T\mathbf{q}' \\ \mathbf{u}_2^T\mathbf{q}' \\ 0 \end{bmatrix}$$

  The last element is zero because $\mathbf{q}'$ is in the range of $\mathbf{F}$ and so can be represented as a linear combination of $\mathbf{u}_1$ and $\mathbf{u}_2$. Since $\mathbf{u}_3$ is orthogonal to these vectors by the definition of the SVD, $\mathbf{u}_3^T\mathbf{q}' = 0$.

- The diagonal matrix $(\mathbf{S}\,\mathbf{W}^{-1} - \mathbf{I})$ has elements that are either 0 or -1. The elements are -1 only for $w_n = 0$.

- Therefore the values of the non-zero elements of the second term are fixed by $\mathbf{y}$ and cannot be modified by the sum with $\mathbf{U}^T\,\mathbf{q}'$. Therefore, the norm of the residual is minimized when $\mathbf{q}'$ is smallest, i.e. when $\mathbf{q}' = \mathbf{0}$.

Finding the least squares solution using the SVD

- Therefore there is no vector in the range of $\mathbf{F}$ closer to $\mathbf{y}$ than:

$$\mathbf{q} = \mathbf{F}\boldsymbol{\theta}_{\text{SVD}}$$

  Thus, since $\boldsymbol{\theta}_{\text{SVD}}$ is that solution that minimizes the norm (and the squared norm of the residual, as we proved earlier):

$$\boldsymbol{\theta}_{\text{SVD}} = \boldsymbol{\theta}_{\text{LS}}$$

  and

$$\mathbf{F}^{\dagger} = \mathbf{F}^{+} \qquad \blacksquare.$$

Finding the least squares solution using the SVD

Using the same example, when $N = 3$ and there are $N' = 2$ non-zero singular values:

$(\mathbf{S}\,\mathbf{W}^{-1} - \mathbf{I})\mathbf{U}^T\,\mathbf{y}$

$$= \left(\begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/\sigma_1 & 0 & 0 \\ 0 & 1/\sigma_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) \begin{bmatrix} \mathbf{u}_1^T\mathbf{y} \\ \mathbf{u}_2^T\mathbf{y} \\ \mathbf{u}_3^T\mathbf{y} \end{bmatrix}$$

$$= \begin{bmatrix} 1-1 & 0 & 0 \\ 0 & 1-1 & 0 \\ 0 & 0 & 0-1 \end{bmatrix} \begin{bmatrix} \mathbf{u}_1^T\mathbf{y} \\ \mathbf{u}_2^T\mathbf{y} \\ \mathbf{u}_3^T\mathbf{y} \end{bmatrix}$$

$$= \begin{bmatrix} 0 \\ 0 \\ -\mathbf{u}_3^T\mathbf{y} \end{bmatrix}$$

Finding the least squares solution using the SVD

$$\mathbf{U}^T\,\mathbf{q}' = \begin{bmatrix} \mathbf{u}_1^T\mathbf{q}' \\ \mathbf{u}_2^T\mathbf{q}' \\ \mathbf{u}_3^T\mathbf{q}' \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T\mathbf{y} \\ \mathbf{u}_2^T\mathbf{y} \\ 0 \end{bmatrix}$$

So:

$$\left\| \mathbf{U}^T\mathbf{q}' - (\mathbf{S}\,\mathbf{W}^{-1} - \mathbf{I})\mathbf{U}^T\,\mathbf{y} \right\|$$

$$= \left\| \begin{bmatrix} \mathbf{u}_1^T\mathbf{q}' \\ \mathbf{u}_2^T\mathbf{q}' \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ -\mathbf{u}_3^T\mathbf{y} \end{bmatrix} \right\|$$

which is minimized when $\mathbf{q}' = 0$, because the third element of the vector is at its minimum (cannot by changed by any $\mathbf{q}'$).

