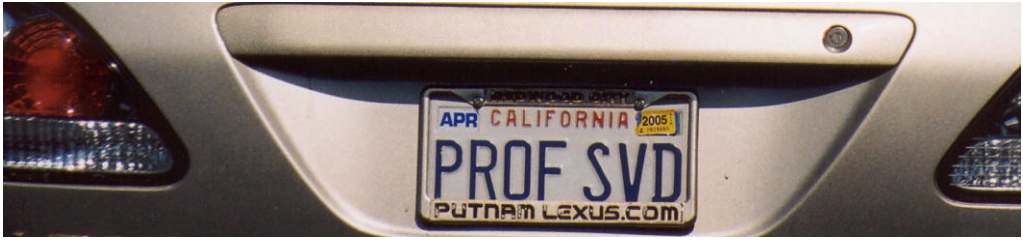
[11] The Singular Value Decomposition

The Singular Value Decomposition



Gene Golub's license plate, photographed by Professor P. M. Kroonenberg of Leiden University.

Frobenius norm for matrices

We have defined a norm for vectors over $\mathbb{R}$: $\|[x_1, x_2, \ldots, x_n]\| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$

Now we define a norm for matrices: interpret the matrix as a vector.

$$\|A\|_F = \sqrt{\text{sum of squares of elements of } A}$$

called the *Frobenius norm* of a matrix.

Squared norm is just sum of squares of the elements.

Example: $\left\| \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \right\|_F^2 = 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2$

Can group in terms of rows .... or columns

$$\left\| \begin{bmatrix} 1 & 2 & 3 \\ \hline 4 & 5 & 6 \end{bmatrix} \right\|_F^2 = (1^2 + 2^2 + 3^2) + (4^2 + 5^2 + 6^2) = \|[1,2,3]\|^2 + \|[4,5,6]\|^2$$

$$\left\| \left[\begin{array}{c|c|c} 1 & 2 & 3 \\ 4 & 5 & 6 \end{array}\right] \right\|_F^2 = (1^2 + 4^2) + (2^2 + 5^2) + (3^2 + 6^2) = \|[1,4]\|^2 + \|[2,5]\|^2 + \|[3,6]\|^2$$

Frobenius norm for matrices

Example: $\left\| \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \right\|_F^2 = 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2$

Can group in terms of rows .... or columns

$$\left\| \left[\begin{array}{ccc} 1 & 2 & 3 \\ \hline 4 & 5 & 6 \end{array}\right] \right\|_F^2 = (1^2 + 2^2 + 3^2) + (4^2 + 5^2 + 6^2) = \|[1,2,3]\|^2 + \|[4,5,6]\|^2$$

$$\left\| \left[\begin{array}{c|c|c} 1 & 2 & 3 \\ 4 & 5 & 6 \end{array}\right] \right\|_F^2 = (1^2 + 4^2) + (2^2 + 5^2) + (3^2 + 6^2) = \|[1,4]\|^2 + \|[2,5]\|^2 + \|[3,6]\|^2$$

Proposition: Squared Frobenius norm of a matrix is the sum of the squared norms of its rows ...

$$\left\| \begin{bmatrix} \mathbf{a}_1 \\ \hline \vdots \\ \hline \mathbf{a}_m \end{bmatrix} \right\|_F^2 = \|\mathbf{a}_1\|^2 + \cdots + \|\mathbf{a}_m\|^2$$

Frobenius norm for matrices

Example: $\left\| \left[\begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \end{array}\right] \right\|_F^2 = 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2$

Can group in terms of rows .... or columns

$$\left\| \left[\begin{array}{ccc} 1 & 2 & 3 \\ \hline 4 & 5 & 6 \end{array}\right] \right\|_F^2 = (1^2 + 2^2 + 3^2) + (4^2 + 5^2 + 6^2) = \|[1,2,3]\|^2 + \|[4,5,6]\|^2$$

$$\left\| \left[\begin{array}{c|c|c} 1 & 2 & 3 \\ 4 & 5 & 6 \end{array}\right] \right\|_F^2 = (1^2 + 4^2) + (2^2 + 5^2) + (3^2 + 6^2) = \|[1,4]\|^2 + \|[2,5]\|^2 + \|[3,6]\|^2$$

Proposition: Squared Frobenius norm of a matrix is the sum of the squared norms of its rows ... or of its columns.

$$\left\| \left[\begin{array}{c|c|c|c} & & & \\ \mathbf{v}_1 & \cdots & \mathbf{v}_n & \\ & & & \end{array}\right] \right\|_F^2 = \|\mathbf{v}_1\|^2 + \cdots + \|\mathbf{v}_n\|^2$$

Low-rank matrices

Saving space and saving time

$$\begin{bmatrix} \mathbf{u} \\ \\ \end{bmatrix} \begin{bmatrix} & \mathbf{v}^T & \end{bmatrix}$$

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

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

Silly compression

Represent a grayscale $m \times n$ image by an $m \times n$ matrix A. (Requires mn numbers to represent.) Find a low-rank matrix $\tilde{A}$ that is as close as possible to A. (For rank r, requires only $r(m+n)$ numbers to represent.)

