KUTAISI INTERNATIONAL UNIVERSITY

Numerical Linear Algebra

Ramaz Botchorishvili

Kutaisi International University

December 21, 2022

Numerical Linear Algebra

Ramaz Botchorishvili

Kutaisi International University

December 21, 2022

Numerical egenvalue problem

- Numerical eigenvalue problem
- Gershgorin theorems
- SVD
- Orthogonal projections
- Orthogonal directions
- Difficulties with theoretical linear algebra
- Q & A

Recap of Previous Lecture

- Invertibility of KKT matrix
- QR factorization for constrained least squares problem
- Hausholder transformations and QR factorization

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, y \neq 0, \lambda \in \mathbb{C}$, $Ay = \lambda y$

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, y \neq 0, \lambda \in \mathbb{C},\ Ay = \lambda y$
- Characteristic polynomial $det(A - \lambda I)$

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, y \neq 0, \lambda \in \mathbb{C},\ Ay = \lambda y$
- Characteristic polynomial $det(A - \lambda I)$
- Characteristic equation $det(A - \lambda I) = 0$

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, y \neq 0, \lambda \in \mathbb{C},\ Ay = \lambda y$
- Characteristic polynomial $det(A - \lambda I)$
- Characteristic equation $det(A - \lambda I) = 0$
- Left eigenvector $y^* A = \lambda y^*, y \neq 0$

Theorem 14.2

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, y \neq 0, \lambda \in \mathbb{C}$, $Ay = \lambda y$
- Characteristic polynomial $det(A - \lambda I)$
- Characteristic equation $det(A - \lambda I) = 0$
- Left eigenvector $y^* A = \lambda y^*, y \neq 0$

Theorem 14.2

- *Rayleigh quotient:*

$$r(x) = \frac{(Ax, x)}{(x, x)}, x \neq 0$$

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, y \neq 0, \lambda \in \mathbb{C},\ Ay = \lambda y$
- Characteristic polynomial $det(A - \lambda I)$
- Characteristic equation $det(A - \lambda I) = 0$
- Left eigenvector $y^* A = \lambda y^*, y \neq 0$

Theorem 14.2

- *Rayleigh quotient:*

$$r(x) = \frac{(Ax, x)}{(x, x)}, x \neq 0$$

- $A = A^* \Rightarrow r(x) \in \mathbb{R}$

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, y \neq 0, \lambda \in \mathbb{C},\ Ay = \lambda y$
- Characteristic polynomial $det(A - \lambda I)$
- Characteristic equation $det(A - \lambda I) = 0$
- Left eigenvector $y^* A = \lambda y^*, y \neq 0$

Theorem 14.2

- *Rayleigh quotient:*

$$r(x) = \frac{(Ax, x)}{(x, x)}, x \neq 0$$

- $A = A^* \Rightarrow r(x) \in \mathbb{R}$
- *x eigenvector with eigenvalue $\lambda \Rightarrow r(x) = \lambda$*

Eigenvalues and eigenvectors, 1

Definition 14.1

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, y \neq 0, \lambda \in \mathbb{C},\ Ay = \lambda y$
- Characteristic polynomial $det(A - \lambda I)$
- Characteristic equation $det(A - \lambda I) = 0$
- Left eigenvector $y^* A = \lambda y^*, y \neq 0$

Theorem 14.2

- *Rayleigh quotient:*
$$r(x) = \frac{(Ax, x)}{(x, x)}, x \neq 0$$
- $A = A^* \Rightarrow r(x) \in \mathbb{R}$
- *x eigenvector with eigenvalue $\lambda \Rightarrow r(x) = \lambda$*
- *$r(x)x$ is the orthogonal projection of Ax onto the line spanned by x:*
$$\|Ax - r(x)x\|_2 = \min_{\mu \in \mathbb{C}} \|Ax - \mu x\|_2$$

Eigenvalues and eigenvectors, 2

Theorem 14.3

Eigenvalues and eigenvectors, 2

Theorem 14.3

- $A \in \mathbb{C}^{n \times n}$

Eigenvalues and eigenvectors, 2

Theorem 14.3

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, \|y\|_2 = 1, y^* A y = r$

Eigenvalues and eigenvectors, 2

Theorem 14.3

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, \|y\|_2 = 1, y^* A y = r$
- $Ax = \lambda x, \|x\|_2 = 1$

Eigenvalues and eigenvectors, 2

Theorem 14.3

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, \|y\|_2 = 1, y^* A y = r$
- $Ax = \lambda x, \|x\|_2 = 1$
- $\Downarrow$
- $|\lambda - r| \leq 2\|A\|_2 \|x - y\|_2$

Eigenvalues and eigenvectors, 2

Theorem 14.3

- $A \in \mathbb{C}^{n \times n}$
- $y \in \mathbb{C}^n, \|y\|_2 = 1, y^*Ay = r$
- $Ax = \lambda x, \|x\|_2 = 1$
- $\Downarrow$
- $|\lambda - r| \leq 2\|A\|_2\|x - y\|_2$

Proof.

$$\lambda - r = x^*Ax - y^*Ay$$

$$x^*Ay = y^*Ax$$

$$x^*Ax - y^*Ay = x^*Ax - x^*Ay + y^*Ax - y^*Ay = x^*A(x - y) + y^*A(x - y)$$

$$\begin{aligned} |\lambda - r| = |x^*A(x - y) + y^*A(x - y)| &\leq |x^*A(x - y)| + |y^*A(x - y)| \\ &\leq \|x^*\|_2\|A\|_2\|x - y\|_2 + \|y^*\|_2\|A\|_2\|x - y\|_2 \\ &\leq 2\|A\|_2\|x - y\|_2 \end{aligned}$$

Eigenvalues and eigenvectors, 3

Theorem 14.4

Left and right eigenvectors corresponding to different eigenvalues are orthogonal one to another

Proof.

Eigenvalues and eigenvectors, 3

Theorem 14.4

Left and right eigenvectors corresponding to different eigenvalues are orthogonal one to another

Proof.

- $Ax = \lambda x, y^* A = \mu y^*$

Eigenvalues and eigenvectors, 3

Theorem 14.4

Left and right eigenvectors corresponding to different eigenvalues are orthogonal one to another

Proof.

- $Ax = \lambda x, y^* A = \mu y^*$
- $(Ax, y) = \lambda(x, y)$

Eigenvalues and eigenvectors, 3

Theorem 14.4

Left and right eigenvectors corresponding to different eigenvalues are orthogonal one to another

Proof.

- $Ax = \lambda x, y^* A = \mu y^*$
- $(Ax, y) = \lambda (x, y)$
- $y^* A = \mu y^* \equiv A^* y = \mu y$

Eigenvalues and eigenvectors, 3

Theorem 14.4

Left and right eigenvectors corresponding to different eigenvalues are orthogonal one to another

Proof.

- $Ax = \lambda x, y^* A = \mu y^*$
- $(Ax, y) = \lambda(x, y)$
- $y^* A = \mu y^* \equiv A^* y = \mu y$
- $\lambda(x, y) = (Ax, y) = (x, A^* y) = \mu(x, y)$

Eigenvalues and eigenvectors, 3

Theorem 14.4

Left and right eigenvectors corresponding to different eigenvalues are orthogonal one to another

Proof.

- $Ax = \lambda x, y^*A = \mu y^*$
- $(Ax, y) = \lambda(x, y)$
- $y^*A = \mu y^* \equiv A^*y = \mu y$
- $\lambda(x, y) = (Ax, y) = (x, A^*y) = \mu(x, y)$
- $(\lambda - \mu)(x, y) = 0$

Eigenvalues and eigenvectors, 3

Theorem 14.4

Left and right eigenvectors corresponding to different eigenvalues are orthogonal one to another

Proof.

