2.5 Iterative Methods

We remember that Gaussian elimination is a finite sequence of $O(n^3)$ floating point operations yielding an exact solution. Such methods are known as **direct methods** for the reason that they render (directly) an exact solution. Note that direct methods stand in contrast to the **root-finding** methods seen in chapter 1, which are **iterative** in form. Systems of linear equations can also be solved iteratively – similar to the FPI method of §1.2. These methods start with an initial guess & converge to a solution vector after subsequent iterations.

Jacobi Method

The Jacobi Method is basically a matrix variant of FPI. Recall that with FPI we reformulate the given equation, solving for the unknown variable. We apply this technique with our first example; we solve the ith equation for the ith unknown & then iterate, as in FPI.

Ex. Use the Jacobi Method to solve: $3u+v=5$, $u+2v=5$

solving: $u = \frac{5-v}{3}$, $v = \frac{5-u}{2}$ Let $(u_0, v_0) = (0,0)$

Jacobi

$$\begin{bmatrix} u_0 \\ v_0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

① $\rightarrow$ $\begin{bmatrix} u_1 \\ v_1 \end{bmatrix} = \begin{bmatrix} \frac{5-v_0}{3} \\ \frac{5-u_0}{2} \end{bmatrix} = \begin{bmatrix} \frac{5}{3} \\ \frac{5}{2} \end{bmatrix}$

② $\rightarrow$ $\begin{bmatrix} u_2 \\ v_2 \end{bmatrix} = \begin{bmatrix} \frac{5-v_1}{3} \\ \frac{5-u_1}{2} \end{bmatrix} = \begin{bmatrix} \frac{5}{6} \\ \frac{5}{3} \end{bmatrix}$ ③ $\rightarrow$ $\begin{bmatrix} u_3 \\ v_3 \end{bmatrix} = \begin{bmatrix} \frac{10}{9} \\ \frac{25}{12} \end{bmatrix} \cdots$

Further steps show: $\lim_{n\to\infty} (u_n, v_n) = \boxed{\langle 1, 2 \rangle}$, the solution.

Ex. Does Jacobi always work? Nope, let's see.

Solve the 2×2 using Jacobi: $u+2v=5$, $3u+v=5$

$\rightarrow$ $\boxed{u = 5-2v, \quad v = 5-3u}$ Let $(u_0, v_0) = (0,0)$

① $\rightarrow$ $\begin{bmatrix} u_1 \\ v_1 \end{bmatrix} = \begin{bmatrix} 5-2v_0 \\ 5-3u_0 \end{bmatrix} = \begin{bmatrix} 5 \\ 5 \end{bmatrix}$

② $\rightarrow$ $\begin{bmatrix} u_2 \\ v_2 \end{bmatrix} = \begin{bmatrix} 5-2v_1 \\ 5-3u_1 \end{bmatrix} = \begin{bmatrix} -5 \\ -10 \end{bmatrix}$ ③ $\rightarrow$ $\begin{bmatrix} u_3 \\ v_3 \end{bmatrix} = \begin{bmatrix} 25 \\ 20 \end{bmatrix} \cdots$

Further steps show: $\lim_{n\to\infty} (u_n, v_n)$ diverges!

This leads us to the following definition...

Def. An $n \times n$ matrix A is called **Strictly Diagonally Dominant** (S.D.D.) if for each $1 \le i \le n$, $\boxed{|a_{ii}| > \sum_{j \neq i} |a_{ij}|}$

Put another way, in absolute value, the main diagonal entry "dominates" its row in the sense that it is larger than the sum of all the other entries in that row.

Returning the previous (2) examples then,

$A = \begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}$ → which is SDD (example 1)

&

$A = \begin{bmatrix} 1 & 2 \\ 3 & 1 \end{bmatrix}$ → which is NOT SDD (example 2)

These crucial observations give us the following key Theorem:

Thm (Convergence of the Jacobi Method)
If A, $n \times n$ & A is SDD, then the following hold:

(1) A is non-singular! (i.e. A is invertible)

(2) $\forall \vec{b} \in \mathbb{R}^n$, the Jacobi Method converges to the unique solution of the system: $A\vec{x} = \vec{b}$, for any choice of $\vec{x}_0 \in \mathbb{R}^n$.

In light of this theorem we now see why the Jacobi Method converged in the case of example 1, but not for example 2.

Please note that when A is not SDD the Jacobi Method can still, possibly, converge - such a convergence is, however, not guaranteed in this case. Paraphrasing, we can say SDD is a sufficient condition for convergence, but not necessary.