Original image (625×1024, so about 625k numbers)

Silly compression

Represent a grayscale $m \times n$ image by an $m \times n$ matrix A. (Requires mn numbers to represent.) Find a low-rank matrix $\tilde{A}$ that is as close as possible to A. (For rank r, requires only $r(m+n)$ numbers to represent.)

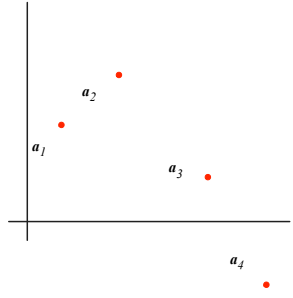
Rank-50 approximation (so about 82k numbers)

The *trolley-line-location* problem

Given the locations of m houses $\mathbf{a}_1, \ldots, \mathbf{a}_m$, we must choose where to run a trolley line.

The trolley line must go through downtown (origin) and must be a straight line.

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



Specify line by unit-norm vector $\mathbf{v}$: line is Span $\{\mathbf{v}\}$.

In measuring objective, how to combine individual objectives?

As in least squares, we minimize the 2-norm of the vector $[d_1, \ldots, d_m]$ of distances.

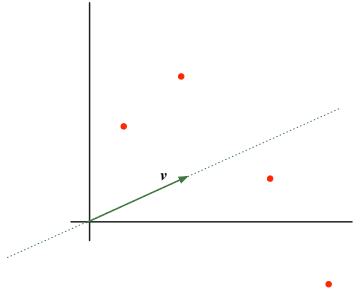
Equivalent to minimizing the square of the 2-norm of this vector, i.e. $d_1^2 + \cdots + d_m^2$.

The *trolley-line-location* problem

Given the locations of m houses $\mathbf{a}_1, \ldots, \mathbf{a}_m$, we must choose where to run a trolley line.

The trolley line must go through downtown (origin) and must be a straight line.

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



Specify line by unit-norm vector $\mathbf{v}$: line is Span $\{\mathbf{v}\}$.

In measuring objective, how to combine individual objectives?

As in least squares, we minimize the 2-norm of the vector $[d_1, \ldots, d_m]$ of distances.

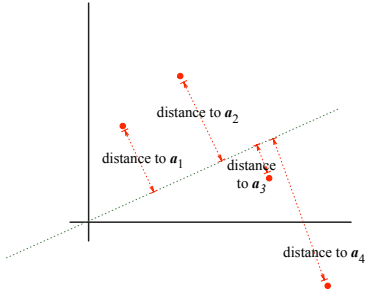
Equivalent to minimizing the square of the 2-norm of this vector, i.e. $d_1^2 + \cdots + d_m^2$.

The *trolley-line-location* problem

Given the locations of m houses $\mathbf{a}_1, \ldots, \mathbf{a}_m$, we must choose where to run a trolley line.

The trolley line must go through downtown (origin) and must be a straight line.

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



Specify line by unit-norm vector $\mathbf{v}$: line is Span $\{\mathbf{v}\}$.

In measuring objective, how to combine individual objectives?

As in least squares, we minimize the 2-norm of the vector $[d_1, \ldots, d_m]$ of distances.

Equivalent to minimizing the square of the 2-norm of this vector, i.e. $d_1^2 + \cdots + d_m^2$.

Solution to the *trolley-line-location* problem

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

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

By the Pythagorean Theorem,

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

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

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

---

using $\mathbf{a}_i^{\|\mathbf{v}} = \langle \mathbf{a}_i, \mathbf{v} \rangle \mathbf{v}$ and hence $\|\mathbf{a}_i^{\|\mathbf{v}}\|^2 = \langle \mathbf{a}_i, \mathbf{v} \rangle^2 \|\mathbf{v}\|^2$

Solution to the *trolley-line-location* problem

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

By the Pythagorean Theorem,

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

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

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

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

using $\mathbf{a}_i^{\parallel \mathbf{v}} = \langle \mathbf{a}_i, \mathbf{v}\rangle \ \mathbf{v}$ and hence $\|\mathbf{a}_i^{\parallel \mathbf{v}}\|^2 = \langle \mathbf{a}_i, \mathbf{v}\rangle^2 \ \|\mathbf{v}\|^2 = \langle \mathbf{a}_i, \mathbf{v}\rangle^2$

Solution to the *trolley-line-location* problem, continued

$$\sum_i (\text{dist from } \mathbf{a}_i \text{ to Span } \{\mathbf{v}\})^2 = \|A\|_F^2 \quad - \quad \left(\langle \mathbf{a}_1, \mathbf{v} \rangle^2 + \cdots + \langle \mathbf{a}_m, \mathbf{v} \rangle^2 \right)$$