- $Ax = \lambda x, y^* A = \mu y^*$
- $(Ax, y) = \lambda(x, y)$
- $y^* A = \mu y^* \equiv A^* y = \mu y$
- $\lambda(x, y) = (Ax, y) = (x, A^* y) = \mu(x, y)$
- $(\lambda - \mu)(x, y) = 0$
- $\lambda \neq \mu \Rightarrow (x, y) = 0 \equiv x \perp y$

$\square$

Eigenvalues and eigenvectors, 4

Theorem 14.5

Two similar matrices have same eigenvalues

Proof.

Eigenvalues and eigenvectors, 4

Theorem 14.5

Two similar matrices have same eigenvalues

Proof.

$A = P^{-1}BP, Ay = \lambda y, BPy = \lambda Py$ $\square$

Definition 14.6

Eigenvalues and eigenvectors, 4

Theorem 14.5

Two similar matrices have same eigenvalues

Proof.

$A = P^{-1}BP, Ay = \lambda y, BPy = \lambda Py$ $\square$

Definition 14.6

Square $n \times n$ matrix A is diagonalizable if

Eigenvalues and eigenvectors, 4

Theorem 14.5

Two similar matrices have same eigenvalues

Proof.

$A = P^{-1}BP, Ay = \lambda y, BPy = \lambda Py$ □

Definition 14.6

Square $n \times n$ matrix A is diagonalizable if

- $\exists P, P^{-1}$

Eigenvalues and eigenvectors, 4

Theorem 14.5

Two similar matrices have same eigenvalues

Proof.

$A = P^{-1}BP, Ay = \lambda y, BPy = \lambda Py$ $\square$

Definition 14.6

Square $n \times n$ matrix A is diagonalizable if

- $\exists P, P^{-1}$
- $\exists D = diag\{d_1, d2, ..., d_n\}$

Eigenvalues and eigenvectors, 4

Theorem 14.5

Two similar matrices have same eigenvalues

Proof.

$A = P^{-1}BP, Ay = \lambda y, BPy = \lambda Py$ □

Definition 14.6

Square $n \times n$ matrix A is diagonalizable if

- $\exists P, P^{-1}$
- $\exists D = diag\{d_1, d2, ..., d_n\}$
- $A = PDP^{-1}$

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square* $n \times n$ *matrix* A

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square* $n \times n$ *matrix* A
- n *linearly independent right eigenvectors* $x_i, i = 1, 2, ..., n$

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square $n \times n$ matrix A*
- *n linearly independent right eigenvectors $x_i, i = 1, 2, ..., n$*
- $\Rightarrow$

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square* $n \times n$ *matrix* A
- n *linearly independent right eigenvectors* $x_i, i = 1, 2, ..., n$
- $\Rightarrow$
- $A = X \Lambda X^{-1}, \Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}, X = (x_1, x_2, ..., x_n)$

Theorem 14.8

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square* $n \times n$ *matrix* A
- n *linearly independent right eigenvectors* $x_i, i = 1, 2, ..., n$
- $\Rightarrow$
- $A = X \Lambda X^{-1}, \Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}, X = (x_1, x_2, ..., x_n)$

Theorem 14.8

- *Square* $n \times n$ *matrix* $A = X \Lambda X^{-1}$

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square* $n \times n$ *matrix* A
- n *linearly independent right eigenvectors* $x_i, i = 1, 2, ..., n$
- $\Rightarrow$
- $A = X \Lambda X^{-1}, \Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}, X = (x_1, x_2, ..., x_n)$

Theorem 14.8

- *Square* $n \times n$ *matrix* $A = X \Lambda X^{-1}$
- $\Downarrow$

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square* $n \times n$ *matrix* A
- n *linearly independent right eigenvectors* $x_i, i = 1, 2, ..., n$
- $\Rightarrow$
- $A = X\Lambda X^{-1}, \Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}, X = (x_1, x_2, ..., x_n)$

Theorem 14.8

- *Square* $n \times n$ *matrix* $A = X\Lambda X^{-1}$
- $\Downarrow$
- $\Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}$

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square* $n \times n$ *matrix* A
- n *linearly independent right eigenvectors* $x_i, i = 1, 2, ..., n$
- $\Rightarrow$
- $A = X\Lambda X^{-1}, \Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}, X = (x_1, x_2, ..., x_n)$

Theorem 14.8

- *Square* $n \times n$ *matrix* $A = X\Lambda X^{-1}$
- $\Downarrow$
- $\Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}$
- $X = (x_1, x_2, ..., x_n)$, $x_i, i = 1, 2, ..., n$ - *right eigenvectors*

Eigenvalues and eigenvectors, 5

Theorem 14.7

Eigenvalue decomposition

- *Square $n \times n$ matrix A*
- *n linearly independent right eigenvectors $x_i, i = 1, 2, ..., n$*
- $\Rightarrow$
- $A = X\Lambda X^{-1}, \Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}, X = (x_1, x_2, ..., x_n)$

Theorem 14.8

- *Square $n \times n$ matrix $A = X\Lambda X^{-1}$*
- $\Downarrow$
- $\Lambda = diag\{\lambda_1, \lambda_2, ..., \lambda_n\}$
- $X = (x_1, x_2, ..., x_n)$, $x_i, i = 1, 2, ..., n$ *- right eigenvectors*
- $X^{-1} = \begin{pmatrix} y_1^T \\ y_1^T \\ .. \\ y_n^T \end{pmatrix}, y_i^T, i = 1, 2, ..., n -$ *left eigenvectors*

Gershgorin disk, 1

Definition 14.9

Gershgorin disk, 1

Definition 14.9

- $D = diag\{d_1, d_2, ..., d_n\}$

Gershgorin disk, 1

Definition 14.9

- $D = diag\{d_1, d_2, ..., d_n\}$
- $A = D + E, E = (e_{ij})_{n \times n}$

Gershgorin disk, 1

Definition 14.9

- $D = diag\{d_1, d_2, ..., d_n\}$
- $A = D + E, E = (e_{ij})_{n \times n}$
- Gershgorin disk: $D_i = \{z \in \mathbb{C}, |z - d_i| \leq \sum_{j=1}^{n} |e_{ij}|\}, i = 1, 2, ..., n$

Theorem 14.10

Gershgorin disk, 1

Definition 14.9

- $D = diag\{d_1, d_2, ..., d_n\}$
- $A = D + E, E = (e_{ij})_{n\times n}$
- Gershgorin disk: $D_i = \{z \in \mathbb{C}, |z - d_i| \leq \sum_{j=1}^{n} |e_{ij}|\}, i = 1, 2, ..., n$

Theorem 14.10

Every eigenvalue of $A = D + E$ lies in one of the Gershgorin disks

Proof.

Gershgorin disk, 1

Definition 14.9

- $D = diag\{d_1, d_2, ..., d_n\}$
- $A = D + E, E = (e_{ij})_{n\times n}$
- Gershgorin disk: $D_i = \{z \in \mathbb{C}, |z - d_i| \leq \sum_{j=1}^{n} |e_{ij}|\}, i = 1, 2, ..., n$

Theorem 14.10

Every eigenvalue of $A = D + E$ lies in one of the Gershgorin disks

Proof.

- $Ax = \lambda x$

Gershgorin disk, 1

Definition 14.9

- $D = diag\{d_1, d_2, ..., d_n\}$
- $A = D + E, E = (e_{ij})_{n \times n}$
- Gershgorin disk: $D_i = \{z \in \mathbb{C}, |z - d_i| \leq \sum_{j=1}^{n} |e_{ij}|\}, i = 1, 2, ..., n$

Theorem 14.10

Every eigenvalue of $A = D + E$ lies in one of the Gershgorin disks

Proof.

