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

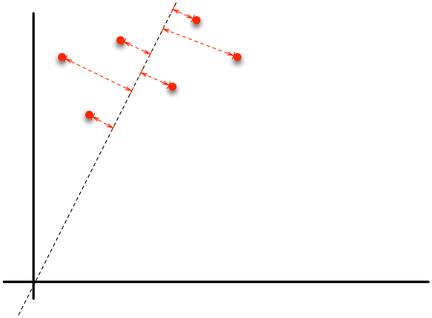
$$1.56^2 + 1.93^2 + 1^2 + 1.23^2 \approx 8.7$$

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

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.



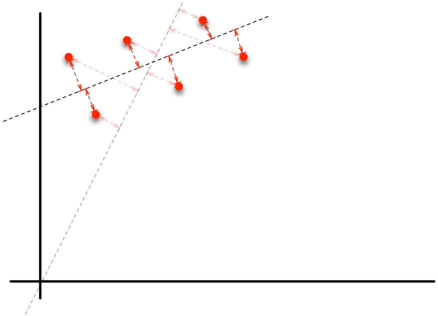
Given points $\mathbf{a}_1, \ldots, \mathbf{a}_m$,

- choose point $\bar{\mathbf{a}}$ and translate each of the input points by subtracting $\bar{\mathbf{a}}$:
$$\mathbf{a}_1 - \bar{\mathbf{a}}, \ldots, \mathbf{a}_m - \bar{\mathbf{a}}$$
- 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 $\bar{\mathbf{a}} = \frac{1}{m}(\mathbf{a}_1 + \cdots + \mathbf{a}_m)$.
(Proof is lovely–maybe we'll see it later.)

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.

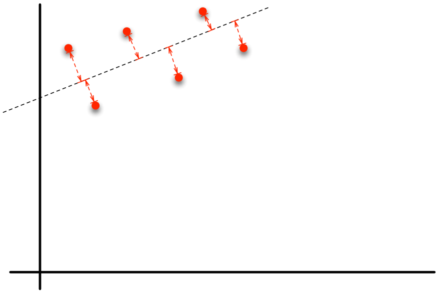
Given points $\mathbf{a}_1, \ldots, \mathbf{a}_m$,

- choose point $\bar{\mathbf{a}}$ and translate each of the input points by subtracting $\bar{\mathbf{a}}$:
$$\mathbf{a}_1 - \bar{\mathbf{a}}, \ldots, \mathbf{a}_m - \bar{\mathbf{a}}$$
- 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 $\bar{\mathbf{a}} = \frac{1}{m}(\mathbf{a}_1 + \cdots + \mathbf{a}_m)$.
(Proof is lovely–maybe we'll see it later.)

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 subtracting $\bar{\mathbf{a}}$:
$$\mathbf{a}_1 - \bar{\mathbf{a}}, \ldots, \mathbf{a}_m - \bar{\mathbf{a}}$$
- 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 $\bar{\mathbf{a}} = \frac{1}{m}(\mathbf{a}_1 + \cdots + \mathbf{a}_m)$.
(Proof is lovely–maybe we'll see it later.)

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. Look at coordinate representation in terms of $\mathbf{v}$:

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.

Visualization revisited

We now can turn a bunch of high-dimensional vectors into a bunch of numbers, plot the numbers on number line. Dimension reduction

What about turning a bunch of high-dimensional vectors into vectors in $\mathbb{R}^2$ or $\mathbb{R}^3$ or $\mathbb{R}^{10}$?

Closest 1-dimensional vector space (trolley-line-location problem):

- *input:* Vectors $\mathbf{a}_1, \ldots \mathbf{a}_m$
- *output:* Orthonormal basis $\{\mathbf{v}_1\}$ for dim-1 vector space $\mathcal{V}_1$ that minimizes $\sum_i (\text{distance from } \mathbf{a}_i \text{ to } \mathcal{V}_1)^2$

We saw: $\sum_i (\text{distance from } \mathbf{a}_i \text{ to Span } \{\mathbf{v}_1\})^2 = ||A||_F^2 - \|A\mathbf{v}_1\|^2$

Therefore: Best vector $\mathbf{v}_1$ is the unit vector that maximizes $\|A\mathbf{v}_1\|$.