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

$$\begin{bmatrix} \mathbf{a}_1 \\ \hline \vdots \\ \hline \mathbf{a}_m \end{bmatrix} \begin{bmatrix} \\ \mathbf{v} \\ \\ \end{bmatrix} = \begin{bmatrix} \langle \mathbf{a}_1, \mathbf{v} \rangle \\ \vdots \\ \langle \mathbf{a}_m, \mathbf{v} \rangle \end{bmatrix} \tag{1}$$

so

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

Substituting into Equation 1, we obtain

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

Therefore the best vector $\mathbf{v}$ is a unit vector that maximizes $||A\mathbf{v}||^2$ (equivalently, maximizes $||A\mathbf{v}||$).

Solution to the *trolley-line-location* problem, continued

$$\sum_i (\text{distance from } \mathbf{a}_i \text{ to Span } \{\mathbf{v}\})^2 \quad = \quad ||A||_F^2 \quad - \quad \|A\mathbf{v}\|^2$$

Therefore the best vector $\mathbf{v}$ is a unit vector that maximizes $||A\mathbf{v}||^2$ (equivalently, maximizes $||A\mathbf{v}||$).

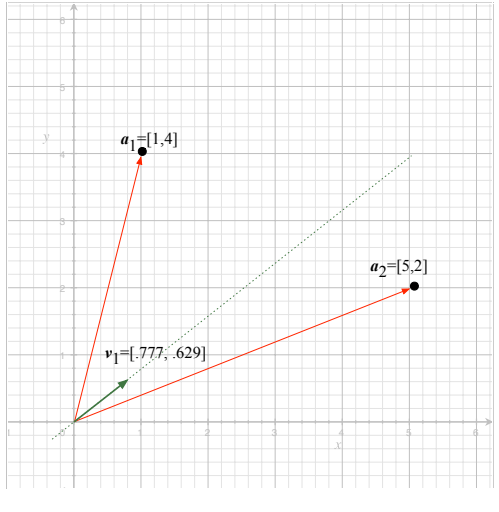
`def trolley_line_location`(A):
    $\mathbf{v}_1 = \arg\max\{||A\mathbf{v}|| \ : \ ||\mathbf{v}|| = 1\}$
    $\sigma_1 = ||A\mathbf{v}_1||$
    return $\mathbf{v}_1$

So far, this is a solution only in *principle* since we have not specified how to actually compute $\mathbf{v}_1$.

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

Trolley-line-location problem, example

Example: Let $A = \begin{bmatrix} 1 & 4 \\ 5 & 2 \end{bmatrix}$, so $\mathbf{a}_1 = [1, 4]$ and $\mathbf{a}_2 = [5, 2]$. In this case, a unit vector maximizing $||A\mathbf{v}||$ is $\mathbf{v}_1 \approx \begin{bmatrix} .78 \\ .63 \end{bmatrix}$. We use σ_1 to denote $||A\mathbf{v}_1||$, which is about 6.1:



Theorem

```
def trolley_line_location(A):
    v_1 = arg max{||Av|| : ||v|| = 1}
    σ_1 = ||Av_1||
    return v_1
```

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

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

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

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

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

Proof: The distance is $\sum_i ||\mathbf{a}_i||^2 - \sum_i ||\mathbf{a}_i^{\parallel \mathbf{v}}||^2$.
The first sum is $||A||_F^2$.
The second sum is the square of the quantity $||A\mathbf{v}_1||$, a quantity we have named σ_1.

Example, continued

Let $A = \begin{bmatrix} 1 & 4 \\ 5 & 2 \end{bmatrix}$, so $\mathbf{a}_1 = [1, 4]$ and $\mathbf{a}_2 = [5, 2]$. **Solution:** $\mathbf{v}_1 \approx \begin{bmatrix} .78 \\ .63 \end{bmatrix}$.

We next calculate the sum of squared distances:

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

$$\begin{aligned} \mathbf{a}_1 - \langle \mathbf{a}_1, \mathbf{v}_1 \rangle \, \mathbf{v}_1 &\approx [1, 4] - (1 \cdot .78 + 4 \cdot .63)[.78, .63] \\ &\approx [1, 4] - 3.3\,[.78, .63] \\ &\approx [-1.6, 1.9] \end{aligned}$$

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

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

$$\begin{aligned} \mathbf{a}_2 - \langle \mathbf{a}_1, \mathbf{v}_1 \rangle \, \mathbf{v}_1 &\approx [5, 2] - (5 \cdot .78 + 2 \cdot .63)[.78, .63] \\ &\approx [5, 2] - 5.1\,[.78, .63] \\ &\approx [1, -1.2] \end{aligned}$$

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

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

The sum of squared distances should be $||A||_F^2 - \sigma_1^2$.