Finding the least squares solution using the SVD

Worked example:

- We already considered the decomposition:

$$\begin{aligned} \mathbf{F} &= \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \\ &= \mathbf{USV}^T \\ &= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \end{aligned}$$

- We can now find the pseudoinverse:

$$\begin{aligned} \mathbf{F}^+ &= \mathbf{VW}^{-1}\mathbf{U}^T \\ &= \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1/2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \end{aligned}$$

Finding the least squares solution using the SVD

Worked example:

- We then solve for the parameter vector:

$$\begin{aligned} \boldsymbol{\theta}_{\mathrm{LS}} &= \mathbf{F}^+\,\mathbf{y} \\ &= \begin{bmatrix} 1/4 & 1/4 \\ -1/4 & -1/4 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 0.75 \\ -0.75 \end{bmatrix} \end{aligned}$$

- If we like, we can find the vector $\mathbf{q}$:

$$\mathbf{q} = \mathbf{F}\,\boldsymbol{\theta}_{\mathrm{LS}} = \begin{bmatrix} 1.5 \\ 1.5 \end{bmatrix}$$

  This is exactly the solution we found earlier by orthogonal projection.

Practical use of the SVD

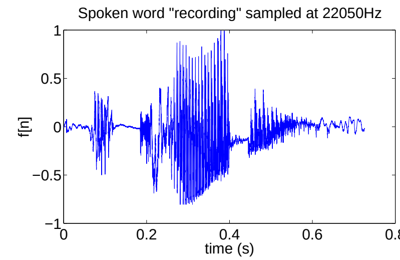
The SVD is very useful for solving many inverse problems. Earlier in the course we saw how convolution can be formulated as a matrix equation:

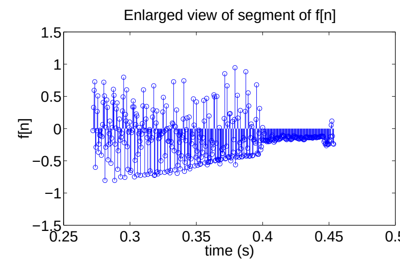
$$\begin{bmatrix} g[0] \\ g[1] \\ g[2] \\ g[3] \\ g[4] \\ g[5] \\ g[6] \\ g[7] \\ g[8] \end{bmatrix} = \begin{bmatrix} h[0] & 0 & 0 & 0 & 0 & 0 & 0 \\ h[1] & h[0] & 0 & 0 & 0 & 0 & 0 \\ h[2] & h[1] & h[0] & 0 & 0 & 0 & 0 \\ 0 & h[2] & h[1] & h[0] & 0 & 0 & 0 \\ 0 & 0 & h[2] & h[1] & h[0] & 0 & 0 \\ 0 & 0 & 0 & h[2] & h[1] & h[0] & 0 \\ 0 & 0 & 0 & 0 & h[2] & h[1] & h[0] \\ 0 & 0 & 0 & 0 & 0 & h[2] & h[1] \\ 0 & 0 & 0 & 0 & 0 & 0 & h[2] \end{bmatrix} \begin{bmatrix} f[0] \\ f[1] \\ f[2] \\ f[3] \\ f[4] \\ f[5] \\ f[6] \end{bmatrix}$$

Will will now perform a deconvolution using the SVD.

Deconvolution using the SVD

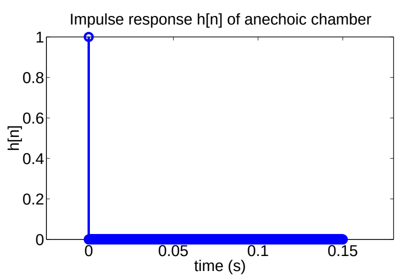
Earlier we convolved the spoken word "recording" ($f[n]$) with a "cave" impulse response.

