Optimization and Numerical Analysis: Solving Linear Systems

Robert Gower

September 20, 2020

Table of Contents

The Problem: Linear Systems

One of the most common and fundamental problems in numerical computing is to solve a linear system:

$$Ax = b$$

where $x \in \mathbb{R}^n$ is *unknown*, $A \in \mathbb{R}^{m\times n}$, and $b \in \mathbb{R}^m$ are given.

$$A = (a_{ij}) = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{d1} & a_{d2} & a_{d3} & \dots & a_{dn} \end{bmatrix}.$$

- Normal matrices: $AA^\top = A^\top A$
- Symmetric matrices: $(a_{ij}) = A = A^\top = (a_{ji})$
- Orthogonal matrices: $AA^\top = A^\top A = I$,

where $I = (\delta_{ij})$ denotes the identity matrix.

What does it mean to be close to a solution?

First we generalize the notation of distance by defining a norm

Definition

We say that the function $\|\cdot\| : x \in \mathbb{R}^n \to R_+$ is a norm if it is

- **Point separating:** $\|x\| = 0 \Leftrightarrow x = 0, \forall x \in E$.
- **Subadditive:** $\|x + y\| \leq \|x\| + \|y\|, \forall x, y \in E$
- **Homogeneous:** $\|ax\| = |a|\|x\|, \forall x \in E, a \in \mathbb{R}$.

The L2 norm: $\|x\|_2 \stackrel{\text{def}}{=} \sqrt{\sum_{i=1}^{n} x_i^2}$.

The L1 norm: $\|x\|_1 \stackrel{\text{def}}{=} \sum_{i=1}^{n} |x_i|$.

What does it mean to be close to a solution?

First we generalize the notation of distance by defining a norm

Definition

We say that the function $\|\cdot\| : x \in \mathbb{R}^n \to R_+$ is a norm if it is

- **Point separating:** $\|x\| = 0 \Leftrightarrow x = 0, \forall x \in E$.
- **Subadditive:** $\|x + y\| \leq \|x\| + \|y\|, \forall x, y \in E$
- **Homogeneous:** $\|ax\| = |a|\|x\|, \forall x \in E, a \in \mathbb{R}$.

The L2 norm: $\|x\|_2 \overset{\text{def}}{=} \sqrt{\sum_{i=1}^{n} x_i^2}$.

The L1 norm: $\|x\|_1 \overset{\text{def}}{=} \sum_{i=1}^{n} |x_i|$.

Exercise: Show that $\|Vy\|_2 = \|y\|_2$ for every $y \in \mathbb{R}^n$ and orthogonal matrix $V \in \mathbb{R}^{n \times n}$.

We can define an *induced* norm over matrices by using vector norms. Let $\|\cdot\| : \mathbb{R}^n \to \mathbb{R}_+$ be a norm.

$$\|A\| \stackrel{\text{def}}{=} \sup_{x\in\mathbb{R}^n, x\neq 0} \frac{\|Ax\|}{\|x\|}.$$

In particular the L2 induced norm is

$$\|A\|_2 \stackrel{\text{def}}{=} \sup_{x\in\mathbb{R}^n, x\neq 0} \frac{\|Ax\|_2}{\|x\|_2}.$$

Exercise

Show that all induced norms satisfy

$\|Ax\| \leq \|A\|\|x\|, \forall x \in \mathbb{R}^n,$

and are submultiplicative. Also show that for $B \in \mathbb{R}^{n\times n}$ we have

$\|AB\|_2 = \|A\|_2\|B\|_2.$

$\|O\|_2 = 1, \quad \forall O \in \mathbb{R}^{n\times n}$ orthonormal matrix.

Other Matrix Norms and Operators

If $A \in \mathbb{R}^{n \times n}$ is a square matrix we can define:

Trace: $\mathrm{Tr}\,(A) \overset{\text{def}}{=} \sum_{i=1}^{n} a_{ii}$

Frobenius norm: $\|A\|_E \overset{\text{def}}{=} \sqrt{\sum_{i,j=1}^{n,m} a_{ij}^2} = \sqrt{\mathrm{Tr}\,(A^\top A)}$

L1 norm: $\|A\|_\infty \overset{\text{def}}{=} \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|_1}{\|x\|_1}.$

Exercise

Let $A, B \in \mathbb{R}^{n \times n}$ and let $O \in \mathbb{R}^{n \times n}$ be an orthogonal matrix. Prove

$$\mathrm{Tr}\,(AB) = \mathrm{Tr}\,(BA)$$

$$\|O^\top A O\|_E = \|A\|_E.$$

Can we get close to a solution $Ax = b$?

$$\begin{bmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 32 \\ 23 \\ 33 \\ 31 \end{bmatrix} \quad \text{with solution} \quad x = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Let us say we find a solution x' that is *close* in the sense that

$$\begin{bmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{bmatrix} \quad \text{with solution} \quad x' = \begin{bmatrix} 9.2 \\ -12.6 \\ 4.5 \\ -1.1 \end{bmatrix}$$

An error on right hand side b of the order of $1/300$ has incurred a significant error in the solution x' of an order of 10.

This large error is due to the *condition number* of A. In algebra

$$A(x+\delta x) = b + \delta b. \tag{1}$$

How big can $\|\delta x\|$ be? Since we know $Ax = b$ we have that

$$A\delta x = \delta b.$$

Assuming A is invertible and left multiplying A^{-1} on both sides

$$\delta x = A^{-1}\delta b \quad \Rightarrow \quad \|\delta x\| \leq \|A^{-1}\|\|\delta b\|.$$

Furthermore $\|b\| = \|Ax\| \leq \|A\|\|x\|$ and thus

$$\frac{1}{\|x\|} \leq \|A\|\frac{1}{\|b\|}.$$

Putting the two above equations together gives

$$\frac{\|\delta x\|}{\|x\|} \leq \underbrace{\|A^{-1}\|\|A\|}_{\stackrel{\text{def}}{=}\mathrm{cond}(A)}\frac{\|\delta b\|}{\|b\|}.$$

$$\frac{\|\delta x\|}{\|x\|} \leq \underbrace{\|A^{-1}\|\|A\|}_{\overset{\text{def}}{=}\text{cond}(A)} \frac{\|\delta b\|}{\|b\|}.$$

Definition

We call $\text{cond}(A) = \|A\|\|A^{-1}\|$ the *condition number* of A.