- $Ax = \lambda x$
- Let $\|x\|_\infty = |x_k| \quad \equiv |x_k| = \max\{|x_1|, |x_2|, ..., |x_n|\} \neq 0$

Gershgorin disk, 1

Definition 14.9

- $D = diag\{d_1, d_2, ..., d_n\}$
- $A = D + E, E = (e_{ij})_{n \times n}$
- Gershgorin disk: $D_i = \{z \in \mathbb{C}, |z - d_i| \leq \sum_{j=1}^{n} |e_{ij}|\}, i = 1, 2, ..., n$

Theorem 14.10

Every eigenvalue of $A = D + E$ lies in one of the Gershgorin disks

Proof.

- $Ax = \lambda x$
- Let $\|x\|_\infty = |x_k| \quad \equiv |x_k| = \max\{|x_1|, |x_2|, ..., |x_n|\} \neq 0$
- $Ax = \begin{pmatrix} d_1 x_1 + \sum_{j=1}^{n} e_{1j} x_j \\ .. \\ d_n x_n + \sum_{j=1}^{n} e_{nj} x_j \end{pmatrix}, \lambda x = \begin{pmatrix} \lambda x_1 \\ .. \\ \lambda x_n \end{pmatrix} \Rightarrow d_k x_k + \sum_{j=1}^{n} e_{kj} x_j = \lambda x_k$

Gershgorin disk, 1

Definition 14.9

- $D = diag\{d_1, d_2, ..., d_n\}$
- $A = D + E, E = (e_{ij})_{n\times n}$
- Gershgorin disk: $D_i = \{z \in \mathbb{C}, |z - d_i| \leq \sum_{j=1}^{n} |e_{ij}|\}, i = 1, 2, ..., n$

Theorem 14.10

Every eigenvalue of $A = D + E$ lies in one of the Gershgorin disks

Proof.

- $Ax = \lambda x$
- Let $\|x\|_\infty = |x_k| \quad \equiv |x_k| = \max\{|x_1|, |x_2|, ..., |x_n|\} \neq 0$
- $Ax = \begin{pmatrix} d_1x_1 + \sum_{j=1}^{n} e_{1j}x_j \\ .. \\ d_nx_n + \sum_{j=1}^{n} e_{nj}x_j \end{pmatrix}, \lambda x = \begin{pmatrix} \lambda x_1 \\ .. \\ \lambda x_n \end{pmatrix} \Rightarrow d_kx_k + \sum_{j=1}^{n} e_{kj}x_j = \lambda x_k$
- $|d_k - \lambda| = |\sum_{j=1}^{n} e_{kj}x_j/x_k| \leq \sum_{j=1}^{n} |e_{kj}| \cdot |x_j/x_k| \leq \sum_{j=1}^{n} |e_{kj}|$

$\square$

Gershgorin disk, 2

Theorem 14.11

Gershgorin's second theorem

Gershgorin disk, 2

Theorem 14.11

Gershgorin's second theorem

- $D_i i = 1, 2, ..., n$ - *Gershgorin disks*

Gershgorin disk, 2

Theorem 14.11

Gershgorin's second theorem

- $D_i i = 1, 2, ..., n$ - *Gershgorin disks*
- $\cup_{i \in \{i_1, i_2, ..., i_k\}} D_i \ \cap \ \cup_{i \in \{i_{k+1}, i_{k+2}, ..., i_n\}} D_i = \emptyset$

Gershgorin disk, 2

Theorem 14.11

Gershgorin's second theorem

- $D_i i = 1, 2, ..., n$ - *Gershgorin disks*
- $\cup_{i\in\{i_1,i_2,...,i_k\}} D_i \ \cap \ \cup_{i\in\{i_{k+1},i_{k+2},...,i_n\}} D_i = \emptyset$
- $\Rightarrow$

Gershgorin disk, 2

Theorem 14.11

Gershgorin's second theorem

- $D_i i = 1, 2, ..., n$ - *Gershgorin disks*
- $\cup_{i \in \{i_1, i_2, ..., i_k\}} D_i \ \cap \ \cup_{i \in \{i_{k+1}, i_{k+2}, ..., i_n\}} D_i = \emptyset$
- $\Rightarrow$
- $\lambda_j \in \cup_{i \in \{i_1, i_2, ..., i_k\}} D_i, j \in \{i_1, i_2, ..., i_k\}$

Gershgorin disk, 2

Theorem 14.11

Gershgorin's second theorem

- $D_i i = 1, 2, ..., n$ - *Gershgorin disks*
- $\cup_{i\in\{i_1,i_2,...,i_k\}} D_i \ \cap \ \cup_{i\in\{i_{k+1},i_{k+2},...,i_n\}} D_i = \emptyset$
- $\Rightarrow$
- $\lambda_j \in \cup_{i\in\{i_1,i_2,...,i_k\}} D_i, j \in \{i_1, i_2, ..., i_k\}$
- $\lambda_l \in \cup_{i\in\{i_{k+1},i_{k+2},...,i_n\}} D_i, l \in \{i_{k+1}, i_{k+2}, ..., i_n\}$

Power method, 1

- Power method

Power method, 1

- Power method
- Input: $A = (a_{ij})_{n \times n}$

Power method, 1

- Power method
- Input: $A = (a_{ij})_{n \times n}$
- Output: approximate dominant eigenvalue and corresponding eigenvector

Power method, 1

- Power method
- Input: $A = (a_{ij})_{n \times n}$
- Output: approximate dominant eigenvalue and corresponding eigenvector
- Algorithm

Power method, 1

- Power method
- Input: $A = (a_{ij})_{n \times n}$
- Output: approximate dominant eigenvalue and corresponding eigenvector
- Algorithm
    1. Choose initial approximation x_0

Power method, 1

- Power method
- Input: $A = (a_{ij})_{n\times n}$
- Output: approximate dominant eigenvalue and corresponding eigenvector
- Algorithm
  1. Choose initial approximation x_0
  2. Do until convergence for $k = 1, 2, ..:$

$$\tilde{x}_k = Ax_{k-1}$$
$$x_k = \tilde{x}_k / \|\tilde{x}_k\|_\infty$$

Theorem 14.12

Power method, 1

- Power method
- Input: $A = (a_{ij})_{n\times n}$
- Output: approximate dominant eigenvalue and corresponding eigenvector
- Algorithm
  1. Choose initial approximation x_0
  2. Do until convergence for $k = 1, 2, ..$:

$$\tilde{x}_k = Ax_{k-1}$$
$$x_k = \tilde{x}_k / \|\tilde{x}_k\|_\infty$$

Theorem 14.12

- $A = (a_{ij})_{n\times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq ... \geq |\lambda_n|$

Power method, 1

- Power method
- Input: $A = (a_{ij})_{n \times n}$
- Output: approximate dominant eigenvalue and corresponding eigenvector
- Algorithm
  1. Choose initial approximation x_0
  2. Do until convergence for $k = 1, 2, ..$:

$$\tilde{x}_k = A x_{k-1}$$
$$x_k = \tilde{x}_k / \|\tilde{x}_k\|_\infty$$

Theorem 14.12

- $A = (a_{ij})_{n \times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq ... \geq |\lambda_n|$
- $\Downarrow$

Power method, 1

- Power method
- Input: $A = (a_{ij})_{n\times n}$
- Output: approximate dominant eigenvalue and corresponding eigenvector
- Algorithm
  1. Choose initial approximation x_0
  2. Do until convergence for $k = 1, 2, ..$:

$$\tilde{x}_k = Ax_{k-1}$$
$$x_k = \tilde{x}_k / \|\tilde{x}_k\|_\infty$$

Theorem 14.12

- $A = (a_{ij})_{n\times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq ... \geq |\lambda_n|$
- $\Downarrow$
- *Power method converges:*

$$\lim_{k\to\infty} \|\tilde{x}_k\| = \lambda_1,\ \lim_{k\to\infty} x_k = \alpha y_1,\ Ay_1 = \lambda_1 y_1, \alpha \in \mathbb{R}$$

Power method, 2

Theorem 14.13

Power method, 2

Theorem 14.13

- $A = (a_{ij})_{n\times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq ... \geq |\lambda_n|$

Power method, 2

Theorem 14.13

- $A = (a_{ij})_{n \times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_n|$
- $\Downarrow$

Power method, 2

Theorem 14.13

- $A = (a_{ij})_{n\times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq ... \geq |\lambda_n|$
- $\Downarrow$
- *Power method converges:*

$$\lim_{k\to\infty} \|\tilde{x}_k\| = \lambda_1, \ \lim_{k\to\infty} x_k = \alpha y_1, \ Ay_1 = \lambda_1 y_1, \alpha \in \mathbb{R}$$

Proof.