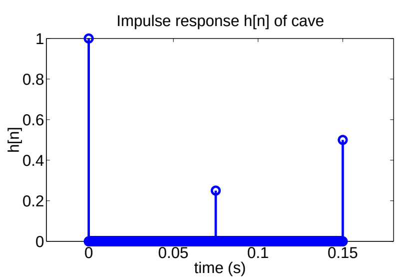




Deconvolution using the SVD

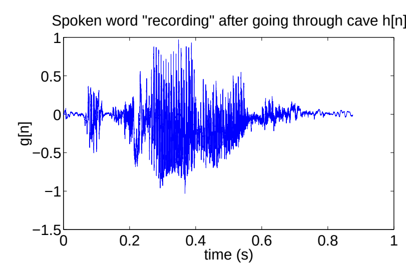
The "cave" impulse response $h[n]$ is shown in the lower figure:

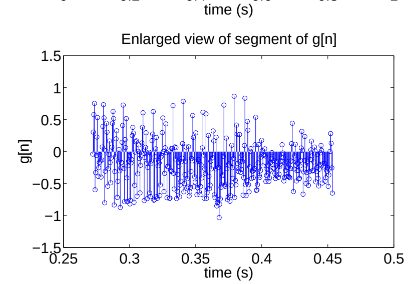




Deconvolution using the SVD

Convolution with the kernel gives $g[n]$:





Deconvolution using the SVD: Implementation

- Inputs:

Original signal	$\mathbf{f}$	(3201×1)
Kernel	$\mathbf{h}$	(661×1)

- Make Toeplitz matrix $\mathbf{H}$ from kernel. $\mathbf{H}$ has size (3861×3201).
- Convolve to get (3861×1) echo signal:

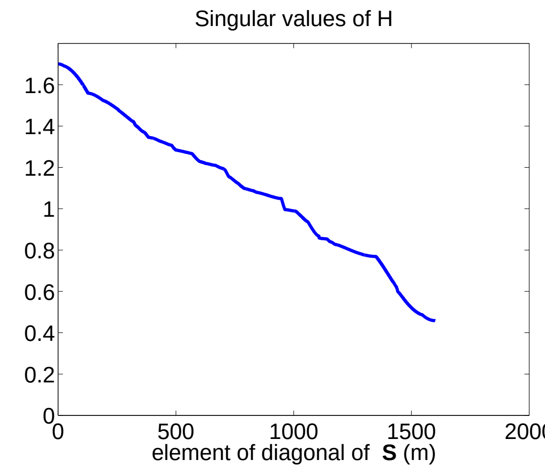
$$\mathbf{g} = \mathbf{H}\mathbf{f}$$

- Perform SVD on $\mathbf{H}$:

$$\begin{array}{ccccc} \mathbf{H} & = & \mathbf{U} & \mathbf{S} & \mathbf{V}^T \\ (3861 \times 3201) & & (3861 \times 3201) & (3201 \times 3201) & (3201 \times 3201) \end{array}$$

Deconvolution using the SVD

We can plot each singular value of $\mathbf{H}$ versus its position on the diagonal of $\mathbf{H}$. We call this the **spectrum of singular values**. Singular values are **always** ordered so that $\sigma_n \geq \sigma_{n+1}$. The flatter this spectrum, the better behaved is the matrix pseudoinverse.



Deconvolution using the SVD: Implementation

- Find the indices for all $\sigma_n = 0$. In practice, we often want to remove all the small singular values. This is because the inverse of small singular values is large and makes the solution "blow-up". In that case, we find the indices of all $\sigma_n \leq \sigma_{\min}$
- Let the number of retained singular values be N'. Then we form the $(N \times N)$ matrix $\mathbf{W}^{-1}$ that has as diagonal the $(N \times 1)$vector:

$$\mathbf{w} = \left[\begin{array}{cccccccc} \frac{1}{\sigma_1} & \frac{1}{\sigma_2} & \ldots & \frac{1}{\sigma_{N'}} & 0 & 0 & \ldots & 0 \end{array} \right]^T$$

Deconvolution using the SVD: Implementation

- $\mathbf{W}^{-1}$ has the form:

$$\mathbf{W}^{-1} = \begin{bmatrix} \frac{1}{\sigma_1} & 0 & \ldots & \ldots & \ldots & \ldots & \ldots & 0 \\ 0 & \frac{1}{\sigma_2} & 0 & \ldots & \ldots & \ldots & \ldots & 0 \\ \vdots & 0 & \ddots & \ddots & \ldots & \ldots & \ldots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ldots & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \frac{1}{\sigma_{N'}} & \ddots & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- Because only the first N' elements of the diagonal are zero, we need only use the first N' columns of $\mathbf{U}$ and $\mathbf{V}$ when calculating $F^+ = \mathbf{V}\mathbf{W}^{-1}\mathbf{U}$.
- For the deconvolution example, none of the singular values is zero, so all $N' = N$