Ex. Determine whether the matrices given are SDD.

$$A = \begin{bmatrix} 3 & 1 & -1 \\ 2 & -5 & 2 \\ 1 & 6 & 8 \end{bmatrix} \longrightarrow \begin{array}{l} |a_{11}| = 3 > 1+1 \checkmark \\ |a_{22}| = 5 > 2+2 \checkmark \\ |a_{33}| = 8 > 1+6 \checkmark \end{array} \quad \text{Yes, } A \text{ is SDD.}$$

$$B = \begin{bmatrix} 3 & 2 & 6 \\ 1 & 8 & 1 \\ 9 & 2 & -2 \end{bmatrix} \longrightarrow |a_{11}| = 3 \overset{\text{No!}}{\not>} 2+6 \quad \text{No, } B \text{ is } \underline{\text{Not}} \text{ SDD.}$$

We now derive a formula for the Jacobi Method in Matrix form. Remember that the Jacobi method is, at heart, a variant of FPI. We let $\boxed{D}$ denote the diagonal matrix consisting of the diagonal entries of A. Let $\boxed{L}$ be the matrix of lower-triangular elements of A and set $\boxed{U}$ equal to A's upper-triangular elements. (Caution: Here L, U are defined differently than we've seen with LU factorization).

We see then that: $A = L + D + U$ holds.

The idea now is to solve the matrix equation $A\vec{x}=\vec{b}$ as in FPI.

$$A\vec{x}=\vec{b} \quad \text{(we want to solve for } \vec{x}\text{)}$$

$$\rightarrow (D+L+U)\vec{x}=\vec{b} \quad (A=D+L+U)$$

$$\rightarrow D\vec{x}=\vec{b}-(L+U)\vec{x} \quad \text{(solve for } D\vec{x}\text{)}$$

$$\rightarrow \boxed{\vec{x}=D^{-1}(\vec{b}-(L+U)\vec{x})}$$

key formula ☺

(observe that D is invertible since A is SDD)

Summary of Jacobi Method Algorithm

Set: $\vec{x}_0$ = initial vector

Update: $\vec{x}_{k+1}=D^{-1}(\vec{b}-(L+U)\vec{x}_k)$ for $k=0,1,2,\ldots$

Ex. Returning to our first example, we use Jacobi in matrix form.

$$\begin{bmatrix} 3 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix}=\begin{bmatrix} 5 \\ 5 \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} u_{k+1} \\ v_{k+1} \end{bmatrix}=D^{-1}(\vec{b}-(L+U)\vec{x}_k)$$

$$=\begin{bmatrix} 1/3 & 0 \\ 0 & 1/2 \end{bmatrix}\left(\begin{bmatrix} 5 \\ 5 \end{bmatrix}-\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} u_k \\ v_k \end{bmatrix}\right)$$

$$=\begin{bmatrix} \frac{5-v_k}{3} \\ \frac{5-u_k}{2} \end{bmatrix}$$

→ which coheres with our original, iterative solution! ✓

Gauss-Seidel Method & SOR

Like the Jacobi Method, the Gauss-Seidel method relies on a "FPI-like" approach, with one key difference: the most recently updated values of the unknowns are used @ each step.

The Gauss-Seidel Method often converges more rapidly than the Jacobi Method (when it does converge). Our previous theorem similarly applies to G-S, showing that this method also converges if A is SPD.

G-S can be written in matrix form & identified as FPI where we isolate $\vec{x}$ in the equation: $(L+D+U)\vec{x} = \vec{b}$

$$\longrightarrow (L+D)\vec{x}_{k+1} = -U\vec{x}_k + \vec{b}$$

here we used updated $\vec{x}_{k+1}$.

G-S Method

$\vec{x}_0$ = initial vector

$$\vec{x}_{k+1} = D^{-1}(\vec{b} - U\vec{x}_k - L\vec{x}_{k+1}) \quad \text{for } k = 0, 1, 2, \ldots$$

note

Ex. Apply the G-S Method to solve.

$$\begin{bmatrix} 3 & 1 & -1 \\ 2 & 4 & 1 \\ -1 & 2 & 5 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \\ 1 \end{bmatrix}$$

G-S iteration:

$$u_{k+1} = \frac{4 - v_k + w_k}{3}, \quad v_{k+1} = \frac{1 - 2u_{k+1} - w_k}{4}, \quad w_{k+1} = \frac{1 + u_{k+1} - 2v_{k+1}}{5}$$

Note: u_{k+1} in v_{k+1}; Note: u_{k+1} and v_{k+1} in w_{k+1}

It is helpful to see the $(L+D)\vec{x}_{k+1} + U\vec{x}_k$ matrix:

$$\begin{bmatrix} u_k & v_k & w_k \\ u_{k+1} & v_k & w_k \\ u_{k+1} & v_{k+1} & w_{k+1} \end{bmatrix}$$