Similarly, small errors in A can also introduce large changes in x and this also depends on the condition number through

$$\frac{\|\delta x\|}{\|x + \delta x\|} \leq \underbrace{\|A^{-1}\|\|A\|}_{\overset{\text{def}}{=}\text{cond}(A)} \frac{\|\delta A\|}{\|A\|},$$

where $\delta A \in \mathbb{R}^{m \times n}$ is the error in A.

Properties of the Condition Number

Theorem

- $cond(A) \geq 1$
- $cond(A) = cond(A^{-1})$
- $cond(\alpha A) = cond(A)$, *for every* $\alpha \neq 0$.
- $cond(O) = 1$ *for every orthonormal matrix* $O \in \mathbb{R}^{n \times n}$.

First we solve the easiest system: Triangular systems. For instance *lower triangular* $Ax = b$ where

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n-1} & a_{1n} \\ 0 & a_{22} & \dots & a_{2n-1} & a_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ & & \dots & a_{n-1n-1} & a_{n-1n} \\ 0 & 0 & \dots & 0 & a_{nn} \end{bmatrix}.$$

In other words

$$\sum_{j=i}^{n} a_{ij} x_j = b_i, \quad \text{for } i = 1, \dots, n. \tag{2}$$

Two efficient algorithms for solving triangular linear systems: **forward substitution** and **backward substitution**.

Backwards substitution method: Starting with $i = n$ we have

$$a_{nn}x_n = b_n.$$

Assuming that $a_{nn} \neq 0$ (otherwise there is no solution) we have that

$$x_n = b_n/a_{nn}.$$

For $i < n$, separating out the x_i term in (2) we have

$$\sum_{j=i+1}^{n} a_{ij}x_j + a_{ii}x_i = b_i. \tag{3}$$

Assuming $a_{ii} \neq 0$ and isolating x_i gives

$$x_i = \frac{b_i - \displaystyle\sum_{j=i+1}^{n} a_{ij}x_j}{a_{ii}}. \tag{4}$$

Algorithm 1 Backward substitution

for $i = n, \ldots, 1$ **do**

$$x_i = \frac{b_i - \sum_{j=i+1}^{n} a_{ij} x_j}{a_{ii}}.$$

Exercise

How many floating point operations does backward substitution cost?

Proof: For a fixed i there are $n - (i + 1)$ summations and multiplications in $\sum_{j=i+1}^{n} a_{ij} x_j$. Consequently there are $2(n - i)$ operations to compute $\frac{b_i - \sum_{j=i+1}^{n} a_{ij} x_j}{a_{ii}}$. Summing up over $i = 1, \ldots n$ we have a total of operations given by

$$\sum_{i=1}^{n} 2(n - i) = 2n^2 - n(n + 1) = n(n - 1).$$

Exercise

What can we do if we find $a_{ii} = 0$? What does it say about this triangular system if $a_{ii} = 0$?

Conclusion: Triangular linear systems are easy to solve.

Idea: Transform all linear systems into triangular systems?

Can we transform A into an upper triangular matrix?

Can we transform A into an upper triangular matrix?

Yes, using *invertible operations*.

Theorem (Invertible operations)

Let $P \in \mathbb{R}^{n \times n}$ be an invertible matrix. Show that

$$\{x \ : \ Ax = b\} \quad = \quad \{x \ : \ PAx = Pb\}$$

Gaussian Elimination Idea: Use sequence of invertible operations $P_1, \ldots, P_k$ such that

$$P_k \cdots P_2 P_1 A = U.$$

Then solve

$$Ux = P_k \cdots P_2 P_1 b.$$

Example of Gaussian Elimination

Consider the linear system

$$
\begin{array}{cccccc}
2x_1 & x_2 & -3x_3 & = & 5 \\
4x_1 & x_2 & 5x_3 & = & -1 \\
10x_1 & -7x_2 & 13x_3 & = & -3
\end{array}
$$

We want to isolate x_1 on the top row.

Example of Gaussian Elimination

Consider the linear system

$$\begin{array}{cccccc} 2x_1 & x_2 & -3x_3 & = & 5 \\ 4x_1 & x_2 & 5x_3 & = & -1 \\ 10x_1 & -7x_2 & 13x_3 & = & -3 \end{array}$$

We want to isolate x_1 on the top row. *Subtracting* two times the first row to the second row ($R_2 \leftarrow R_2 - 2R_1$) gives

$$\begin{array}{cccccc} 2x_1 & x_2 & -3x_3 & = & 5 \\ & -x_2 & 11x_3 & = & -11 \\ 10x_1 & -7x_2 & 13x_3 & = & -3 \end{array}$$

Example of Gaussian Elimination

Consider the linear system

$$\begin{array}{cccccc} 2x_1 & x_2 & -3x_3 & = & 5 \\ 4x_1 & x_2 & 5x_3 & = & -1 \\ 10x_1 & -7x_2 & 13x_3 & = & -3 \end{array}$$

We want to isolate x_1 on the top row. *Subtracting* two times the first row to the second row ($R_2 \leftarrow R_2 - 2R_1$) gives

$$\begin{array}{cccccc} 2x_1 & x_2 & -3x_3 & = & 5 \\ & -x_2 & 11x_3 & = & -11 \\ 10x_1 & -7x_2 & 13x_3 & = & -3 \end{array}$$

Subtracting five times the first row to the third row ($R_3 \leftarrow R_3 - 5R_1$) gives

$$\begin{array}{cccccc} 2x_1 & x_2 & -3x_3 & = & 5 \\ & -x_2 & 11x_3 & = & -11 \\ & -12x_2 & 28x_3 & = & -28 \end{array}$$

$$\begin{array}{cccccc} 2x_1 & x_2 & -3x_3 & = & 5 \\ & -x_2 & 11x_3 & = & -11 \\ & -12x_2 & 28x_3 & = & -28 \end{array}$$