$||A||_F^2 = 1^2 + 4^2 + 5^2 + 2^2 = 46$ and $\sigma_1 \approx 6.1$ so $||A||_F^2 - \sigma_1^2$ is about 8.7. ✓

Application to voting data

Let $\mathbf{a}_1, \ldots, \mathbf{a}_{100}$ be the voting records for US Senators.
Same as you used in politics lab.
These are 46-vectors with ± 1 entries.
Find the unit-norm vector $\mathbf{v}$ that minimizes least-squares distance from $\mathbf{a}_1, \ldots, \mathbf{a}_{100}$ to Span $\{\mathbf{v}\}$.
Look at projection along $\mathbf{v}$ of each of these vectors.

Not so meaningful:

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	not so moderate Republican from Idaho
Vitter	0.108031374	not so moderate Republican from Louisiana

We'll have to come back to this data.

Best rank-one approximation to a matrix

A rank-one matrix is a matrix whose row space is one-dimensional.
All rows must lie in Span $\{\mathbf{v}\}$ for some vector $\mathbf{v}$.
That is, every row is a scalar multiple of $\mathbf{v}$.

A rank-one matrix can be written as $\begin{bmatrix} \mathbf{u} \\ \\ \end{bmatrix} \begin{bmatrix} \quad \mathbf{v} \quad \end{bmatrix}$

Goal: Given matrix A, find the rank-one matrix $\tilde{A}$ that minimizes $\|A - \tilde{A}\|_F$.

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

How close is $\tilde{A}$ to A?

$$\begin{aligned} ||A - \tilde{A}||_F^2 &= \sum_i ||\text{row } i \text{ of } A - \tilde{A}||^2 \\ &= \sum_i ||\text{distance from } \mathbf{a}_i \text{ to Span } \{\mathbf{v}\}||^2 \end{aligned}$$

To minimize the sum of squares of distances, choose $\mathbf{v}$ to be first right singular vector. Sum of squared distances is $||A||_F^2 - \sigma_1^2$.
$\tilde{A}$= closest rank-one matrix.

An expression for the best rank-one approximation

Using the formula $\mathbf{a}_i^{\parallel \mathbf{v}_1} = \langle \mathbf{a}_i, \mathbf{v}_1 \rangle \mathbf{v}_1$, we obtain

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

The first vector in the outer product can be written as $A\mathbf{v}_1$. We obtain

$$\tilde{A} = \begin{bmatrix} \\ A\mathbf{v}_1 \\ \\ \end{bmatrix} \begin{bmatrix} & \mathbf{v}_1^T & \end{bmatrix}$$

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

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

Remember $\sigma_1 = \|A\mathbf{v}_1\|$. Define $\mathbf{u}_1$ to be the norm-one vector such that $\sigma_1 \mathbf{u}_1 = A\mathbf{v}_1$. Then

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

Best rank-one approximation

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

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

Best rank-one approximation: example

Example: For the matrix $A = \begin{bmatrix} 1 & 4 \\ 5 & 2 \end{bmatrix}$, the first right singular vector is $\mathbf{v}_1 \approx \begin{bmatrix} .78 \\ .63 \end{bmatrix}$ and the first singular value σ_1 is about 6.1. The first left singular vector is $\mathbf{u}_1 \approx \begin{bmatrix} .54 \\ .84 \end{bmatrix}$, meaning $\sigma_1 \mathbf{u}_1 = A\mathbf{v}_1$.

We then have

$$\begin{aligned} \tilde{A} &= \sigma_1 \mathbf{u}_1 \mathbf{v}_1^T \\ &\approx 6.1 \begin{bmatrix} .54 \\ .84 \end{bmatrix} \begin{bmatrix} .78 & .63 \end{bmatrix} \\ &\approx \begin{bmatrix} 2.6 & 2.1 \\ 4.0 & 3.2 \end{bmatrix} \end{aligned}$$

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

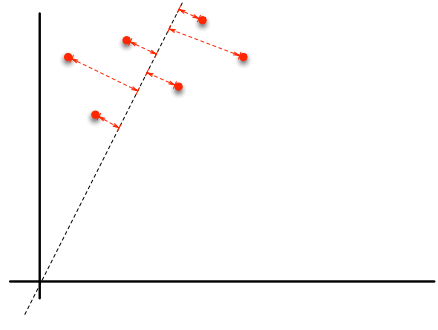
$$1.56^2 + 1.93^2 + 1^2 + 1.23^2 \approx 8.7$$

$$||A - \tilde{A}||_F^2 = ||A||_F^2 - \sigma_1^2 \approx 8.7. \checkmark$$

The closest one-dimensional affine space

In *trolley-line problem*, line must go through origin: closest one-dimensional *vector space*.
Perhaps line *not* through origin is much closer.

An arbitrary line (one not necessarily passing through the origin) is a one-dimensional *affine* space.