Deconvolution using the SVD: Implementation in Matlab

```
% code to perform deconvolution using the SVD
% given: 1. Toeplitz convolution matrix H
%        2. Convolved signal g
% output: Deconvolved signal fDeconv

[U,S,V] = svd(H,0); % decompose H so that H = USV'
                    % Second argument = 0 gives SVD
                    % with MxN sized matrix U which
                    % is what we want.

s = diag(S); % put all the singular values into a vector

[M,N] = size(H);
```

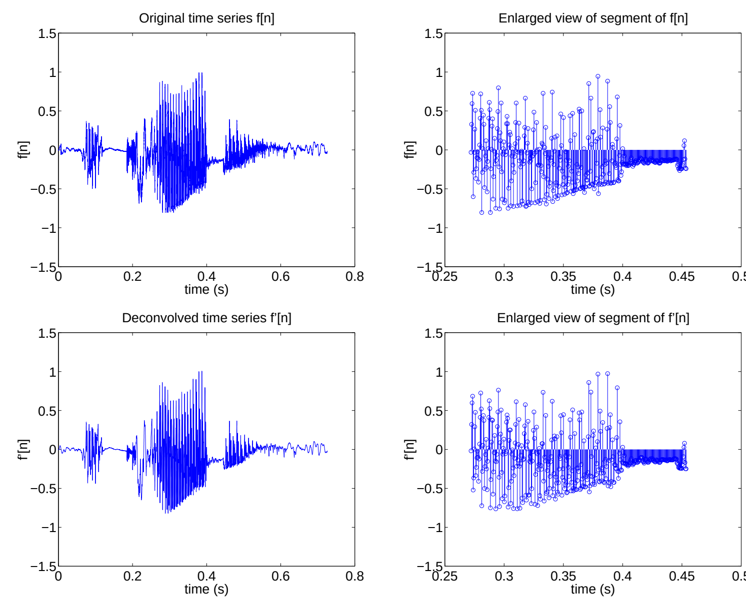

Deconvolution using the SVD: Implementation in Matlab

```
tol = max(M,N) * max(s) * eps; % Set minimum retained singular
                               % value equal to the maximum
                               % dimension of H times
                               % the maximum singular value
                               % times the numeric precision
                               % of the computer.
                               % Numbers smaller than
                               % tol are in effect
                               % zero on the particular
                               % computer being used.

ind = find(s > tol); % find the indices of the singular
                     % values that we wish to keep
```


Deconvolution using the SVD: Implementation in Matlab

```
NPrime = length(ind); % get the number of retained values N'
WInvDiag = zeros(size(s)); % make a blank diagonal for W's inverse

% place the inverse of the retained SVs in the vector
% that will become the diagonal of W inverse
WInvDiag(1:NPrime) = ones(NPrime,1) ./ s(1:NPrime);

WInv = diag(WInvDiag); % make this diagonal matrix with
                       % 1/sv_1 1/sv_2 ... 1/sv_N' 0 ... 0
                       % on diagonal

% get the pseudoinverse of H
HPlus = V(:,1:NPrime)*WInv*U(:,1:NPrime)';
% note the use of ' instead of .'. We use the
% Hermitian conjugate instead of the transpose
% so the pseudoinverse can be used for complex
% matrices too.

fDeconv = HPlus * g; % get the deconvolved signal (unknown vector theta)
```


Deconvolution using the SVD



Deconvolution using the SVD

- The SVD does a near perfect job of deconvolving the echo signal.
- All long distance telephone lines have echo-cancellers that perform similar deconvolutions. However, they cannot use the SVD to do this because it is too slow.
- This deconvolution took over 10 minutes on a Pentium 4 2.4GHz processor.