Power method, 2

Theorem 14.13

- $A = (a_{ij})_{n\times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq ... \geq |\lambda_n|$
- $\Downarrow$
- *Power method converges:*

$$\lim_{k\to\infty} \|\tilde{x}_k\| = \lambda_1, \ \lim_{k\to\infty} x_k = \alpha y_1, \ Ay_1 = \lambda_1 y_1, \alpha \in \mathbb{R}$$

Proof.

$$\tilde{x}_k = Ax_{k-1}, x_k = \tilde{x}_k/\|\tilde{x}_k\| \Rightarrow x_k = Ax_{k-1}/\|Ax_{k-1}\|$$

$$x_1 = Ax_0/\|Ax_0\|$$

$$x_{k+1} = A^{k+1}x_0/\|A^{k+1}x_0\|?$$

Power method, 2

Theorem 14.13

- $A = (a_{ij})_{n\times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq ... \geq |\lambda_n|$
- $\Downarrow$
- *Power method converges:*

$$\lim_{k\to\infty} \|\tilde{x}_k\| = \lambda_1, \ \lim_{k\to\infty} x_k = \alpha y_1, \ Ay_1 = \lambda_1 y_1, \alpha \in \mathbb{R}$$

Proof.

$$\tilde{x}_k = Ax_{k-1}, x_k = \tilde{x}_k/\|\tilde{x}_k\| \Rightarrow x_k = Ax_{k-1}/\|Ax_{k-1}\|$$

$$x_1 = Ax_0/\|Ax_0\|$$

$$x_{k+1} = A^{k+1}x_0/\|A^{k+1}x_0\|?$$

Proof by math induction: valid for k=0, assume for k: $x_k = A^k x_0/\|A^k x_0\|$

Power method, 2

Theorem 14.13

- $A = (a_{ij})_{n\times n}$, $\lambda_1 > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n|$
- $\Downarrow$
- *Power method converges:*

$$\lim_{k\to\infty} \|\tilde{x}_k\| = \lambda_1, \ \lim_{k\to\infty} x_k = \alpha y_1, \ Ay_1 = \lambda_1 y_1, \alpha \in \mathbb{R}$$

Proof.

$$\tilde{x}_k = Ax_{k-1}, x_k = \tilde{x}_k/\|\tilde{x}_k\| \Rightarrow x_k = Ax_{k-1}/\|Ax_{k-1}\|$$

$$x_1 = Ax_0/\|Ax_0\|$$

$$x_{k+1} = A^{k+1}x_0/\|A^{k+1}x_0\|?$$

Proof by math induction: valid for k=0, assume for k: $x_k = A^k x_0/\|A^k x_0\|$

$$x_{k+1} = Ax_k/\|Ax_k\| \Rightarrow x_{k+1} = \frac{A(A^k x_0/\|A^k x_0\|)}{\|A(A^k x_0/\|A^k x_0\|)\|} = A^{k+1}x_0/\|A^{k+1}x_0\| \quad \square$$

Power method, 3

Proof.

Proof by math induction: valid for k=0, assume for k: $x_k = A^k x_0 / \|A^k x_0\|$

$x_{k+1} = Ax_k / \|Ax_k\| \Rightarrow x_{k+1} = \frac{A(A^k x_0/\|A^k x_0\|)}{\|A(A^k x_0/\|A^k x_0\|)\|} = A^{k+1} x_0 / \|A^{k+1} x_0\|$

Power method, 3

Proof.

Proof by math induction: valid for k=0, assume for k: $x_k = A^k x_0 / \|A^k x_0\|$

$x_{k+1} = Ax_k / \|Ax_k\| \Rightarrow x_{k+1} = \frac{A(A^k x_0/\|A^k x_0\|)}{\|A(A^k x_0/\|A^k x_0\|)\|} = A^{k+1} x_0 / \|A^{k+1} x_0\|$

- Assume eigenvectors $y_i, i = 1, 2, ..., n$ are lenearly independent

Power method, 3

Proof.

Proof by math induction: valid for k=0, assume for k: $x_k = A^k x_0/\|A^k x_0\|$

$x_{k+1} = Ax_k/\|Ax_k\| \Rightarrow x_{k+1} = \frac{A(A^k x_0/\|A^k x_0\|)}{\|A(A^k x_0/\|A^k x_0\|)\|} = A^{k+1}x_0/\|A^{k+1}x_0\|$

- Assume eigenvectors $y_i, i = 1, 2, ..., n$ are lenearly independent
-

$$\alpha_1 \neq 0, x_0 = \sum_{i=1}^{n} \alpha_i y_i \Rightarrow A^k x_0 = \sum_{i=1}^{n} \alpha_i A^k y_i = \sum_{i=1}^{n} \alpha_i \lambda_i^k y_i$$

$$A^k x_0 = \lambda_1^k(\alpha_1 y_1 + \sum_{i=2}^{n} \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i), \; \lim_{k\to\infty} (\frac{\lambda_i}{\lambda_1})^k = 0$$

$$\Rightarrow \lim_{k\to\infty} x_k = \alpha y_1, \alpha \in \mathbb{R}$$

$$Ax_k = AA^k x_0/\|A^k x_0\| = \frac{\lambda_1^{k+1}(\alpha_1 y_1 + \sum_{i=2}^{n} \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i)}{\|\lambda_1^k(\alpha_1 y_1 + \sum_{i=2}^{n} \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i)\|}$$

$$\Rightarrow \lim_{k\to\infty} \|\tilde{x}_{k+1}\| = \lim_{k\to\infty} \|Ax_k\| = \lambda_1$$

Power method, 4

- $$x_k = A^k x_0 / \|A^k x_0\| = \frac{\lambda_1^k(\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i)}{\|\lambda_1^k(\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i)\|}$$

$$\|x_k - \alpha_1 y_1\| \leq \frac{\sum_{i=2}^n |\alpha_i| (\frac{\lambda_i}{\lambda_1})^k \|y_i\|)}{\|\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i\|}$$

$$\Rightarrow$$

$$\|x_k - \alpha_1 y_1\| \leq c \cdot (\frac{\lambda_2}{\lambda_1})^k, c \in \mathbb{R}$$

Power method, 4

- $$x_k = A^k x_0 / \|A^k x_0\| = \frac{\lambda_1^k(\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i)}{\|\lambda_1^k(\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i)\|}$$
$$\|x_k - \alpha_1 y_1\| \leq \frac{\sum_{i=2}^n |\alpha_i| (\frac{\lambda_i}{\lambda_1})^k \|y_i\|)}{\|\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i\|}$$
$$\Rightarrow$$
$$\|x_k - \alpha_1 y_1\| \leq c \cdot (\frac{\lambda_2}{\lambda_1})^k, c \in \mathbb{R}$$
- Power method converges if $|\lambda_1| > |\lambda_2|$