Given points $\mathbf{a}_1, \ldots, \mathbf{a}_m$,

- choose point $\bar{\mathbf{a}}$ and translate each of the input points by subtracing $\bar{\mathbf{a}}$:
$$\mathbf{a}_1 - \bar{\mathbf{a}}, \ldots, \mathbf{a}_m - \bar{\mathbf{a}}$$
- find the one-dimensional vector space closest to these translated points, and then translate that vector space by adding back $\bar{\mathbf{a}}$.

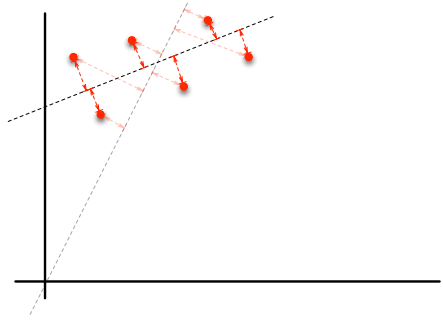
Best choice of $\bar{\mathbf{a}}$ is the *centroid* of the input points, the vector $\bar{\mathbf{a}} = \frac{1}{m}(\mathbf{a}_1 + \cdots + \mathbf{a}_m)$

Translating the points by subtracting off the centroid is called *centering* the points.

The closest one-dimensional affine space

In *trolley-line problem*, line must go through origin: closest one-dimensional *vector space*.
Perhaps line *not* through origin is much closer.

An arbitrary line (one not necessarily passing through the origin) is a one-dimensional *affine* space.



Given points $\mathbf{a}_1, \ldots, \mathbf{a}_m$,

- choose point $\bar{\mathbf{a}}$ and translate each of the input points by subtracing $\bar{\mathbf{a}}$:
$$\mathbf{a}_1 - \bar{\mathbf{a}}, \ldots, \mathbf{a}_m - \bar{\mathbf{a}}$$
- find the one-dimensional vector space closest to these translated points, and then translate that vector space by adding back $\bar{\mathbf{a}}$.

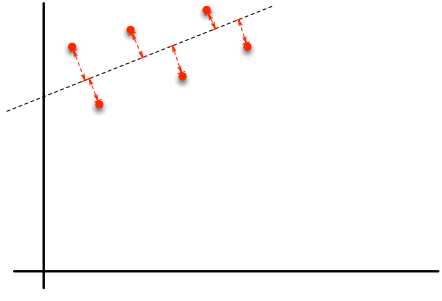
Best choice of $\bar{\mathbf{a}}$ is the *centroid* of the input points, the vector $\bar{\mathbf{a}} = \frac{1}{m}(\mathbf{a}_1 + \cdots + \mathbf{a}_m)$

Translating the points by subtracting off the centroid is called *centering* the points.

The closest one-dimensional affine space

In *trolley-line problem*, line must go through origin: closest one-dimensional *vector space*.
Perhaps line *not* through origin is much closer.

An arbitrary line (one not necessarily passing through the origin) is a one-dimensional *affine* space.



Given points $\mathbf{a}_1, \ldots, \mathbf{a}_m$,

- choose point $\bar{\mathbf{a}}$ and translate each of the input points by subtracing $\bar{\mathbf{a}}$:
$$\mathbf{a}_1 - \bar{\mathbf{a}}, \ldots, \mathbf{a}_m - \bar{\mathbf{a}}$$
- find the one-dimensional vector space closest to these translated points, and then translate that vector space by adding back $\bar{\mathbf{a}}$.

Best choice of $\bar{\mathbf{a}}$ is the *centroid* of the input points, the vector $\bar{\mathbf{a}} = \frac{1}{m}(\mathbf{a}_1 + \cdots + \mathbf{a}_m)$

Translating the points by subtracting off the centroid is called *centering* the points.

Politics revisited

We center the voting data, and find the closest one-dimensional vector space Span $\{\mathbf{v}_1\}$.
Now projection along $\mathbf{v}$ gives better spread.

Which 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'}
```

Similarly, only three of the senators to the right of the origin are Democrat.

Principal component

Closest line is the axis of *maximum variance*.
Given vectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m$ in $\mathbb{R}^n$, choose a line in $\mathbb{R}$ to maximize

$$(f(\mathbf{a}_1) - \mu)^2 + (f(\mathbf{a}_2) - \mu)^2 + \cdots + (f(\mathbf{a}_m) - \mu)^2$$

where $\mu = \frac{1}{m}(f(\mathbf{a}_1) + f(\mathbf{a}_2) + \cdots + f(\mathbf{a}_m))$
and $f(\mathbf{x})$ is the point on the line closest to $\mathbf{x}$

Good way to visualize data spread across one dimension.

What about two dimensions or more?

Closest dimension-k vector space

Computational Problem: *closest low-dimensional subspace:*

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

$$\sum_i (\text{distance from } \mathbf{a}_i \text{ to } \mathcal{V}_k)^2$$

algorithm for one dimension: choose unit-norm vector $\mathbf{v}$ that maximizes $||A\mathbf{v}||$
Natural generalization of this algorithm in which an *orthonormal* basis is sought.

In the i^{th} iteration, the vector $\mathbf{v}$ selected is the one that maximizes $||A\mathbf{v}||$ subject to being orthogonal to all previously selected vectors:

• Let $\mathbf{v}_1$ be the norm-one vector $\mathbf{v}$ maximizing $||A\mathbf{v}||$,
• let $\mathbf{v}_2$ be the norm-one vector $\mathbf{v}$ orthog. to $\mathbf{v}_1$ that maximizes $||A\mathbf{v}||$,
• let $\mathbf{v}_3$ be the norm-one vector $\mathbf{v}$ orthog. to $\mathbf{v}_1$ and $\mathbf{v}_2$ that maximizes $||A\mathbf{v}||$,
and so on.