Now isolate x_2 on the second row by $R_3 \leftarrow R_3 + 12R_2$ giving

$$\begin{array}{cccccc} 2x_1 & x_2 & -3x_3 & = & 5 \\ & -x_2 & 11x_3 & = & -11 \\ & -12x_2 & 28x_3 & = & -28 \end{array}$$

Now isolate x_2 on the second row by $R_3 \leftarrow R_3 + 12R_2$ giving

$$\begin{array}{cccccc} 2x_1 & x_2 & -3x_3 & = & 5 \\ & -x_2 & 11x_3 & = & -11 \\ & & -104x_3 & = & 104 \end{array}$$

Now we have an upper triangular system! Easy to solve.
But what were these operations, e.g. $R_3 \leftarrow R_3 + 12R_2$? Are they invertible operations? YES

Let $A^0 = A$ and let $A^k = P_{k-1}A^{k-1}$ where $a_{ij}^k = 0$ for $1 \le j \le k$ and $i \ge j+1$. To generate A^{k+1} from A^k we need to perform a *row operation*.

$$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \vdots & 0 & 0 \\ \vdots & & 1 & 0 & 0 & \vdots \\ \vdots & \vdots & -a_{k+1k}^k / a_{kk}^k & 1 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & -a_{nk}^k / a_{kk}^k & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11}^k & a_{12}^k & a_{13}^k & \dots & a_{1n}^k \\ 0 & \ddots & \vdots & \vdots & a_{2n}^k \\ \vdots & 0 & a_{kk}^k & \vdots & \vdots \\ \vdots & 0 & a_{k+1k}^k & \dots & a_{k+1n}^k \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & a_{nk}^k & \dots & a_{nn}^k \end{bmatrix}$$

$$= \begin{bmatrix} a_{11}^k & a_{12}^k & a_{13}^k & \dots & a_{1(k+1)}^k & \dots & a_{1n}^k \\ 0 & \ddots & \vdots & \dots & a_{2(k+1)}^k & \dots & a_{2n}^k \\ \vdots & 0 & a_{kk}^k & \vdots & \vdots & \dots & \vdots \\ \vdots & 0 & 0 & \vdots & a_{(k+1)(k+1)}^{k+1} & \dots & a_{(k+1)n}^{k+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & & \\ 0 & 0 & 0 & \dots & a_{n(k+1)}^{k+1} & \dots & a_{nn}^{k+1} \end{bmatrix}$$

These row operations can be represented in a much more compact.

$$P_k = I - v_k e_k^\top, \tag{5}$$

where $e_k = (0, \cdots, \underset{kth}{1}, 0, \cdots, 0) \in \mathbb{R}^n$ is the kth unit coordinate vector and $v_k = (0, \ldots, 0, \underset{(k+1)th}{\frac{a_{k+1k}^k}{a_{kk}^k}}, \ldots, \frac{a_{nk}^k}{a_{kk}^k})$. With this notation we can write

$$P_k A^k = A^{k+1}.$$

Also these row operations are invertible!

Lemma

Let P_k be the kth row operation. It follows

1. $P_k^{-1} = I + v_k e_k^\top$. *(Invertible)*
2. $P_{k-1}^{-1} P_k^{-1} = I + v_k e_k^\top + v_{k-1} e_{k-1}^\top$ *(Compositions are lower triangular)*

Lemma

Let P_k be the kth row operation. It follows

1. $P_k^{-1} = I + v_k e_k^\top$.
2. $P_{k-1}^{-1} P_k^{-1} = I + v_k e_k^\top + v_{k-1} e_{k-1}^\top$

Proof.

1. By direct computation we have

$$(I + v_k e_k^\top)(I - v_k e_k^\top) = I + v_k e_k^\top - v_k e_k^\top - v_k e_k^\top v_k e_k^\top = I - v_k e_k^\top v_k e_k^\top .$$

   The support of v_k does not intersect with the support of e_k thus $e_k^\top v_k = 0$.

2. Again by computation

$$P_{k-1}^{-1} P_k^{-1} = (I + v_{k-1} e_{k-1}^\top)(I + v_k e_k^\top) = I + v_{k-1} e_{k-1}^\top + v_k e_k^\top + v_{k-1}(e_{k-1}^\top v_k) e_k^\top .$$

   This inner product $e_{k-1}^\top v_k$ is between two vector with disjoint support, thus $e_{k-1}^\top v_k = 0$ and the result follows.

Gaussian Elimination overview

Gaussian elimination applies n row operations until the matrix is upper triangular

$$P_n P_{n-1} \cdots P_1 A = U. \tag{6}$$

Then solves the upper triangular system

$$Ux = P_n P_{n-1} \cdots P_1 b.$$

The cost of applying P_k is $(n-k-1)n$ consequently the cost of performing (6) is

$$\sum_{k=1}^{n} (n-k-1)n = O(n^3).$$

Choosing a Pivot

Three strategies

- Default: Choosing a_{kk} as the pivot.
- Partial Pivot: On column k we choosing the element below the diagonal with the largest absolute value

$$i_{\text{pivot}} = \arg\max_{i \geq k} |a_{ik}|.$$

- Total Pivot: Choose the largest element below or to the right of the diagonal

$$(i_{\text{pivot}}, j_{\text{pivot}}) = \arg\max_{i,j \geq k} |a_{ij}|.$$

Both Partial and Total pivoting improves numerical stability of Gaussian Elimination.

Gaussian Elimination gives a Triangular Decomposition

Since by Lemma 6 the matrix P_k is invertible we have that the product of row operations in (6) is also invertible with

$$(P_n P_{n-1} \cdots P_1)^{-1} = P_1^{-1} \cdots P_{n-1}^{-1} P_n^{-1} \overset{\text{def}}{=} L. \tag{7}$$

Again by Lemma 6 and induction we have that L is lower triangular. Left multiplying (6) by L we have

$$A = LU. \tag{8}$$

This is known as the LU decomposition. This decomposition can be used to efficiently solve multiple linear systems

$$Ax^i = b_i, \quad \text{for } = 1, \ldots, 10.$$

Each system $Ax = b_i$ can be solved with two triangular solves

$$\begin{array}{rrcl} \text{First lower triangular solve}: & Ly & = & b_i \\ \text{Second upper triangular solve}: & Ux^i & = & y \end{array}$$

Each system $Ax = b_i$ can be solved with two triangular solves

$$\begin{array}{rrcl} \text{First lower triangular solve}: & Ly & = & b_i \\ \text{Second upper triangular solve}: & Ux^i & = & y \end{array}$$

The two together give: $\quad Ly = b \quad \Leftrightarrow \quad L\underbrace{Ux^i}_{y} = b_i \quad \Leftrightarrow \quad Ax^i = b_i.$

Thus cost of solving each system if $O(n^2)$.