Closest k-dimensional vector space:

- *input:* Vectors $\mathbf{a}_1, \ldots \mathbf{a}_m$, integer k
- *output:* Orthonormal basis $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ for dim-k vector space $\mathcal{V}_k$ that minimizes $\sum_i (\text{distance from } \mathbf{a}_i \text{ to } \mathcal{V}_k)^2$

Let $\mathbf{v}_1, \ldots, \mathbf{v}_k$ be an orthonormal basis for a subspace $\mathcal{V}$

$$\mathbf{a}_1^{\perp\mathcal{V}} = \mathbf{a}_1 - \mathbf{a}_1^{\|\mathcal{V}}$$
$$\vdots$$
$$\mathbf{a}_m^{\perp\mathcal{V}} = \mathbf{a}_m - \mathbf{a}_m^{\|\mathcal{V}}$$

By the Pythagorean Theorem,

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

Closest 1-dimensional vector space (trolley-line-location problem):

- *input:* Vectors $\mathbf{a}_1, \ldots \mathbf{a}_m$
- *output:* Orthonormal basis $\{\mathbf{v}_1\}$ for dim-1 vector space $\mathcal{V}_1$ that minimizes $\sum_i (\text{distance from } \mathbf{a}_i \text{ to } \mathcal{V}_1)^2$

We saw: $\sum_i (\text{distance from } \mathbf{a}_i \text{ to Span } \{\mathbf{v}_1\})^2 = ||A||_F^2 - \|A\mathbf{v}_1\|^2$

Therefore: Best vector $\mathbf{v}_1$ is the unit vector that maximizes $\|A\mathbf{v}_1\|$.

Closest k-dimensional vector space:

- *input:* Vectors $\mathbf{a}_1, \ldots \mathbf{a}_m$, integer k
- *output:* Orthonormal basis $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ for dim-k vector space $\mathcal{V}_k$ that minimizes $\sum_i (\text{distance from } \mathbf{a}_i \text{ to } \mathcal{V}_k)^2$

Let $\mathbf{v}_1, \ldots, \mathbf{v}_k$ be an orthonormal basis for a subspace $\mathcal{V}$

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

By the Pythagorean Theorem,

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

Thus For an orthonormal basis $\mathbf{v}_1, \ldots, \mathbf{v}_k$ of $\mathcal{V}$,

$$\sum_i (\text{dist from } \mathbf{a}_i \text{ to } \mathcal{V})^2 = \|A\|_F^2 - \left(\|A\mathbf{v}_1\|^2 + \cdots + \|A\mathbf{v}_k\|^2\right)$$

Therefore choosing a k-dimensional space $\mathcal{V}$ minimizing the sum of squared distances to $\mathcal{V}$ is equivalent to choosing k orthonormal vectors $\mathbf{v}_1, \ldots, \mathbf{v}_k$ to maximize $\|A\mathbf{v}_1\|^2 + \cdots + \|A\mathbf{v}_k\|^2$.

How to choose such vectors? A greedy algorithm.

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 dim-k 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.

Algorithm: In i^{th} iteration, select unit vector $\mathbf{v}$ that maximizes $\|A\mathbf{v}\|$ among those vectors orthogonal to all previously selected vectors