```
def find_right_singular_vectors(A):
  for i = 1, 2, ..., min{m, n}
    v_i = arg max{||Av|| : ||v|| = 1,
          v is orthog. 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.
```

```
def find_right_singular_vectors(A):
  for i = 1, 2, ..., min{m, n}
    v_i = arg max{||Av|| : ||v|| = 1,
          v is orthog. 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]
```

Proposition: The singular values are positive and in noninreasing order.

Proof: $\sigma_i = \|A\mathbf{v}_i\|$ and norm of a vector is nonnegative. The algorithm stops before it would choose a vector $\mathbf{v}_i$ such that $\|A\mathbf{v}_i\|$ is zero, so the singular values are positive. First right singular vector is chosen most freely, followed by second, etc. QED

Proposition: Right singular vectors are orthonormal.

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

Theorem: Let A be an $m \times n$ matrix, and let $\mathbf{a}_1, \ldots, \mathbf{a}_m$ be its rows. Let $\mathbf{v}_1, \ldots, \mathbf{v}_r$ be its right singular vectors, and let $\sigma_1, \ldots, \sigma_r$ be its singular values. For $k = 1, 2, \ldots, r$, Span $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ is the k-dimensional vector space $\mathcal{V}$ that minimizes

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

Proposition: Left singular vectors $\mathbf{u}_1, \ldots, \mathbf{u}_t$ are orthonormal. (See text for proof.)

Closest k-dimensional affine space

Use the centering technique:
Find the centroid $\bar{\mathbf{a}}$ of the input points $\mathbf{a}_1, \ldots, \mathbf{a}_m$, and subtract it from each of the input points. Then find a basis $\mathbf{v}_1, \ldots, \mathbf{v}_k$ for the k-dimensional vector space closest to $\mathbf{a}_1 - \bar{\mathbf{a}}, \ldots, \mathbf{a}_m - \bar{\mathbf{a}}$. The k-dimensional affine space closest to the original points $\mathbf{a}_1, \ldots, \mathbf{a}_m$ is

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

Deriving the Singular Value Decomposition

Let A be an $m \times n$ matrix.

We have defined a procedure to obtain

$\mathbf{v}_1, \ldots, \mathbf{v}_r$	the right singular vectors	orthonormal by choice
$\sigma_1, \ldots, \sigma_r$	the singular values	positive
$\mathbf{u}_1, \ldots, \mathbf{u}_r$	the left singular vectors	orthonormal by Proposition

such that $\sigma_i \mathbf{u}_i = A\mathbf{v}_i$ for $i = 1, \ldots, r$.

Express equations using matrix-matrix multiplication:

$$\left[\begin{array}{c|c|c} \sigma_1\mathbf{u}_1 & \cdots & \sigma_r\mathbf{u}_r \end{array}\right] = \left[\begin{array}{c} \\ A \\ \\ \end{array}\right] \left[\begin{array}{c|c|c} \mathbf{v}_1 & \cdots & \mathbf{v}_r \end{array}\right]$$

We rewrite equation as