Theorem

Let $A \in \mathbb{R}^{n\times n}$ be an invertible matrix such that the submatrix

$$A_{1:k,1:k} \stackrel{def}{=} \begin{bmatrix} a_{11} & \dots & a_{1k} \\ \vdots & \vdots & \vdots \\ a_{k1} & \dots & a_{kk} \end{bmatrix} \quad \text{is invertible for } k = 1, \dots, n.$$

Then the LU decomposition exists. If $L_{ii} = 1$ is enforced, the decomposition is unique.

Gauss Jordan Method for Inversion

We can use Gaussian elimination to compute the inverse of A.
Setup the systems

$$Ax^i = e_i, \quad \text{for } i = 1, \ldots, n.$$

Gauss Jordan Method for Inversion

We can use Gaussian elimination to compute the inverse of A.
Setup the systems

$$Ax^i = e_i, \quad \text{for } i = 1, \ldots, n.$$

In other words

$$AX \stackrel{\text{def}}{=} A[x^1, \ldots, x^n] = I.$$

Thus the solution is $X = A^{-1}$.

Gauss Jordan Method for Inversion

We can use Gaussian elimination to compute the inverse of A.
Setup the systems

$$Ax^i = e_i, \quad \text{for } i = 1, \ldots, n.$$

In other words

$$AX \stackrel{\text{def}}{=} A[x^1, \ldots, x^n] = I.$$

Thus the solution is $X = A^{-1}$.
Apply row operations until A is the identity matrix. That is

$$P_k \cdots P_1 A = I.$$

Consequently

$$P_k \cdots P_1 AX = X = P_k \cdots P_1 I = A^{-1}.$$

Thus apply the same operations simultaneously to the identity matrix to get A^{-1}.

Example of Gauss Jordan (and Partial Pivot)

Let us invert the following matrix

$$
\begin{array}{cccccc}
x_1 & -3x_2 & 14x_3 & = & 1 & 0 & 0 \\
x_1 & -2x_2 & 10x_3 & = & 0 & 1 & 0 \\
-2x_1 & 4x_2 & -19x_3 & = & 0 & 0 & 1
\end{array}
$$

Using a partial pivot gives ($R_1 \leftrightarrow R_3$)

$$
\begin{array}{cccccc}
-2x_1 & 4x_2 & -19x_3 & = & 0 & 0 & 1 \\
x_1 & -2x_2 & 10x_3 & = & 0 & 1 & 0 \\
x_1 & -3x_2 & 14x_3 & = & 1 & 0 & 0
\end{array}
$$

$$\begin{array}{cccccc} -2x_1 & 4x_2 & -19x_3 & = & 0 & 0 & 1 \\ x_1 & -2x_2 & 10x_3 & = & 0 & 1 & 0 \\ x_1 & -3x_2 & 14x_3 & = & 1 & 0 & 0 \end{array}$$

Now isolating x_1 using row operations

$$\begin{array}{ccccccc} -2x_1 & 4x_2 & -19x_3 & = & 0 & 0 & 1 \\ x_1 & -2x_2 & 10x_3 & = & 0 & 1 & 0 \\ x_1 & -3x_2 & 14x_3 & = & 1 & 0 & 0 \end{array}$$

Now isolating x_1 using row operations $R_2 \leftarrow R_2 + \frac{1}{2}R_1$ and $R_3 \leftarrow R_3 + \frac{1}{2}R_1$ gives

$$\begin{array}{ccccccc} -2x_1 & 4x_2 & -19x_3 & = & 0 & 0 & 1 \\ 0 & 0 & 1/2x_3 & = & 0 & 1 & 1/2 \\ 0 & -x_2 & 9/2x_3 & = & 1 & 0 & 1/2 \end{array}$$

Second phase: Using a total pivot gives $C_2 \leftrightarrow C_3$ and $R_2 \leftrightarrow R_3$

$$\begin{array}{ccccccc} -2x_1 & -19x_3 & 4x_2 & = & 0 & 0 & 1 \\ 0 & 9/2x_3 & -x_2 & = & 1 & 0 & 1/2 \\ 0 & 1/2x_3 & 0 & = & 0 & 1 & 1/2 \end{array}$$

$$\begin{array}{ccccccc} -2x_1 & -19x_3 & 4x_2 & = & 0 & 0 & 1 \\ 0 & 9/2x_3 & -x_2 & = & 1 & 0 & 1/2 \\ 0 & 1/2x_3 & 0 & = & 0 & 1 & 1/2 \end{array}$$

Isolating x_3 gives

$$\begin{array}{ccccccc} -2x_1 & 0 & 4x_2 & = & 38/9 & 0 & 28/9 \\ 0 & 9/2x_3 & -x_2 & = & 1 & 0 & 1/2 \\ 0 & 0 & 1/9x_2 & = & -1/9 & 1 & 4/9 \end{array}$$

Isolating x_2 gives

$$\begin{array}{ccccccc} -2x_1 & 0 & 0 & = & 4 & 2 & 4 \\ 0 & 9/2x_3 & 0 & = & 0 & 9 & 9/2 \\ 0 & 0 & 1/9x_2 & = & -1/9 & 1 & 4/9 \end{array}$$

$$\begin{array}{ccccccc} -2x_1 & 0 & 0 & = & 4 & 2 & 4 \\ 0 & 9/2x_3 & 0 & = & 0 & 9 & 9/2 \\ 0 & 0 & 1/9x_2 & = & -1/9 & 1 & 4/9 \end{array}$$

Finally scaling the rows : $R_1 \leftarrow -1/2R_1$
$R_2 \leftarrow 2/9R_2$
$R_3 \leftarrow 9R_3$
and switching $R_2 \leftrightarrow R_3$ gives

$$\begin{array}{ccccccc} x_1 & 0 & 0 & = & -2 & -1 & -2 \\ 0 & 0 & x_2 & = & -1 & 9 & 4 \\ 0 & x_3 & 0 & = & 0 & 2 & 1 \end{array}$$

Consequently

$$A^{-1} = \begin{bmatrix} -2 & -1 & -2 \\ -1 & 9 & 4 \\ 0 & 2 & 1 \end{bmatrix}$$

Cholesky Decomposition

We say a matrix is positive definite if it is symmetric and if

$$v^\top A v > 0, \quad \forall v \neq 0. \tag{9}$$

For positive definite matrices we can efficiently compute an LU decomposition with $L = U^\top$.