- $\mathbf{v}_1$ = norm-one vector $\mathbf{v}$ maximizing $\|A\mathbf{v}\|$,
- $\mathbf{v}_2$ = norm-one vector $\mathbf{v}$ orthog. to $\mathbf{v}_1$ that maximizes $\|A\mathbf{v}\|$,
- $\mathbf{v}_3$ = 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}}
  until Av = 0 for every vector v orthogonal
            to v_1, ..., v_i     Define σ_i = ||Av_i||.
  return [v_1, v_2, ..., v_r]   r = number of iterations.
```

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 dim-k vector space $\mathcal{V}_k$ that minimizes $\sum_i(\text{distance from } \mathbf{a}_i \text{ to } \mathcal{V}_k)^2$

Algorithm: In i^{th} iteration, select vector $\mathbf{v}$ that maximizes $\|A\mathbf{v}\|$ among those vectors orthogonal to all previously selected vectors. $\Rightarrow \mathbf{v}_1, \ldots, \mathbf{v}_k$

Theorem: For each $k \geq 0$, the first k right singular vectors span the k-dimensional space $\mathcal{V}_k$ that minimizes $\sum_i(\text{distance from } \mathbf{a}_i \text{ to } V_k)^2$.

Proof: by induction on k. The case $k = 0$ is trivial.

Assume the theorem holds for $k = q - 1$. We prove it for $k = q$.

Suppose $\mathcal{W}$ is a q-dimensional space. Let $\mathbf{w}_q$ be a unit vector in $\mathcal{W}$ that is orthogonal to $\mathbf{v}_1, \ldots, \mathbf{v}_{q-1}$. (Why is there such a vector?) Let $\mathbf{w}_1, \ldots, \mathbf{w}_{q-1}$ be vectors such that $\mathbf{w}_1, \ldots, \mathbf{w}_q$ form an orthonormal basis for $\mathcal{W}$. (Why are there such vectors?)

By choice of $\mathbf{v}_q$, $\|A\mathbf{v}_q\| \geq \|A\mathbf{w}_q\|$.

By the induction hypothesis, Span $\{\mathbf{v}_1, \ldots, \mathbf{v}_{q-1}\}$ is the $(q-1)$-dimensional space minimizing sum of squared distances, so $\|A\mathbf{v}_1\|^2 + \cdots + \|A\mathbf{v}_{q-1}\|^2 \geq \|A\mathbf{w}_1\|^2 + \cdots + \|A\mathbf{w}_{q-1}\|^2$.

```
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}}
  until Av = 0 for every vector v orthogonal
           to v_1, ..., v_i
  return [v_1, v_2, ..., v_r]
```

Define $\sigma_i = \|A\mathbf{v}_i\|$.

r = number of iterations.

Proposition: The singular values $\sigma_1, \ldots, \sigma_r$ are positive and in nonincreasing order.

Proof: $\sigma_i = \|A\mathbf{v}_i\|$ and norm of a vector is nonnegative. Algorithm stops before it would choose a vector $\mathbf{v}_i$ such that $\|A\mathbf{v}_i\|$ is zero, so 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:

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

We rewrite equation as

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

Deriving the Singular Value Decomposition

We rewrite equation as

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

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

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

Deriving the Singular Value Decomposition

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

$$\begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix} = \left[\begin{array}{c|c|c} & & \\ \mathbf{u}_1 & \cdots & \mathbf{u}_n \\ & & \end{array}\right] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} \left[\begin{array}{c} \mathbf{v}_1^T \\ \hline \vdots \\ \hline \mathbf{v}_n^T \end{array}\right]$$

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

where U and V are column-orthogonal and Σ is diagonal with positive diagonal elements. called the (compact) *singular value decomposition* (SVD) of A.

Existence of SVD

Lemma: Each row of A lies in the span of the right singular vectors. **Proof** ....

Let $\mathcal{V} = \text{Span } \{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$.

By termination condition, $A\mathbf{v} = \mathbf{0}$ for every vector $\mathbf{v}$ orthogonal to $\mathcal{V}$.

For each row $\mathbf{a}_i$, write $\mathbf{a}_i = \mathbf{a}_i^{||\mathcal{V}} + \mathbf{a}_i^{\perp\mathcal{V}}$.