$$\left[\begin{array}{c|c|c} \mathbf{u}_1 & \cdots & \mathbf{u}_r \end{array}\right] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix} = \left[\begin{array}{c} \\ A \\ \\ \end{array}\right] \left[\begin{array}{c|c|c} \mathbf{v}_1 & \cdots & \mathbf{v}_r \end{array}\right]$$

Deriving the Singular Value Decomposition

We rewrite equation as

$$\begin{bmatrix} \mathbf{u}_1 & \cdots & \mathbf{u}_r \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix} = \begin{bmatrix} & A & \end{bmatrix} \begin{bmatrix} \mathbf{v}_1 & \cdots & \mathbf{v}_r \end{bmatrix}$$

Assume number r of singular values is n. Then the rightmost matrix is square and its columns are orthonormal, so it is an orthogonal matrix, so its inverse is its transpose. Multiplying both sides of equation, we obtain

$$\begin{bmatrix} & A & \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1 & \cdots & \mathbf{u}_n \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \hline \vdots \\ \hline \mathbf{v}_n^T \end{bmatrix}$$

$$A = U\Sigma V^T$$

where U and V are column-orthogonal and Σ is diagonal with positive diagonal elements.

called the (compact) *singular value decomposition* (SVD) of A.

The Singular Value Decomposition

The (compact) SVD of a matrix A is the factorization of A as

$$A = U\Sigma V^T$$

where U and V are column-orthogonal and Σ is diagonal with positive diagonal elements.

In general, Σ is allowed to have zero diagonal elements.

Traditionally, the term *SVD* refers to a decomposition in which U and V are both square, and Σ consists of a diagonal matrix with extra rows (if A has more rows than columns) or extra columns (if A has more columns than rows).

The term *reduced SVD* is used when Σ is required to be a diagonal matrix. We use just the reduced SVD.

Existence of the Singular Value Decomposition

Theorem: Every matrix A has a (reduced) SVD

We outlined a construction using the procedure `find_right_singular_vectors`(A). We made the assumption that the number of iterations equals the number of columns of A. For a more general proof, see the text.

SVD of the transpose

We can go from the SVD of A to the SVD of A^T.

$$\begin{bmatrix} & A & \end{bmatrix} = \begin{bmatrix} & U & \end{bmatrix} \begin{bmatrix} & \Sigma & \end{bmatrix} \begin{bmatrix} & V^T & \end{bmatrix}$$

Define $\bar{U} = V$ and $\bar{V} = U$. Then

$$\begin{bmatrix} & A^T & \end{bmatrix} = \begin{bmatrix} & \bar{U} & \end{bmatrix} \begin{bmatrix} & \Sigma & \end{bmatrix} \begin{bmatrix} & \bar{V}^T & \end{bmatrix}$$

Best rank-k approximation in terms of the singular value decomposition

Start by writing SVD of A:

$$\begin{bmatrix} & A & \end{bmatrix} = \begin{bmatrix} & U & \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_1 \end{bmatrix} \begin{bmatrix} & V^T & \end{bmatrix}$$

Replace $\sigma_{k+1}, \ldots, \sigma_n$ with zeroes. We obtain