Theorem

Cholesky theorem Let $A \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. There exists a lower triangular matrix $B \in \mathbb{R}^{n\times n}$ such that $A = BB^\top$.

Proof.

By induction in next slides. Induction hypothesis: If rows 1 to $j-1$ of B exist, then row j exists. □

First we write

$$A = \begin{bmatrix} b_{11} & 0 & \dots & 0 \\ b_{21} & b_{22} & 0 & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \dots & b_{nn} \end{bmatrix} \begin{bmatrix} b_{11} & b_{21} & \dots & b_{n1} \\ 0 & b_{22} & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & b_{nn} \end{bmatrix}$$

Base case 1st row: From the first column of the above we have

$$a_{:1} = \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{n1} \end{bmatrix} = b_{11} \begin{bmatrix} b_{11} \\ b_{21} \\ \vdots \\ b_{n1} \end{bmatrix} = b_{11} b_{:1}.$$

The first line gives: $b_{11}^2 = a_{11}$ thus $b_{11} = \sqrt{a_{11}}$. This gives the row of B.

First we write

$$A = \begin{bmatrix} b_{11} & 0 & \dots & 0 \\ b_{21} & b_{22} & 0 & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \dots & b_{nn} \end{bmatrix} \begin{bmatrix} b_{11} & b_{21} & \dots & b_{n1} \\ 0 & b_{22} & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & b_{nn} \end{bmatrix}$$

Base case 1st row: From the first column of the above we have

$$a_{:1} = \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{n1} \end{bmatrix} = b_{11} \begin{bmatrix} b_{11} \\ b_{21} \\ \vdots \\ b_{n1} \end{bmatrix} = b_{11} b_{:1}.$$

The first line gives: $b_{11}^2 = a_{11}$ thus $b_{11} = \sqrt{a_{11}}$. This gives the row of B. Now note that $a_{ij} = b_{i:}^\top b_{j:}$.

Let

$$BB^\top = \begin{bmatrix} - & b_{1:}^\top & - \\ - & b_{2:}^\top & - \\ & \vdots & \\ - & b_{n:}^\top & - \end{bmatrix} \begin{bmatrix} | & | & \dots & | \\ b_{1:} & b_{2:} & \dots & b_{n:} \\ | & | & \dots & | \end{bmatrix}$$

Induction: Suppose we know the rows 1 to $j-1$ of B. Thus we know $b_{1:}$ to $b_{j-1:}$. To calculate $b_{j:}$ we use that $a_{ij} = \langle b_{i:}, b_{j:} \rangle$ thus

$$a_{:j} = \sum_{i=1}^{n} \langle b_{j:}, b_{i:} \rangle \, e_i = \sum_{i=1}^{n} \sum_{k=1}^{\min\{j,i\}} b_{jk} b_{ik} e_i = \sum_{k=1}^{\min\{j,i\}} b_{jk} b_{:k}.$$

Isolating $b_{j:}$ gives

$$b_{jj} b_{:j} = a_{:j} - \sum_{k=1}^{j-1} b_{jk} b_{:k} \overset{\text{def}}{=} v.$$

Using $b_{jj} b_{:j} = v$ we have that $b_{jj} = \sqrt{v_j} = \sqrt{a_{jj} - \sum_{k=1}^{j-1} b_{jk}^2}$.
Therefore

$$b_{:j} = \frac{v}{\sqrt{v_j}} = \frac{a_{:j} - \sum_{k=1}^{j-1} b_{jk} b_{:k}}{\sqrt{b_{jj}}}.$$

This completes the induction and provides the following algorithm

Algorithm 2 $(B) =$Cholesky Decomposition(A)

1: **for** $j = 1, \ldots, n$ **do**
2: Calculate $v = a_{:j} - \sum_{k=1}^{j-1} b_{jk} b_{:k}$
3: Set $b_{:j} = v / \sqrt{v_j}$

Exercise

Show that the number of flops of the Cholesky algorithm is proportional to $O(n^3)$.

Solution: The summation in computing v in

$$v = a_{:j} - \sum_{k=1}^{j-1} b_{jk} b_{:k}$$

is where most of the effort goes. Since there are k elements in $b_{:k}$ it costs k to add on $b_{jk} b_{:k}$.

$$\begin{aligned}
\sum_{j=1}^{n} \sum_{k=1}^{j-1} k &= \sum_{j=1}^{n} \frac{(j-1)j}{2} \\
&\leq \sum_{j=1}^{n} \frac{j^2}{2} \leq \frac{1}{2} \int_{x=0}^{n} x^2 dx \\
&= \left. \frac{x^3}{6} \right|_n - \left. \frac{x^3}{6} \right|_0 = \frac{n^3}{6}.
\end{aligned}$$

Using the Cholesky decomposition, we can uncover many properties of positive definite matrices.

Theorem

Let A be a positive definite matrix. It follows that

1. *The Cholesky decomposition $B^\top B = A$ always exists. We can prove this by construction. That is, using induction we can show that Algorithm 2 works. This boils down to showing that $v_j \neq 0$ does not occur.*
2. $det(A) = (b_1 \cdots b_n)^2$. *Indeed, using properties of the determinant we have that*

$$\begin{aligned} det(A) &= det(B^\top B) = det(B^\top)det(B) \\ &= det(B)^2 = (b_{11} \cdots b_{nn})^2. \end{aligned}$$

Eigenvalues are important

Watch the collapse of Tacoma Narrows Bridge as it resonates in the wind. This resonance is related to the smallest eigenvalue of the structural equations:
https://www.youtube.com/watch?v=XggxeuFDaDU
We say that $x \neq 0 \in \mathbb{R}^n$ is an eigenvector with associated eigenvalue $\lambda \in \mathbb{R}$ of A if

$$Ax = \lambda x \;\Leftrightarrow\; (A - \lambda I)x = 0.$$

Since $x \neq 0$ shows that $A - \lambda I$ is not invertible and consequently

$$\det(A - \lambda I) = 0. \tag{10}$$

Compute all eigenvalues by finding roots of this n dim polynomial.