Approximation of matrices using the SVD

- The singular values tell us the relative importance of the singular vectors pairs $(\mathbf{u}_n, \mathbf{v}_n)$.
- The SVD can be rewritten as a series:

$$\mathbf{F} = \sum_{n=1}^{N} \sigma_n \mathbf{u}_n \mathbf{v}_n^T$$

  Here, each outer product $\mathbf{u}_n \mathbf{v}_n^T$ is a matrix that has the same size as $\mathbf{F}$.
- Since both $\mathbf{u}$ and $\mathbf{v}$ are unit vectors, the larger the value of σ_n, the larger is the contribution of one of these matrices to the series.
- One of the best (but slow) ways we can compress an image is to use only the first few terms in this series (Recall $\sigma_n \geq \sigma_{n+1}$).

Approximation of matrices using the SVD

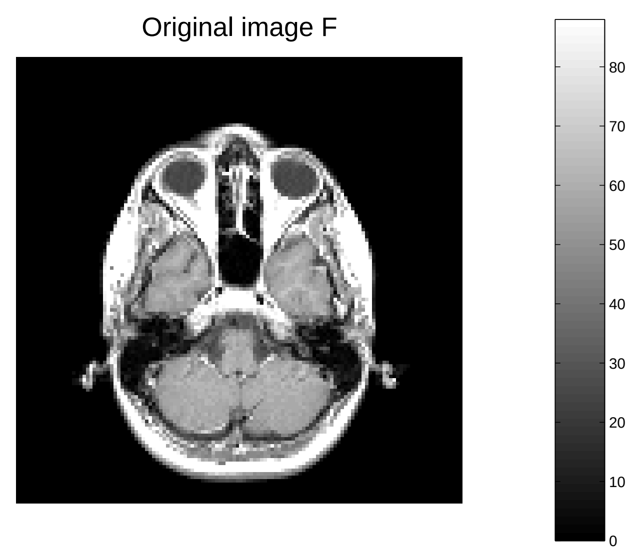
- Let $\mathbf{F}$ be a matrix containing any image. We can approximate the image with the first N' terms as:

$$\mathbf{F} \approx \sum_{n=1}^{N'} \sigma_n \mathbf{u}_n \mathbf{v}_n^T$$

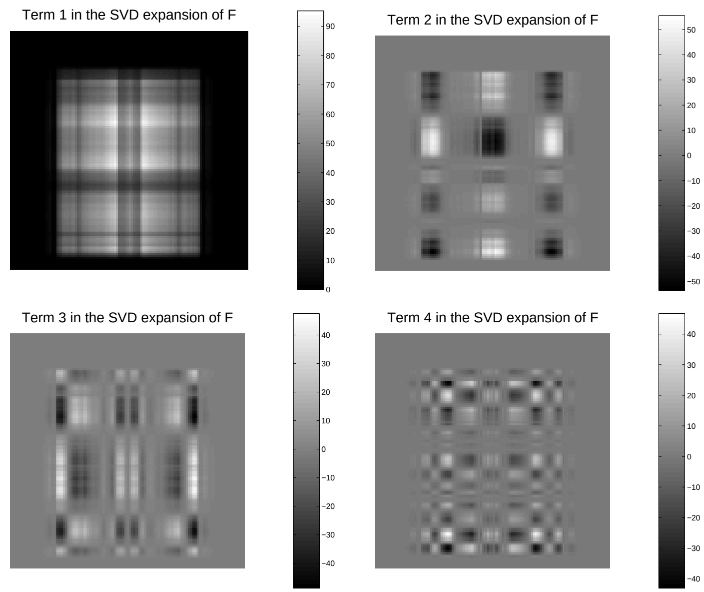
- Let's first get an idea of what the series terms look like.
- Then we'll sum these terms up to increasing values of N' and watch what happens.