Power method, 4

- $$x_k = A^k x_0 / \|A^k x_0\| = \frac{\lambda_1^k(\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i)}{\|\lambda_1^k(\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i)\|}$$
$$\|x_k - \alpha_1 y_1\| \leq \frac{\sum_{i=2}^n |\alpha_i| (\frac{\lambda_i}{\lambda_1})^k \|y_i\|)}{\|\alpha_1 y_1 + \sum_{i=2}^n \alpha_i (\frac{\lambda_i}{\lambda_1})^k y_i\|}$$
$$\Rightarrow$$
$$\|x_k - \alpha_1 y_1\| \leq c \cdot (\frac{\lambda_2}{\lambda_1})^k, c \in \mathbb{R}$$
- Power method converges if $|\lambda_1| > |\lambda_2|$
- Power method also converges if more then one dominant eigenvalue exists

Power method, 5

- Power method for finding least dominant eigenvalue

Power method, 5

- Power method for finding least dominant eigenvalue
- $|\lambda_1| > |\lambda_2| \geq ... \geq |\lambda_{n-1}| > |\lambda_n|$

Power method, 5

- Power method for finding least dominant eigenvalue
- $|\lambda_1| > |\lambda_2| \geq ... \geq |\lambda_{n-1}| > |\lambda_n|$
- $|\lambda_1|^{-1} < |\lambda_2|^{-1} \leq ... \leq |\lambda_{n-1}|^{-1} < |\lambda_n|^{-1}$ - eigenvalues of A^{-1}

Power method, 5

- Power method for finding least dominant eigenvalue
- $|\lambda_1| > |\lambda_2| \geq ... \geq |\lambda_{n-1}| > |\lambda_n|$
- $|\lambda_1|^{-1} < |\lambda_2|^{-1} \leq ... \leq |\lambda_{n-1}|^{-1} < |\lambda_n|^{-1}$ - eigenvalues of A^{-1}
- Approach: apply power method for A^{-1}

Power method, 5

- Power method for finding least dominant eigenvalue
- $|\lambda_1| > |\lambda_2| \geq ... \geq |\lambda_{n-1}| > |\lambda_n|$
- $|\lambda_1|^{-1} < |\lambda_2|^{-1} \leq ... \leq |\lambda_{n-1}|^{-1} < |\lambda_n|^{-1}$ - eigenvalues of A^{-1}
- Approach: apply power method for A^{-1}
- Inverse power method converges if $|\lambda_1|^{-1} > |\lambda_2|^{-1}$

Power method, 5

- Power method for finding least dominant eigenvalue
- $|\lambda_1| > |\lambda_2| \geq ... \geq |\lambda_{n-1}| > |\lambda_n|$
- $|\lambda_1|^{-1} < |\lambda_2|^{-1} \leq ... \leq |\lambda_{n-1}|^{-1} < |\lambda_n|^{-1}$ - eigenvalues of A^{-1}
- Approach: apply power method for A^{-1}
- Inverse power method converges if $|\lambda_1|^{-1} > |\lambda_2|^{-1}$
- Reciprocal of found approximation value should be taken

Power method, 6

- Power method with shift

Power method, 6

- Power method with shift
- Consider $A - \sigma I$

Power method, 6

- Power method with shift
- Consider $A - \sigma I$
- Eigenvectors of $A - \sigma I$ and A are the same

Power method, 6

- Power method with shift
- Consider $A - \sigma I$
- Eigenvectors of $A - \sigma I$ and A are the same
- Eigenvalues of $A - \sigma I$ are $\lambda_i - \sigma, i = 1, 2, .., n$

Power method, 6

- Power method with shift
- Consider $A - \sigma I$
- Eigenvectors of $A - \sigma I$ and A are the same
- Eigenvalues of $A - \sigma I$ are $\lambda_i - \sigma, i = 1, 2, .., n$
- Is faster convergence possible ? $\sigma = ?$

Power method, 6

- Power method with shift
- Consider $A - \sigma I$
- Eigenvectors of $A - \sigma I$ and A are the same
- Eigenvalues of $A - \sigma I$ are $\lambda_i - \sigma, i = 1, 2, .., n$
- Is faster convergence possible ? $\sigma = ?$
- Approach: apply inverse iterations to $A - \sigma I$

Singular value decomposition, 1

Theorem 6.1 (SVD). *Any $m \times n$ matrix A, with $m \geq n$, can be factorized*

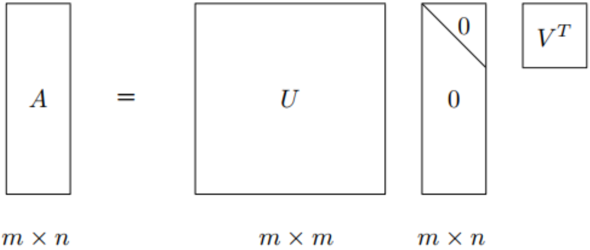
$$A = U \begin{pmatrix} \Sigma \\ 0 \end{pmatrix} V^T, \tag{6.1}$$

where $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ are orthogonal, and $\Sigma \in \mathbb{R}^{n \times n}$ is diagonal,

$$\Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n),$$

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0.$$

Figure: Singular value decomposition, L.Elden



Singular value decomposition, 2

- Low rank approximation

$$A = \sum_{i=1}^{n} \sigma_i u_i v_i^T \approx \sum_{i=1}^{k} \sigma_i u_i v_i^T =: A_k.$$

-

Theorem 6.6. *Assume that the matrix* $A \in \mathbb{R}^{m \times n}$ *has rank* $r > k$*. The matrix*

approximation problem

$$\min_{\operatorname{rank}(Z)=k} \|A - Z\|_2$$

has the solution

Singular value decomposition, 3

Theorem 6.6. *Assume that the matrix $A \in \mathbb{R}^{m \times n}$ has rank $r > k$. The matrix*

64 Chapter 6. Singular Value Decomposition

approximation problem

$$\min_{\operatorname{rank}(Z)=k} \|A - Z\|_2$$

has the solution

$$Z = A_k := U_k \Sigma_k V_k^T,$$

where $U_k = (u_1, \ldots, u_k)$, $V_k = (v_1, \ldots, v_k)$, and $\Sigma_k = \operatorname{diag}(\sigma_1, \ldots, \sigma_k)$. The minimum is

$$\|A - A_k\|_2 = \sigma_{k+1}.$$

Figure: From L Elden

A. Computational Difficulties of Theoretical Linear Algebra

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.14

Solving linear system $Ax = b$ by Cramer's rule

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.14

Solving linear system $Ax = b$ by Cramer's rule

- Cramer's rule needs determinants

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.14

Solving linear system $Ax = b$ by Cramer's rule

- Cramer's rule needs determinants
- Computing determinant of $n \times n$ matrix costs approximately $n!$ FLOPS

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.14

Solving linear system $Ax = b$ by Cramer's rule

- Cramer's rule needs determinants
- Computing determinant of $n \times n$ matrix costs approximately $n!$ FLOPS
- FLOPS = floating point operations per second

A. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.14

Solving linear system $Ax = b$ by Cramer's rule

- Cramer's rule needs determinants
- Computing determinant of $n \times n$ matrix costs approximately $n!$ FLOPS
- FLOPS = floating point operations per second
- Solving linear system $Ax = b$ with twenty unknowns will take millions of years on today's fastest computer