Setting $\vec{x}_0 = \langle 0,0,0 \rangle$, we have:

(1) $\rightarrow$ $\begin{bmatrix} u_1 \\ v_1 \\ w_1 \end{bmatrix} = \begin{bmatrix} \frac{4-0-0}{3} = \frac{4}{3} \\ \frac{1-8/3-0}{4} = \frac{-5}{12} \\ \frac{1+4/3+5/6}{5} = \frac{19}{30} \end{bmatrix} \approx \begin{bmatrix} 1.\overline{3} \\ -.4167 \\ 0.6\overline{3} \end{bmatrix}$ (2) $\rightarrow$ $\begin{bmatrix} u_2 \\ v_2 \\ w_2 \end{bmatrix} = \begin{bmatrix} 1.68\overline{3} \\ -.750 \\ .8367 \end{bmatrix} \cdots$

Because A is SPD here, the iteration converges to the unique solution, $\vec{x} = \langle 2, -1, 1 \rangle$.

) Note that an additional, common variant of the G-S method called **SOR** (Successive-Over-Relaxation) uses an additional parameter: ω (the relaxation parameter), with $\omega > 1$, in attempt to speed up the convergence of the G-S method.

Why do these methods work? Remember that the Jacobi & G-S methods are FPI (in Matrix Form), & from Ch.1 results, we know FPI converges when $|g'(r)|<1$, etc. We seek a similar condition here.

Proof (Sketch) of Jacobi & G-S Methods:

Def.: The **spectral radius** of a square matrix (say w/ entries in $\mathbb{C}$), denoted, $\rho(A)$ is defined as:

$$\rho(A) = \text{Max}\{|\lambda_1|, \ldots, |\lambda_k|\}$$

, the maximum, in abs. value, of the eigenvalues of the matrix.

It is beyond the scope of this class to show this directly, but it can be shown (see Appendix A) that it is sufficient to show $\rho(D^{-1}(L+U))<1$ in order to guarantee convergence of the Jacobi/G-S methods.

(Why this quantity? Recall that: $\vec{x}_{k+1} = -D^{-1}(L+U)\vec{x}_k + D^{-1}\vec{b}$ — direction converged change)

One more matter of bookkeeping, we can show that the above-noted spectral radius condition is equivalent to showing

$\rho\left((L+D)^{-1}U\right)<1$, we use this more convenient form.

Thm If $A_{n\times n}$ is SDD Then (1) A is non-singular &
(2) $\forall \vec{b}\in\mathbb{R}^n$, $\vec{x}_0\in\mathbb{R}^n$, G-S applied to $A\vec{x}=\vec{b}$ converges.

Relying upon the previous comments, we need to show:

~~ADD~~ A (S.D.D.) $\rightarrow \rho((L+D)^{-1}U)<1$.

Let λ be an eigenvalue of $(L+D)^{-1}U$, with corresponding eigenvector $\vec{v}$. Normalize $\vec{v}$ so that $v_m=1$ (and all other components: $v_i\le 1$). Note that the entries of L are a_{ij} for $i>j$, & the entries of U are a_{ij} for $i<j$. Then, viewing row m of the eigenvalue equation for matrix $(L+D)^{-1}U$ gives:

$$\lambda(D+L)\vec{v}=U\vec{v}$$

(since: ~~$(L+D)^{-1}\vec{v}\rightarrow$~~ $((L+D)^{-1}U)\vec{v}=\lambda\vec{v}$)

This gives us the following string of inequalities:

$$\text{fixed } m \rightarrow |\lambda|\left(\sum_{i>m}|a_{mi}|\right) < |\lambda|\left(|a_{mm}|-\sum_{i<m}|a_{mi}|\right) \quad (\text{since } A \text{ is SDD})$$

$$\rightarrow \le |\lambda|\left(|a_{mm}|-\left|\sum_{i<m}a_{mi}v_i\right|\right) \quad (\text{since } v_i\le 1)$$

$$\rightarrow \le |\lambda|\left|a_{mm}+\sum_{i<m}a_{mi}v_i\right| \quad (|a-b|\le|a|+|b|)$$

$$\rightarrow = \left|\sum_{i>m}a_{mi}v_i\right| \le \sum_{i>m}|a_{mi}|. \quad \text{This shows } |\lambda|<1!$$

$(=\bar{U}\vec{v})$

In conclusion, A (S.D.D.) $\Rightarrow$ G-S / Jacobi converge!