Approximation of matrices using the SVD

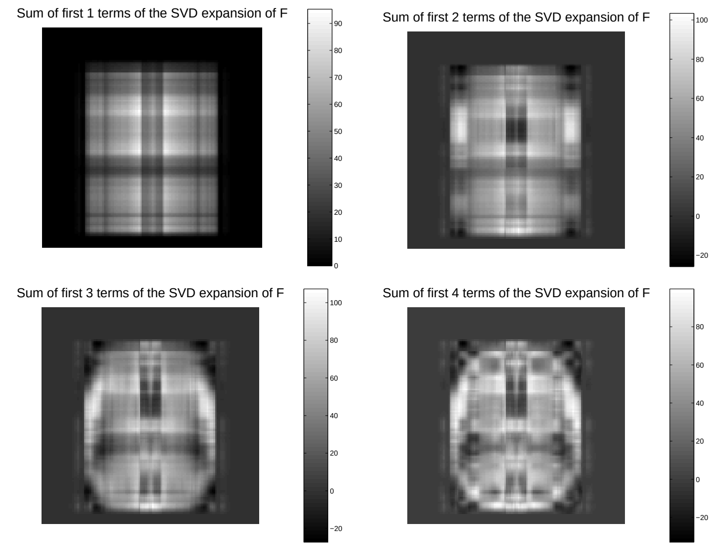
Matrix to be approximated (or image to be compressed)



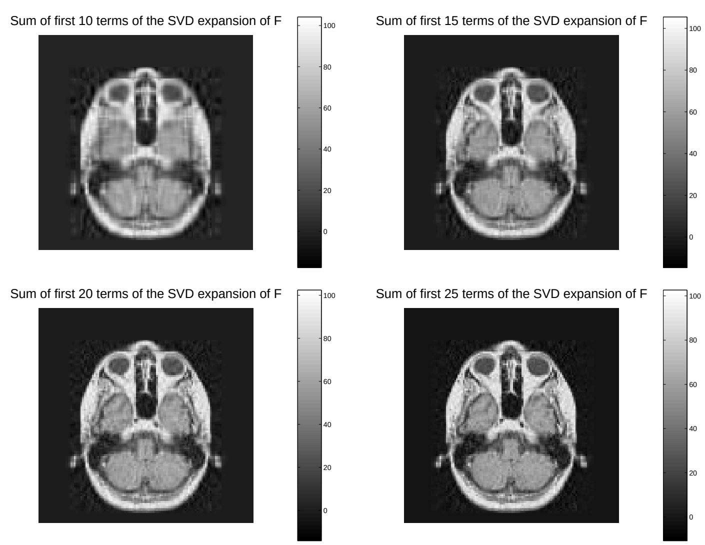
Approximation of matrices using the SVD



Approximation of matrices using the SVD



Approximation of matrices using the SVD



Approximation of matrices using the SVD

- Since $\mathbf{F}$ is a (128×128) image, we could have summed all 128 terms in the series.
- However, we see that by the time we sum up only the first 25 of these terms, the image looks very similar to the original.
- The original image required storage of $128 \times 128 = 16384$ numbers.
- The compressed image requires storage of the 25 $\mathbf{u}_n$ vectors of length 128, the 25 $\mathbf{v}_n$ vectors of length 128 and the 25 singular values. The storage requirement is thus $50 \times 128 + 25 = 6425$.
- We have compressed the image by a factor of 2.55, while not severely compromising image quality.
- How can we decide how many terms to use?

Approximation of matrices using the SVD

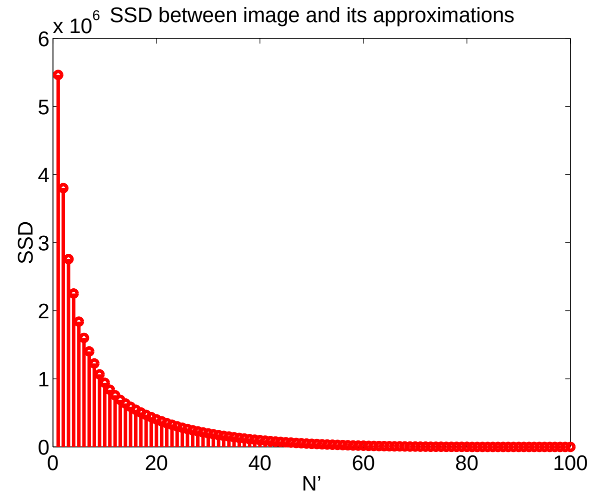
- We can compare two matrices $\mathbf{A}$ and $\mathbf{B}$, both having I rows and J columns, by calculating the sum of the squared differences (SSD):