Theorem (Abel–Ruffini theorem)

There is no exact algebraic formula for the roots of a polynomial with degree 5 or more.

Definition (Eigenpairs and Spectrum)

Let $A \in \mathbb{R}^{n\times n}$, $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{C}$. We say that x is an eigenvector and λ an eigenvalue of A if $x \neq 0$ and

$$Ax = \lambda x.$$

We also refer to (x, λ) as an eigenpair of A. We say $\lambda(A) \subset \mathbb{C}$ is the spectrum of A if $\lambda(A)$ contains all the eigenvalues of A, that is

$$\lambda(A) \stackrel{\text{def}}{=} \{\lambda \mid \exists x \in \mathbb{R}^n \text{ such that } x \neq 0,\ Ax = \lambda x\}.$$

We say that A is invertible if $0 \notin \lambda(A)$.

Exercise

If $A = \operatorname{diag}(a_1, \ldots, a_n)$ then

$$\lambda(A) = \{a_1, \ldots, a_n\}.$$

Exercise

If $O \in \mathbb{R}^{n \times n}$ is an orthogonal matrix then every $\lambda \in \lambda(O)$ is such that $|\lambda| = 1$.

Exercise

If $A = \mathrm{diag}\,(a_1, \ldots, a_n)$ then

$$\lambda(A) = \{a_1, \ldots, a_n\}.$$

Exercise

If $O \in \mathbb{R}^{n\times n}$ is an orthogonal matrix then every $\lambda \in \lambda(O)$ is such that $|\lambda| = 1$.

Proof.

Let (x, λ) be such that $Ox = \lambda x$. If follows that

$$\langle x, x\rangle = \left\langle x, O^\top Ox \right\rangle = \langle Ox, Ox\rangle = \|Ox\|_2^2 = |\lambda|^2 \langle x, x\rangle .$$

Dividing by $\langle x, x\rangle$ on both sides gives the result. □

Maybe we should transform A into diagonal or orthogonal?

Definition (Similarity transform)

We say that $A \in \mathbb{R}^{n\times n}$ is similar to $B \in \mathbb{R}^{n\times n}$ if there exists $P \in \mathbb{R}^{n\times n}$ invertible such that

$$A = P^{-1}BP.$$

We say that A is diagonalizable when B is a diagonal matrix.

Lemma

If $A, B \in \mathbb{R}^{n\times n}$ are similar matrices then $\lambda(A) = \lambda(B)$.

Definition (Similarity transform)

We say that $A \in \mathbb{R}^{n\times n}$ is similar to $B \in \mathbb{R}^{n\times n}$ if there exists $P \in \mathbb{R}^{n\times n}$ invertible such that

$$A = P^{-1}BP.$$

We say that A is diagonalizable when B is a diagonal matrix.

Lemma

If $A, B \in \mathbb{R}^{n\times n}$ are similar matrices then $\lambda(A) = \lambda(B)$.

Proof: Consider $\lambda \in \lambda(A)$. Then there exists $x \in \mathbb{R}^n$ such that $Ax = \lambda x$. By the similarity of A and B we have that $P^{-1}BPx = \lambda x$. Left multiplying by P shows that $\lambda \in \lambda(B)$ with associated eigenvector Px. $\square$

Can we transform A into diagonal or orthogonal?

Theorem (Spectral Theorem for symmetric matrices)

Symmetric matrices are diagonalizable. That is, let $A \in \mathbb{R}^{n \times n}$ *with* $A = A^\top$*. Then there exists an orthogonal matrix* $V \in \mathbb{R}^{n \times n}$ *and* $\Lambda = diag(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n \times n}$ *such that*

$$A = V \Lambda V^\top.$$

Proof: See Theorem 8.1.1 and proof in *Matrix Computations*, Golub & Van Loan 2013.

Theorem (Singular Value Decomposition)

Let $A \in \mathbb{R}^{m\times n}$. There exists orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that

$$U^\top AV \;=\; \Sigma = diag(\sigma_1, \ldots, \sigma_p), \quad where\ p = \min\{n,\, m\},$$

and where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p \geq 0$.

Proof: Not so easy. See Theorem 2.4.1 in the book *Matrix Computations*, Golub & Van Loan 2013.
Common notation:

$$\sigma_{\max}(A) \;=\; \sigma_1 \;=\; \max_{i=1,\ldots,p} \sigma_i.$$

$$\sigma_{\min}(A) \;=\; \sigma_p \;=\; \min_{i=1,\ldots,p} \sigma_i.$$

Let $A \in \mathbb{R}^{n \times n}$ be a symmetric matrix.

Spectral Theorem $\Rightarrow$ there exists $V \in \mathbb{R}^{n \times n}$ and diagonal matrix $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ such that

$$A = V \Lambda V^\top \quad \Rightarrow \quad V^\top A V = \Lambda.$$

Idea: Transform A into diagonal matrix using similarity transforms. This gives the Jacobi method.

Notation: $I_d \in \mathbb{R}^{d \times d}$ is the $d \times d$ identity matrix. Thus

$$I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Jacobi Method

Main idea: Iteratively minimize *off–diagonal* elements.
Offset: The sum of the squares of te off-diagonals element:

$$\mathrm{off}(A) = \sum_{i=1}^{n} \sum_{j \neq i} a_{ij}^2 = \|A\|_F^2 - \sum_{i=1}^{n} a_{ii}^2. \tag{11}$$

Iteration:

1. Find largest off diagonal element

$$a_{pq} = \max_{1 \leq i < j \leq n} |a_{ij}|.$$

2. Replace a_{pq} by a zero by using similarity transformations.
3. Use the Givens/Jacobi Transform for this.

Givens/Jacobi Transform

$$J(p,q,\theta) = \begin{array}{cc} \begin{array}{ccccccc} & & p & & q & & \end{array} & \\ \begin{bmatrix} 1 & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \dots & c & \dots & s & \dots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \dots & -s & \dots & c & \dots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 0 & \dots & 1 \end{bmatrix} & \begin{array}{c} \\ \\ p \\ \\ q \\ \\ \\ \end{array} \end{array}$$

Where $c = \cos(\theta)$ and $s = \sin(\theta)$.