$$\begin{bmatrix} & \tilde{A} & \end{bmatrix} = \begin{bmatrix} & U & \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix} \begin{bmatrix} & V^T & \end{bmatrix}$$

This gives the same approximation as before.

Example: Senators

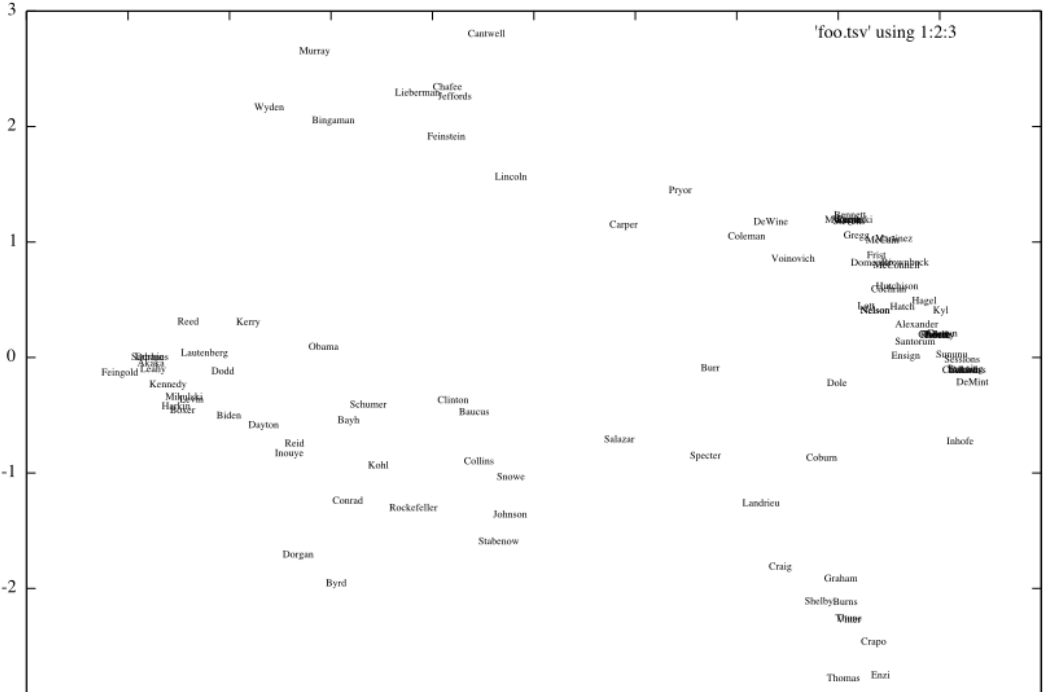
First center the data. Then find first two right singular vectors $\mathbf{v}_1$ and $\mathbf{v}_2$.
Projecting onto these gives two coordinates.
To find singular vectors,

- make a matrix A whose rows are the centered versions of vectors
- find SVD of A using `svd` module.

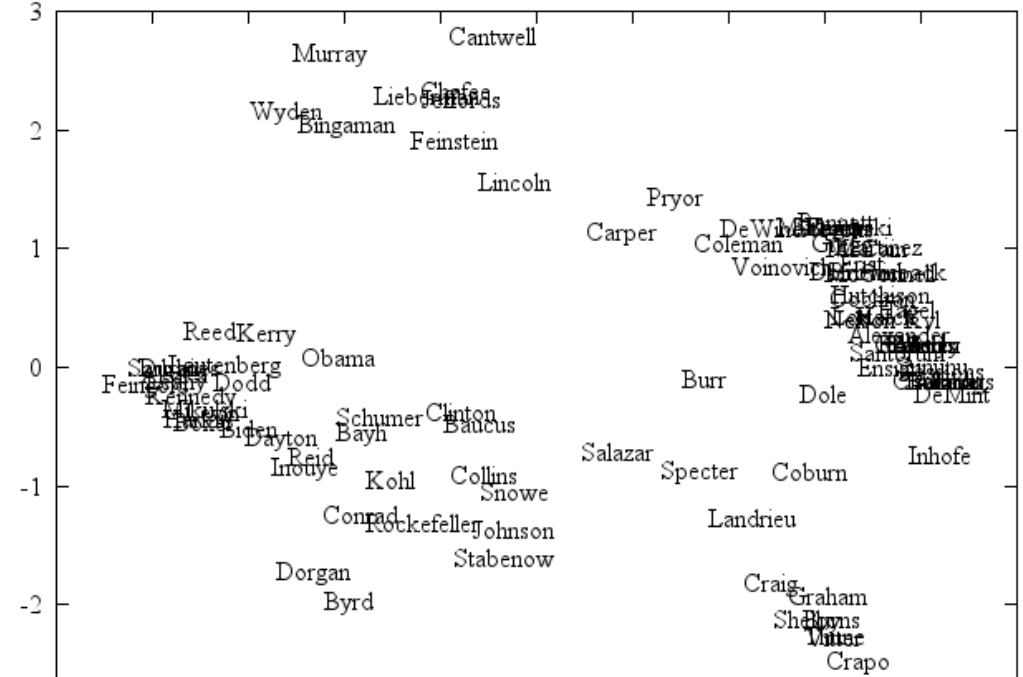
```
>>> U, Sigma, V = svd.factor(A)
```

- first two columns of V are first two right singular vectors.

Example: Senators, two principal components



Example: Senators, two principal components



Function interpretation of SVD

$A = U\Sigma V^T$

The function $\mathbf{x} \mapsto A\mathbf{x}$ can be written as the composition of three functions:

- $\mathbf{x} \mapsto V^T\mathbf{x}$
- $\mathbf{y} \mapsto \Sigma\mathbf{y}$
- $\mathbf{z} \mapsto U\mathbf{z}$

Assuming number of rows of A is at least number of columns, here's an interpretation:

- (`vec2rep`) $\mathbf{x} \mapsto V^T\mathbf{x}$ means: "given a vector $\mathbf{x}$, find its coordinate representation in terms of the columns of V."
- (scaling of coordinates) $y \mapsto \Sigma\mathbf{y}$ means: "given a coordinate representation, scale the coordinates by some numbers (the diagonal elements of Σ)"
- (`rep2vec`) $\mathbf{z} \mapsto U\mathbf{z}$ means: "given some coordinates, interpret those coordinates as coefficients of the columns of U, and find the corresponding vector."

So, for any $m \times n$ matrix A with $m \geq n$, multiplication of a vector by A can be interpreted as:

- find the coordinates of the vector in terms of one orthonormal basis,
- scale those coordinates, and
- find the vector with the scaled coordinates over another orthonormal basis.

Uses of SVD

The most famous use of SVD is in principal components analysis and its cousins. However, SVD is useful for more prosaic problems:

- Computing rank: rank is the number of singular values above some small specified tolerance.
- Useful in computing orthonormal bases of Null A and Col A.
- least-squares: unlike QR decomposition, SVD can be used even when matrix A does not have linearly independent columns.

Least squares via SVD

Algorithm for finding minimizer of $\|\mathbf{b} - A\mathbf{x}\|$:

> Find singular value decomposition (U, Σ, V) of A
> return $V\Sigma^{-1}U^T\mathbf{b}$

Justification: Let $\hat{x}$ be the vector returned by the algorithm.

$$\begin{aligned} A\hat{\mathbf{x}} &= (U\Sigma V^T)(V\Sigma^{-1}U^T\mathbf{b}) \\ &= U\Sigma\Sigma^{-1}U^T b \\ &= UU^T\mathbf{b} \\ &= U(\text{coord. repr. of } \mathbf{b}^{||} \text{ in terms of cols of } U) \\ &= \mathbf{b}^{||} \end{aligned}$$