$$C(\mathbf{A}, \mathbf{B}) = \sum_{i=1}^{I} \sum_{j=1}^{J} (A_{ij} - B_{ij})^2$$

  where A_{ij} is the element at the ith row and jth column of $\mathbf{A}$.

- Let $\mathbf{F}_{N'}$ be the approximation to $\mathbf{F}$ obtained when N' elements of the series are summed.
- We will plot $C(\mathbf{F}, \mathbf{F}_{N'})$ versus N'.

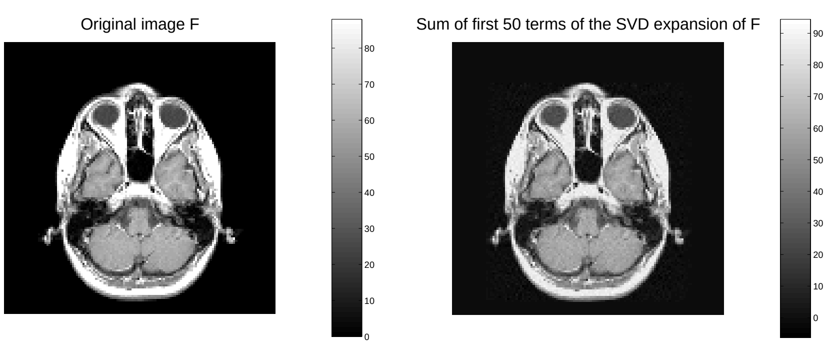
Approximation of matrices using the SVD



This graph tells us not much improvement will be observed if we sum more than ≈ 50 terms.

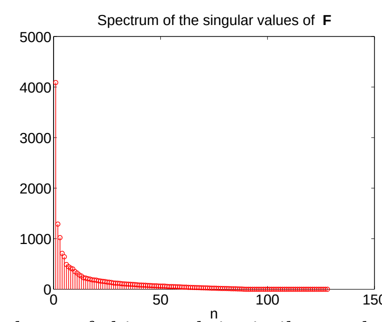
Approximation of matrices using the SVD

Now we visually compare $\mathbf{F}_{50}$ with $\mathbf{F}$:



Approximation of matrices using the SVD

Finally, we look at the spectrum of singular values:



- We see that the shape of this graph is similar to the SSD curve. The singular value spectrum tells us that the singular vectors whose index n is more than ≈ 50 do not contribute significant "energy" to the image.
- The SVD is excellent for image compression as it finds the most important independent basis vectors for a particular image. It also provides these vectors in order of decreasing importance. In what applications is this a useful property?

Approximation of matrices using the SVD: Matlab implementation

```
load mri % load Matlab's MRI image demo

F = double(D(:,:,1,5)); % select a 2D slice
                        % F is 128x128

[U,S,V] = svd(F,0);

s = diag(S);

imExpan = zeros(size(F));

NPrime = 100; % number of terms in our series
ssd = [];

for n = 1:NPrime

  u = U(:,n);
  v = V(:,n);
```



```
  im(:,:,n) = s(n) * u*v'; % evaluate this series element

  imExpan = imExpan + im(:,:,n); % add this term to series

  % calculate the SSD
  ssd(n) = sum( (imExpan(:) - F(:)).^2);

end % for n
```


Optional reading

1. "Numerical Recipes in C". Press, Teukolsky, Vetterling and Flannery, 2nd Edition (1995) pp. 59-70
2. "Matrix Computations". Golub and Van Loan, Third Edition (1996) pp. 70-75, 80, 253-264,
3. Jain, pp. 4-6, 176-180, 299-301.