B. Computational Difficulties of Theoretical Linear Algebra

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
-

  Computing the unique solution of a linear system by matrix inversion

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.15

Computing the unique solution of a linear system by matrix inversion

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.15

Computing the unique solution of a linear system by matrix inversion

- $Ax = b$, $x = A^{-1}b$

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.15

Computing the unique solution of a linear system by matrix inversion

- $Ax = b$, $x = A^{-1}b$
- Algorithm:
  1. compute matrix inverse A^{-1}
  2. compute solution $x = A^{-1}b$

B. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving a least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.15

Computing the unique solution of a linear system by matrix inversion

- $Ax = b$, $x = A^{-1}b$
- Algorithm:
  1. compute matrix inverse A^{-1}
  2. compute solution $x = A^{-1}b$
- Computing matrix inverse is not practical:
  1. using standard elimination method is approximately 2.5 times faster
  2. other methods are often more accurate

C. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

C. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.16

Solving a least squares problem by normal equations

C. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.16

Solving a least squares problem by normal equations

- The least squares problem: $\min_x \|Ax - b\|_2, A \in \mathbb{R}^{n\times m}$

C. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.16

Solving a least squares problem by normal equations

- The least squares problem: $\min_x \|Ax - b\|_2, A \in \mathbb{R}^{n \times m}$
- Algorithm:

C. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.16

Solving a least squares problem by normal equations

- The least squares problem: $\min_x \|Ax - b\|_2, A \in \mathbb{R}^{n \times m}$
- Algorithm:
  1. Compute gradient of $\|Ax - b\|_2$

C. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.16

Solving a least squares problem by normal equations

- The least squares problem: $\min_x \|Ax - b\|_2, A \in \mathbb{R}^{n\times m}$
- Algorithm:
  1. Compute gradient of $\|Ax - b\|_2$
  2. Obtain normal equation by setting gradient to zero: $A^T Ax = A^T b$

C. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.16

Solving a least squares problem by normal equations

- The least squares problem: $\min_x \|Ax - b\|_2, A \in \mathbb{R}^{n \times m}$
- Algorithm:
  1. Compute gradient of $\|Ax - b\|_2$
  2. Obtain normal equation by setting gradient to zero: $A^T Ax = A^T b$
  3. Solve normal equation and obtain solution to least squares problem

C. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Computing the eigenvalues of a matrix by finding the zeros of its characteristic polynomial.
- Finding the singular values by computing the eigenvalues of A^TA

Example 14.16

Solving a least squares problem by normal equations

- The least squares problem: $\min_x \|Ax - b\|_2, A \in \mathbb{R}^{n\times m}$
- Algorithm:
  1. Compute gradient of $\|Ax - b\|_2$
  2. Obtain normal equation by setting gradient to zero: $A^TAx = A^Tb$
  3. Solve normal equation and obtain solution to least squares problem
- Solving normal equation is not practical:
  1. Explicit formation of A^TA may cause errors(remember $a + b \neq b + a$)
  2. Normal equation is more sensitive to perturbations than $Ax = b$ and it can lead to solution with errors

D. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

D. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.17

Finding the eigenvalues of a matrix using its characteristic polynomial

D. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.17

Finding the eigenvalues of a matrix using its characteristic polynomial

- The eigenvalue problem:
  $Ax_i = \lambda_i x_i, A \in \mathbb{R}^{n \times n}, x_i \in \mathbb{R}^n, x_i \neq 0, \lambda_i \neq 0, i = 1, 2, ..n$

D. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.17

Finding the eigenvalues of a matrix using its characteristic polynomial

- The eigenvalue problem:
  $Ax_i = \lambda_i x_i, A \in \mathbb{R}^{n\times n}, x_i \in \mathbb{R}^n, x_i \neq 0, \lambda_i \neq 0, i = 1, 2, ..n$
- Algorithm:
  1. Define characteristic polynomial $|Ax - \lambda I|$
  2. Find zeros of characteristic polynomial, solve $|Ax - \lambda I| = 0$

D. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.17

Finding the eigenvalues of a matrix using its characteristic polynomial

- The eigenvalue problem:
  $Ax_i = \lambda_i x_i, A \in \mathbb{R}^{n \times n}, x_i \in \mathbb{R}^n, x_i \neq 0, \lambda_i \neq 0, i = 1, 2, ..n$
- Algorithm:
  1. Define characteristic polynomial $|Ax - \lambda I|$
  2. Find zeros of characteristic polynomial, solve $|Ax - \lambda I| = 0$
- Solving characteristic equation is not practical:
  1. perturbed coefficients are computed for characteristic polynomial
  2. Zeroes of certain polynomials are sensitive to perturbations, e.g. Wilkinson polynomial, $n = 20$

E. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

E. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.18

E. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.18

- Singular value decomposition

$$A = U\Sigma V^*, A \in \mathbb{R}^{m \times n}$$

- Σ is "diagonal" matrix with singular values $\sigma_i, i = 1, 2, ..n$ on the diagonal

E. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.18

- Singular value decomposition

$$A = U\Sigma V^*, A \in \mathbb{R}^{m\times n}$$

- Σ is "diagonal" matrix with singular values $\sigma_i, i = 1, 2, ..n$ on the diagonal
- Algorithm:
  1. Compute $A^T A$ and find it's eigenvalues $\lambda_i, i = 1, 2, ..n$
  2. Set $\sigma_i = \lambda_i, i = 1, 2, ..n$

E. Computational Difficulties of Theoretical Linear Algebra

- solving a linear system by Cramer's rule
- Computing the unique solution of a linear system by matrix inversion
- Solving least squares problem by normal equations
- Finding the eigenvalues of a matrix using characteristic polynomial
- Finding the singular values by computing the eigenvalues of $A^T A$

Example 14.18

- Singular value decomposition

$$A = U\Sigma V^*, A \in \mathbb{R}^{m\times n}$$

- Σ is "diagonal" matrix with singular values $\sigma_i, i = 1, 2, ..n$ on the diagonal
- Algorithm:
  1. Compute $A^T A$ and find it's eigenvalues $\lambda_i, i = 1, 2, ..n$
  2. Set $\sigma_i = \lambda_i, i = 1, 2, ..n$
- Algorithm not viable: explicit computing of $A^T A$ might introduce errors

Orthogonal projection, 1

Definition 14.19

- orthogonal vectors
- $x, y \in \mathbb{R}^n$
- $x \perp y \quad \equiv \quad x^T y = 0$, or $(x, y) = 0$, $(., .)$ - inner product

Definition 14.20

- V subspace of $\mathbb{R}^n$
- $x \in \mathbb{R}^n$
- $x \perp V$ if
- $(x, v) = 0 \;\; \forall v \in V$

Definition 14.21

- Orthogonal sets
- $V, W \neq \emptyset,\; V, W \subset \mathbb{R}^n$
- $V \perp W$ if $(v, w) = 0 \;\; \forall v \in V, \forall w \in W$

Orthogonal projection, 2

Definition 14.22

- $V \subset \mathbb{R}^n$
- Orthogonal complement $V^\perp = \{x : x \perp V, x \in \mathbb{R}^n\}$

Theorem 14.23

- V *is subspace of* $\mathbb{R}^n$
- $\Rightarrow$
- $V^\perp$ *is subspace of* $\mathbb{R}^n$
- $V^\perp \cap V = \emptyset$
- $\dim(V^\perp) + \dim(V) = n$
- $V^\perp \oplus V = \mathbb{R}^n \;\equiv\; \forall x \in \mathbb{R}^n \; x = p + o, p \in V, o \in V^\perp$

Orthogonal projection, 3

Definition 14.24

- V is subspace of $\mathbb{R}^n$
- $x \in \mathbb{R}^n$
- x_p is orthogonal projection of x onto V if
- $x_p \in V$
- $x - x_p \perp V$

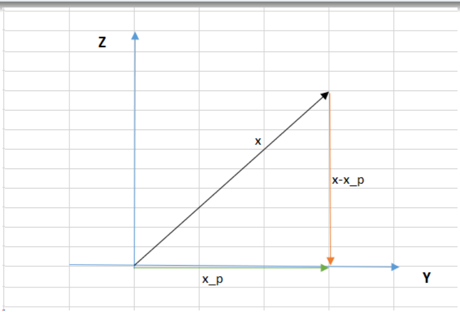


Figure: Orthogonal projection

Orthogonal projection, 4

Theorem 14.25

- $x \in \mathbb{R}^n$, *V is subspace of* $\mathbb{R}^n$
- x_p *is orthogonal projection of x onto V*
- $v \in V, v \neq x_p$
- $\Rightarrow$
- $\|x - x_p\|_2 < \|x - v\|_2$

Proof.