Givens/Jacobi Transform

$$
J(p,q,\theta) = \begin{array}{cc} \begin{array}{ccccccc} & & p & & q & & \end{array} & \\ \begin{bmatrix} 1 & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \dots & c & \dots & s & \dots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \dots & -s & \dots & c & \dots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 0 & \dots & 1 \end{bmatrix} & \begin{array}{c} \\ \\ p \\ \\ q \\ \\ \\ \end{array} \end{array}
$$

Where $c = \cos(\theta)$ and $s = \sin(\theta)$. Outer product version:

$$
\begin{aligned}
J(p,q,\theta) &= I_n + (c-1)e_p e_p^\top + (c-1)e_q e_q^\top + s e_p e_q^\top - s e_q e_p^\top \\
&= I_n - \begin{bmatrix} e_p & e_q \end{bmatrix} I_2 \begin{bmatrix} e_p^\top \\ e_q^\top \end{bmatrix} + \begin{bmatrix} e_p & e_q \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} e_p^\top \\ e_q^\top \end{bmatrix}
\end{aligned}
$$

Jacobia Similar Transform

Carefully choosing θ and appling Jacobi similar transform

$$B = J(p, q, \theta) A J(p, q, \theta)^\top, \tag{12}$$

eliminates a_{pq} (and a_{qp} because of symmetry).
Exercise: Show that B is a similar matrix to A.

Jacobia Similar Transform

Carefully choosing θ and appling Jacobi similar transform

$$B = J(p, q, \theta) A J(p, q, \theta)^\top, \tag{12}$$

eliminates a_{pq} (and a_{qp} because of symmetry).
Exercise: Show that B is a similar matrix to A.
Proof: Show that $J(p, q, \theta)$ is an orthogonal matrix. Indeed, let
$J = I_n - \begin{bmatrix} e_p & e_q \end{bmatrix} I_2 \begin{bmatrix} e_p^\top \\ e_q^\top \end{bmatrix} + \begin{bmatrix} e_p & e_q \end{bmatrix} O \begin{bmatrix} e_p^\top \\ e_q^\top \end{bmatrix}$ where $O = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}$.
Part I: First show that

$$(O)^\top O = \begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix} = \begin{bmatrix} c^2 + s^2 & 0 \\ 0 & c^2 + s^2 \end{bmatrix} = I_2.$$

Thus O is an orthogonal matrix.

Part II: Let $\overline{M} \stackrel{\text{def}}{=} \begin{bmatrix} e_p & e_q \end{bmatrix} M \begin{bmatrix} e_p^\top \\ e_q^\top \end{bmatrix}$ for every $M \in \mathbb{R}^{2\times 2}$.

This notation gives

$$J = I - \overline{I}_2 + \overline{O}.$$

Part I gives that $\overline{O}^\top \overline{O} = \overline{I}_2$.

Part II: Let $\overline{M} \stackrel{\text{def}}{=} \begin{bmatrix} e_p & e_q \end{bmatrix} M \begin{bmatrix} e_p^\top \\ e_q^\top \end{bmatrix}$ for every $M \in \mathbb{R}^{2\times 2}$.

This notation gives

$$J = I - \overline{I}_2 + \overline{O}.$$

Part I gives that $\overline{O}^\top \overline{O} = \overline{I}_2$. Consequently

$$\begin{aligned} J^\top J &= (I - \overline{I}_2 + \overline{O}^\top)(I - \overline{I}_2 + \overline{O}) \\ &= I - \overline{I}_2 + \overline{I}_2 + (I - \overline{I}_2)\overline{O} + \overline{O}^\top (I - \overline{I}_2). \\ &= I + (I - \overline{I}_2)\overline{O} + \overline{O}^\top (I - \overline{I}_2) \end{aligned}$$

Now note that

$$(I - \overline{I}_2)\overline{O + I_2} \;=\; 0 \;=\; (\overline{O + I_2})^\top (I - \overline{I}_2)$$

because of disjoint support.

Choosing θ

$$B = J(p,q,\theta)AJ(p,q,\theta)^\top, \tag{13}$$

The pth and qth row and column of B gives

$$\begin{bmatrix} b_{pp} & b_{pq} \\ b_{qp} & b_{qq} \end{bmatrix} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}^\top \begin{bmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix}. \tag{14}$$

Choosing θ

$$B = J(p,q,\theta)AJ(p,q,\theta)^\top, \tag{13}$$

The pth and qth row and column of B gives

$$\begin{bmatrix} b_{pp} & b_{pq} \\ b_{qp} & b_{qq} \end{bmatrix} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}^\top \begin{bmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix}. \tag{14}$$

Equation (16) gives diagonal terms

$$b_{pp} = \begin{bmatrix} ca_{pp} + sa_{qp} & ca_{pq} + sa_{qq} \end{bmatrix} \begin{bmatrix} c \\ -s \end{bmatrix} = c^2 a_{pp} - s^2 a_{qq}.$$

$$b_{qq} = \begin{bmatrix} -sa_{pp} + ca_{qp} & -sa_{pq} + ca_{qq} \end{bmatrix} \begin{bmatrix} s \\ c \end{bmatrix} = c^2 a_{qq} - s^2 a_{pp}$$

$$b_{pp} + b_{qq} = (s^2 - 1)a_{pp} - s^2 a_{qq} + (s^2 - 1)a_{qq} - s^2 a_{pp} = a_{pp} + a_{qq}$$

Choosing θ

$$B = J(p,q,\theta)AJ(p,q,\theta)^\top, \tag{15}$$

The pth and qth row and column of B gives

$$\begin{bmatrix} b_{pp} & b_{pq} \\ b_{qp} & b_{qq} \end{bmatrix} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}^\top \begin{bmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix}. \tag{16}$$

Equation (16) gives off-diagonal terms

$$b_{pq} = cs(a_{pp} - a_{qq}) + (c^2 - s^2)a_{pq}.$$

Choose θ so that $b_{pq} = 0$. Set to zero, divide through by $c^2 a_{pq}$:

$$-t^2 + 2Kt + 1 = 0, \tag{17}$$

where $t = \tan(\theta) = c/s$ and $K = \frac{a_{pp} - a_{qq}}{2a_{pq}}$.