```
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]
```

$$
\begin{aligned}
0 &= \left\langle \mathbf{a}_i, \mathbf{a}_i^{\perp\mathcal{V}} \right\rangle = \left\langle \mathbf{a}_i^{||\mathcal{V}} + \mathbf{a}_i^{\perp\mathcal{V}}, \mathbf{a}_i^{\perp\mathcal{V}} \right\rangle \\
&= \left\langle \mathbf{a}_i^{||\mathcal{V}}, \mathbf{a}_i^{\perp\mathcal{V}} \right\rangle + \left\langle \mathbf{a}_i^{\perp\mathcal{V}}, \mathbf{a}_i^{\perp\mathcal{V}} \right\rangle \\
&= 0 + \|\mathbf{a}_i^{\perp\mathcal{V}}\|^2
\end{aligned}
$$

so $\mathbf{a}_i^{\perp\mathcal{V}} = \mathbf{0}$. Shows $\mathbf{a}_i = \mathbf{a}_i^{||\mathcal{V}}$, which shows that $\mathbf{a}_i$ lies in $\mathcal{V}$. QED

Existence, continued

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

The j^{th} column of the first matrix on the right-hand side is

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

which is $A\mathbf{v}_j$, which is $\sigma_j \mathbf{u}_j$

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

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.

Different flavors of SVD $A = U\Sigma V^T$ of an $m \times n$ matrix:

- *traditional*: U is $m \times m$, V^T is $n \times n$, and Σ is $m \times n$
- *reduced*, or *thin*:
  - If $m \geq n$ then U is $m \times n$, V^T is $n \times n$.
  - If $m \leq n$ then U is $m \times m$, V^T is $m \times n$.
- *compact:* thin, but omit zero singular values.

We never use the traditional SVD; we mostly use compact SVD.

Properties of the SVD

- Row space of A = row space of V^T
- Col space of A = col space of U.

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:

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

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.

Computing SVD

- I derived the SVD assuming a procedure to solve this problem:

$$\arg\max\{||A\mathbf{v}|| \ : \ \|\mathbf{v}\| = 1, \mathbf{v} \text{ is orthog. to } \mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_{i-1}\}$$

- Later we give a procedure to *approximately* solve this problem.
- The most efficient method for computing the SVD is beyond the scope of the course.

Example: Senators

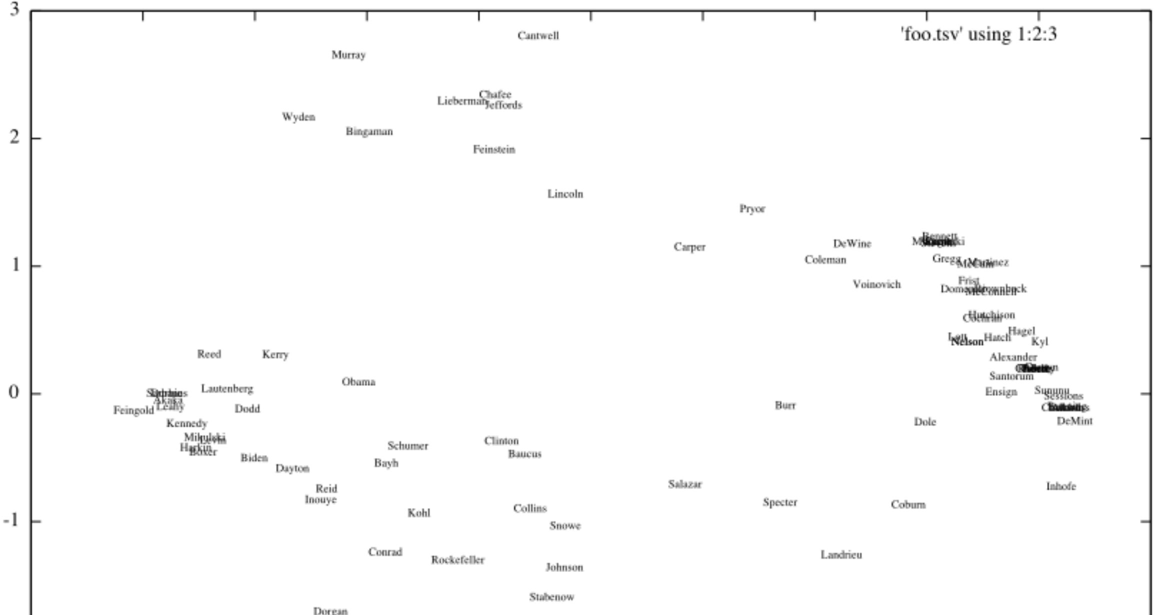
First center the data. Then find first two right singular vectors $\mathbf{v}_1$ and $\mathbf{v}_2$.
Projecting onto these gives two coordinates.
To find singular vectors,

- 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