- $x - x_p \perp V$, $x_p - v \in V \Rightarrow (x - x_p, x_p - v) = 0$
- $x - v = x - x_p + x_p - v$
- $$\|x - v\|_2^2 = (x - v, x - v) = (x - x_p + x_p - v, x - x_p + x_p - v) =$$
$$\|x - x_p\|_2 * 2 + (x - x_p, x_p - v) + (x_p - v, x - x_p) + \|x_p - v\|_2^2 =$$
$$\|x - x_p\|_2^2 + \|x_p - v\|_2^2 > \|x - x_p\|_2^2$$

□

Orthogonal projection, 5

Theorem 14.26

- *$x \in \mathbb{R}^n$, V is subspace of $\mathbb{R}^n$*
- *x_p is orthogonal projection of x onto V*
- $v \in V, v \neq x_p$
- $\Rightarrow$
- $\|x - x_p\|_2 < \|x - v\|_2$

Theorem 14.27

- *X linear space with inner product $(.,.)$*
- *$V \subset X$, is subspace of X*
- $x \in X, p, v \in V, p \neq v$
- $x - p \perp V$
- $\Rightarrow$
- $\|x - p\| < \|x - v\|$

Orthogonal projection, 6

Theorem 14.28

- *X linear space with inner product $(.,.)$*
- *$V \subset X$, is subspace of X*
- $x \in X, p, v \in V, p \neq v$
- $x - p \perp V$
- $\Rightarrow$
- $\|x-p\| < \|x-v\|$

Theorem 14.29

- *X linear space with inner product $(.,.)$*
- *$V \subset X$, is subspace of X*
- $x \in X, p \in V$
- $\|x-p\| < \|x-v\|, \forall v \in V, p \neq v$
- $\Rightarrow$
- $x - p \perp V$

Orthogonal projection, 7

Theorem 14.30

- *X linear space with inner product $(.,.)$*
- *$V \subset X$, is subspace of X*
- $x \in X, p \in V$
- $\|x - p\| < \|x - v\|, \forall v \in V, p \neq v$
- $\Rightarrow$
- $x - p \perp V$

Proof.

- $w \in V, p - tw \in V, t \in \mathbb{R}$
- $f(t) = \|x - p + tw\|^2$
- $f(0) = \|x - p\|^2 < \|x - p + tw\|^2, t \neq 0$
- $\Rightarrow f(0) = \min_{t \in \mathbb{R}} f(t),\ t_{min} = arg \min_{t \in \mathbb{R}} f(t),\ t_{min} = 0$

$\square$

Orthogonal projection, 8

Proof.

- $\forall w \in V, p - tw \in V, t \in \mathbb{R}$
- $f(t) = \|x - p + tw\|^2$
- $f(0) = \|x - p\|^2 < \|x - p + tw\|^2, t \neq 0$
- $\Rightarrow f(0) = \min_{t\in\mathbb{R}} f(t),\ t_{min} = arg\min_{t\in\mathbb{R}} f(t),\ t_{min} = 0$
- $$f(t) = (x + p - tw, x - p + tw) = (x - p, x - p) + (x - p, tw) + (tw, x - p) + (tw, tw) = \|x - p\|^2 + 2t(x - p, w) + t^2\|w\|^2$$
- $f'(t) = 2(x - p, w) + 2t\|w\|^2$
- $f'(t_{min}) = 0 \Rightarrow f'(0) = 0 \Rightarrow 2(x - p, w) = 0$
- $(x - p, w) = 0\ \forall w \in V \Rightarrow x - p \perp V$

$\square$

Conjugate .., 1

Definition 14.31

- $A \in \mathbb{R}^{n \times n}, A = A^T > 0$
- Inner product - $(x, y)_A = (Ax, y), \ x, y \in \mathbb{R}^n$
- Energetic norm, A-norm - $\|x\|_A = (Ax, x), x \in \mathbb{R}^n$

Definition 14.32

- A-conjugate vectors (A-orthogonal vectors) x, y
- $x, y \in \mathbb{R}^n \setminus \{0\}$
- $A \in \mathbb{R}^{n \times n}, A = A^T > 0$
- if
- $x^T A y = 0,$
- or $(Ax, y) = 0$, $(.,.)$ - inner product
- or $(x, y)_A = 0$

Conjugate .., 2

Theorem 14.33

Projection onto subspace and quadratic form

- $A \in \mathbb{R}^{n\times n}, A = A^T > 0$
- $Ax_* = b, x_*, b \in \mathbb{R}^n$
- $V_k \subset \mathbb{R}^n,\; V_k = span\{v_1, v_2, ..., v_k\}$
- $v_1, v_2, ..., v_k$ - *linearly independent*
- $\Rightarrow$
- $x_k = arg\min_{x\in V_k} \|x - x_*\|_A$
- $x_k = arg\min_{x\in V_k} f(x),\; f(x) = 0.5(Ax, x) - (b, x)$

Proof.

$$\|x - x_*\|_A^2 = (A(x - x_*), x - x_*) =$$
$$(Ax, x) - (Ax_*, x) - (Ax, x_*) + (Ax_*, x_*) =$$
$$(Ax, x) - 2(Ax_*, x) + (Ax_*, x_*) = (Ax, x) - 2(b, x) + (Ax_*, x_*) =$$
$$2(0.5(Ax, x) - (b, x)) + (Ax_*, x_*) = 2f(x) + \|x_*\|_A^2$$

Conjugate .., 3

Theorem 14.34

Projection onto subspace and quadratic form

- $A \in \mathbb{R}^{n\times n}, A = A^T > 0$
- $Ax_* = b, x_*, b \in \mathbb{R}^n$
- $V_k \subset \mathbb{R}^n,\ \ V_k = span\{v_1, v_2, ..., v_k\}$
- $v_1, v_2, ..., v_k$ - *linearly independent*
- $\Rightarrow$
- $x_k = arg\min_{x\in V_k} \|x - x_*\|_A$
- $x_k = arg\min_{x\in V_k} f(x),\ \ f(x) = 0.5(Ax, x) - (b, x)$

Conclusion: A-orthogonality is important:

- if $x_k - x_*$ is $A-$orthogonal to V_k then

$$f(x_k) = min_{x\in V_k} f(x)$$

- The inverse is also correct

Conjugate .., 4

Conclusion: A-orthogonality is important:

- if error $e^{(k)} = x_k - x_*$ is $A-$orthogonal to V_k then

$$f(x_k) = min_{x \in V_k} f(x)$$

- The inverse is also correct

Example 14.35

Check $A-$orthogonality:

- $V_k = span\{v_1, v_2, ..., v_k\}$
- $e^{(k)} = x_k - x_*$ is $A-$orthogonal to V_k: $e^{(k)} \perp^A V_k$
- $e^{(k)} \perp^A V_k \Rightarrow$

$$\begin{aligned}
(e^{(k)}, v_i)_A &= 0, i = 1, 2, ..., k \\
(Ae^{(k)}, v_i) &= 0, i = 1, 2, ..., k \\
(A(x_k - x_*), v_i) &= 0, i = 1, 2, ..., k \\
(Ax_k - b, v_i) &= 0, i = 1, 2, ..., k
\end{aligned}$$

- The approach is general: works for linear spaces with inner product!