Choosing θ

$$B = J(p, q, \theta) A J(p, q, \theta)^\top, \tag{15}$$

The pth and qth row and column of B gives

$$\begin{bmatrix} b_{pp} & b_{pq} \\ b_{qp} & b_{qq} \end{bmatrix} = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}^\top \begin{bmatrix} a_{pp} & a_{pq} \\ a_{qp} & a_{qq} \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix}. \tag{16}$$

Equation (16) gives off-diagonal terms

$$b_{pq} = cs(a_{pp} - a_{qq}) + (c^2 - s^2) a_{pq}.$$

Choose θ so that $b_{pq} = 0$. Set to zero, divide through by $c^2 a_{pq}$:

$$-t^2 + 2Kt + 1 = 0, \tag{17}$$

where $t = \tan(\theta) = c/s$ and $K = \frac{a_{pp} - a_{qq}}{2a_{pq}}$. The solutions are

$$t = K \pm \sqrt{K^2 + 1}.$$

In the Jacobi method choose the smallest root

$$t = \min\{K + \sqrt{K^2 + 1}, K - \sqrt{K^2 + 1}\}.$$

Choosing θ

Using

$$t = \min\{K + \sqrt{K^2+1}, K - \sqrt{K^2+1}\},$$

we can then recover c and s using that

$$c = \frac{1}{\sqrt{1+t^2}}, \quad s = ct.$$

This gives us the following method for calculating c and s.

Algorithm: $(c, s) =$ **Calculate Jacobi Transform**(p, q, A)

1: $K = \frac{a_{pp} - a_{qq}}{2a_{pq}}$
2: $t = \min\{K + \sqrt{K^2+1}, K - \sqrt{K^2+1}\}$.
3: $c = \frac{1}{\sqrt{1+t^2}}$
4: $s = ct$

Applying the Jacobi transform iteratively to minimize the off diagonal elements of A gives the Jacobi Method.

Algorithm 3 Jacobi Method(ϵ, A)

Initialize: $k = 0$ and $A^0 = A$.
while off$(A^{k+1}) < \epsilon$ **do**
  Choose (p, q) so that $a_{pq} = \max_{i \neq j} |a_{pq}|$
  $(c, s) =$Calculate Jacobi Transform$((p, q, A^k))$
  $A^{k+1} = J(p, q, \theta)^\top A^k J(p, q, \theta)$.

Now we prove it works!

Lemma

1. *Let*

$$O = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}.$$

   Show that $O^\top O = OO^\top = I$*, that is,* O *is an orthogonal matrix.*

2. *Prove that* $Tr(AB) = Tr(BA)$ *for compatible matrices.*
3. *Let* $\|A\|_F^2 = Tr\left(A^\top A\right)$ *and let* J *be an orthogonal matrix. Prove that* $\|J^\top AJ\|_F^2 = \|A\|_F^2$.
4. *Consider* (12) *and show that* $b_{ii} = a_{ii}$ *for* $i = \{1, \ldots, n\} \setminus \{p, q\}$.
5. *Show that* $J(p, q, \theta)$ *is an orthogonal matrix.*

Theorem

Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix. The iterates A^k of the Jacobi method converges to a diagonal matrix at a rate of

$$off(A^k) \leq \left(1 - \frac{2}{n(n-1)}\right)^k off(A).$$

Proof I: Given that $J \equiv J(p, q, \theta)$ is an orthogonal matrix, for $B = J^\top AJ$ we have that

$$\|A\|_F^2 = \|B\|_F^2.$$

Applying the Frobenius norm to both sides gives

$$a_{pp}^2 + a_{qq}^2 + 2a_{pq}^2 = b_{pp}^2 + b_{qq}^2 + 2b_{pq}^2 = b_{pp}^2 + b_{qq}^2. \tag{18}$$

Proof II: Since $b_{pq} = 0$ we have that

$$
\begin{aligned}
\mathrm{off}(B) &= \|B\|_F^2 - \sum_{i=1}^{n} b_{ii}^2 \\
&= \|A\|_F^2 - \sum_{i=1, i\neq p,q}^{n} b_{ii}^2 - b_{pp}^2 - b_{qq}^2 \\
&= \|A\|_F^2 - \sum_{i=1, i\neq p,q}^{n} a_{ii}^2 - b_{pp}^2 - b_{qq}^2 \\
&= \|A\|_F^2 - \sum_{i=1}^{n} a_{ii}^2 + a_{pp}^2 + a_{qq}^2 - b_{pp}^2 - b_{qq}^2 \\
&\overset{(18)}{=} \mathrm{off}(A) - 2a_{pq}^2.
\end{aligned}
$$

$\Rightarrow$ The off diagonal terms are decreasing.

Proof III:

$$\mathrm{off}(B) = \mathrm{off}(A) - 2a_{pq}^2.$$

Since a_{pq} is the largest it is bigger than the average

$$a_{pq}^2 \;\geq\; \frac{\sum_{i\neq j} a_{ij}^2}{n(n-1)} \;=\; \frac{\mathrm{off}(A)}{n(n-1)}.$$

Thus finally

$$\mathrm{off}(B) \leq \mathrm{off}(A) - \frac{2}{n(n-1)}\mathrm{off}(A) = \left(1 - \frac{2}{n(n-1)}\right)\mathrm{off}(A).$$

That is, applying k steps of Algorithm 3 we have that

$$\mathrm{off}(A^k) \leq \left(1 - \frac{2}{n(n-1)}\right)^k \mathrm{off}(A). \quad \square$$

G.,R & P Richtárik, Randomized Iterative Methods for Linear Systems arXiv:1506.03296