Last comments: Why use iterative methods to solve $A\vec{x}=\vec{b}$ in lieu of an exact method like LU factorization?

There are two general reasons/settings to prefer iterative methods.

(1) Recall that LU/Gaussian requires $\mathcal{O}(n^3)$ operations, whereas, it can be shown, Jacobi & G-S require only $\mathcal{O}(n^2)$, per step. If we have a "good" initial guess $\vec{x}^{(0)}$, then perhaps, only a few iterations of G-S will yield a nice approximate solution (This is sometimes called polishing). Also, for dynamical systems, with small initial changes in a given configuration of a large system, iterative methods might be preferable.

(2) If a matrix is **sparse** (i.e. contains many zeros), note that Gaussian Elimination may cause an unwanted **fill-in** where the coefficient matrix changes from sparse to full. In general, iterative methods will not be susceptible to these sources of slow-down.

Bonus | § 2.5 #5

Thm

Gerŝgorin Circle Theorem

Let A $n \times n$ matrix, R_i denotes the circle in the complex plane with center a_{ii} & radius $\sum_{j=1, j\neq i}^{n} |a_{ij}|$,

$$R_i = \left\{ z \in \mathbb{C} \;\middle|\; |z - a_{ii}| \leq \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| \right\}$$

The eigenvals of A are contained within $R = \bigcup_{i=1}^{n} R_i$.

Moreover, the union of any k of these circles that do not intersect the remaining $(n-k)$ contains precisely k (counting multiplicities) of the eigenvals.

Pf Suppose λ an eigenval for A w/ eigen vector $\vec{x}$, where $\|\vec{x}\|_\infty = 1$. Since $A\vec{x} = \lambda\vec{x}$, equivalently, we have:

$$\sum_{j=1}^{n} a_{ij} x_j = \lambda x_i \qquad i = 1, 2, \ldots, n \quad \text{(each row)}$$

If k is an integer with $|x_k| = \|\vec{x}\|_\infty = 1$, this eqtn with $i = k$, implies:

$$\sum_{j=1}^{n} a_{kj} x_j = \lambda x_k.$$

Thus,

$$\sum_{\substack{j=1 \\ j\neq k}}^{n} a_{kj} x_j = \lambda x_k - a_{kk} x_k = (\lambda - a_{kk}) x_k,$$

and:

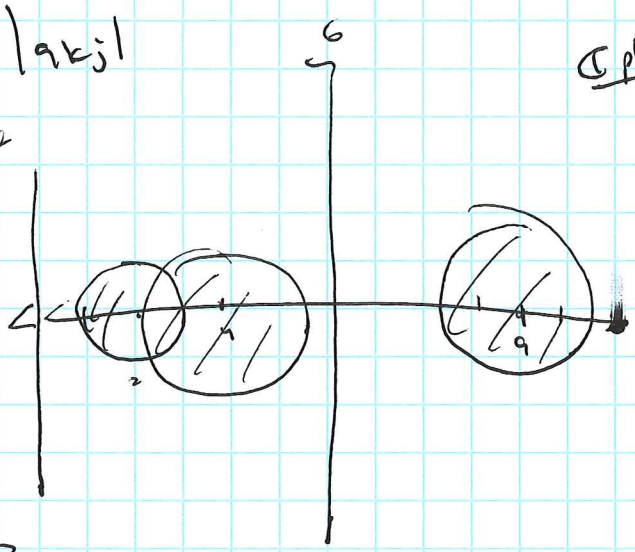
$$|\lambda - a_{kk}| \cdot |x_k| = \left| \sum_{\substack{j=1 \\ j\neq k}}^{n} a_{kj} x_j \right| \leq \sum_{\substack{j=1 \\ j\neq k}}^{n} |a_{kj}||x_j|$$

Since $|x_j| \leq |x_k| = 1$, for all $j = 1, 2, \ldots, n$.

$$|\lambda - a_{kk}| \leq \sum_{\substack{j=1 \\ j\neq k}}^{n} |a_{kj}|$$

Thus $\lambda \in R_k$.

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



Then: $R_1 = \{ z \in \mathbb{C} \mid |z-4| \leq 2 \}$ & $R_3 = \{ z \in \mathbb{C} \mid |z-9| \leq 2 \}$.

$R_2 = \{ z \in \mathbb{C} \mid |z-2| \leq 1 \}$

Since R_1 & R_2 are disjoint from R_3, there are two eigenvalues within $R_1 \cup R_2$ & one within R_3.

Also: for $\rho(A)$ $\rightarrow$ $\boxed{7 \leq \rho(A) \leq 11}$