Conjugate .., 5

Example 14.36

- $V_k = span\{v_1, v_2, ..., v_k\}$
- $e^{(k)} = x_k - x_*$ is $A-$orthogonal to V_k: $e^{(k)} \perp^A V_k$
- $e^{(k)} \perp^A V_k \Rightarrow$

$$
\begin{aligned}
(e^{(k)}, v_i)_A &= 0, i = 1, 2, ..., k \\
(Ae^{(k)}, v_i) &= 0, i = 1, 2, ..., k \\
(A(x_k - x_*), v_i) &= 0, i = 1, 2, ..., k \\
(Ax_k - b, v_i) &= 0, i = 1, 2, ..., k
\end{aligned}
$$

- The approach is general: works for linear spaces with inner product!
- Ritz method
- $A = \frac{d^2}{dz^2}$, $(\varphi, g) = \int_a^b \varphi(z)g(z)dz$
- $(\varphi, g)_A = \int_a^b \frac{d^2}{dz^2}\varphi(z)g(z)dz$, $+$homogeneous boundary conditions
- $\frac{d^2u(z)}{dz^2} = c(z)$ is similar to $Ax = b$

Conjugate .., 6

Conclusion: A-orthogonality is important:

- if error $e^{(k)} = x_k - x_*$ is $A-$orthogonal to V_k then

$$f(x_k) = min_{x \in V_k} f(x)$$

- The inverse is also correct

Example 14.37

Some iterative methods connected to quadratic form

- Steepest descent
- Minimal residuals

$A-$orthogonality?

Method of orthogonal directions, 1

Steepest descent with orthogonal directions

- Pick orthogonal directions $d_0, d_1, ..., d_{n-1},\ d_i \in \mathbb{R}^n, i = 0, 1, 2, ..., n-1$
- Do one step in each direction with right length
- Done after n steps

22 Jonathan Richard Shewchuk

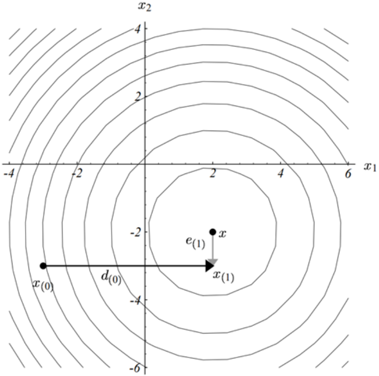


Figure 21: The Method of Orthogonal Directions. Unfortunately, this method only works if you already know the answer.

Method of orthogonal directions, 2

22 Jonathan Richard Shewchuk

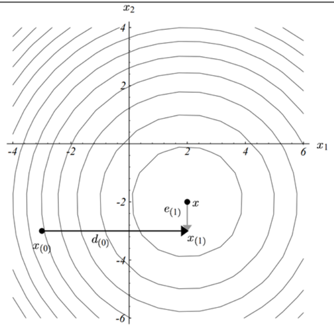


Figure 21: The Method of Orthogonal Directions. Unfortunately, this method only works if you already know the answer.

- $x^{(k+1)} = x^{(k)} + \alpha_k d_k$
- $e^{(k+1)} \perp d_k \Rightarrow (e^{(k+1)}, d_k) = 0$
- $e^{(k+1)} = x^{(k+1)} - x, (x^{(k+1)} - x, d_k) = 0$
- $(x^{(k)} + \alpha_k d_k - x, d_k) = 0 \Rightarrow \alpha_k = \frac{(d_k, e^{(k)})}{(d_k, d_k)}$
- Cannot compute α_k !!

Method of orthogonal directions, 3

- $x^{(k+1)} = x^{(k)} + \alpha_k d_k$
- $e^{(k+1)} \perp d_k \Rightarrow (e^{(k+1)}, d_k) = 0$
- $e^{(k+1)} = x^{(k+1)} - x, (x^{(k+1)} - x, d_k) = 0$
- $(x^{(k)} + \alpha_k d_k - x, d_k) = 0 \Rightarrow \alpha_k = \frac{(d_k, e^{(k)})}{(d_k, d_k)}$
- Cannot compute α_k !!
- What if different inner product used?
- $e^{(k)}$ is unknown
- $Ae^{(k)} = A(x^{(k)} - x) = Ax^{(k)} - Ax = Ax^{(k)} - b = -r_k$
- r_k can be computed for any $x^{(k)}$!!
- Conjugated directions?

Conjugated directions algorithm, 1

- $d_0, d_1, ..., d_{n-1},\ d_i \perp^A d_j,\ d_i \in \mathbb{R}^n,\ i,j = 0,1,2,...,n-1$
- $A = A^T > 0, A \in \mathbb{R}^{n\times n}$

$$x^{(k+1)} = x^{(k)} + \alpha_k d_k$$

$$\alpha_k = arg \min_{\alpha\in\mathbb{R}} f(x^{(k)} + \alpha d_k)$$

$$f(x) = 0.5(Ax, x) - (b, x)$$

Computing α_k similar to steepest descent

- $\frac{df(x^{(k)}+\alpha d_k)}{d\alpha} = 0$

$$\frac{df(x^{(k)} + \alpha d_k)}{d\alpha} =$$

$$\frac{d}{d\alpha}[0.5(A(d_k, x^{(k)} + \alpha d_k), d_k, x^{(k)} + \alpha d_k) + (b, d_k, x^{(k)} + \alpha d_k)] =$$

$$\frac{d}{d\alpha}[0.5(Ax^{(k)}, x^{(k)}) + 2\alpha(Ax^{(k)}, d_k) + \alpha_k^2(Ad_k, d_k) - (b, x^{(k)}) - \alpha(b, d_k$$

$$= (Ax^{(k)}, d_k) - (b, d_k) + \alpha(Ad_k, d_k) = -(r_k, d_k) + \alpha(Ad_k, d_k)$$

- $\alpha_k = \frac{(r_k,d_k)}{(Ad_k,d_k)}$

Conjugated directions algorithm, 2

- $d_0, d_1, ..., d_{n-1},\ d_i \perp^A d_j,\ d_i \in \mathbb{R}^n,\ i,j = 0,1,2,...,n-1$
- $A = A^T > 0, A \in \mathbb{R}^{n \times n}$
- $k = 0$
- Do until accuracy criterion satisfied

$$r_k = b - Ax^{(k)}$$
$$\alpha_k = \frac{(r_k, d_k)}{(Ad_k, d_k)}$$
$$x^{(k+1)} = x^{(k)} + \alpha_k d_k$$

- converges to exact solution in n iterations

Conjugated gradients algorithm

$$d_{(0)} = r_{(0)} = b - Ax_{(0)},$$

$$\alpha_{(i)} = \frac{r_{(i)}^T r_{(i)}}{d_{(i)}^T A d_{(i)}} \qquad \text{(by Equations 32 and 42)},$$

$$x_{(i+1)} = x_{(i)} + \alpha_{(i)} d_{(i)},$$

$$r_{(i+1)} = r_{(i)} - \alpha_{(i)} A d_{(i)},$$

$$\beta_{(i+1)} = \frac{r_{(i+1)}^T r_{(i+1)}}{r_{(i)}^T r_{(i)}},$$

$$d_{(i+1)} = r_{(i+1)} + \beta_{(i+1)} d_{(i)}.$$

Figure: Conjucated Gradients, source J.R.Shevchuk

- conjugated directions are defined for each iteration
- converges in *n* iterations

Q & A