Finding Eigenvectors

- Once the eigenvalues are obtained, use

$$[A-\lambda I]\{x\}=\{0\}$$

  to solve for $\{x\}$
- Note that a unique solution does not exist
- It will only give the direction of the vector
- Example: $$A=\begin{bmatrix}3 & 1\\ 1 & 3\end{bmatrix}$$ Eigenvalues: 2 and 4
  - For 2, $x_1+x_2=0$ => Eigenvector is $\{1,-1\}^T$
  - For 4, $-x_1+x_2=0$ => Eigenvector is $\{1,1\}^T$

Finding Eigenvectors: Multiple eigenvalues

- What if an eigenvalue is repeated? Known as the algebraic multiplicity. E.g., in the QR method, eigenvalue "1" had an algebraic multiplicity of 2.

$$A = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \qquad \text{Eigenvalues}: 4, 1, 1$$

- For this value, we get $x_1+x_2+x_3=0$
- Eigenvectors may be taken as $\{1,-1,0\}^T$ and $\{1,0,-1\}^T$. Two linearly independent eigenvectors for the same eigenvalue: called geometric multiplicity of "2".

Finding Eigenvectors: Multiple eigenvalues

- Another example:

$$A = \begin{bmatrix} 2 & 1 \\ 0 & 2 \end{bmatrix}$$

- Eigenvalue "2" has algebraic multiplicity of 2
- For this value, we get $x_2=0$
- A single eigenvector $\{1,0\}^T$: geometric multiplicity is 1.
- Geometric multiplicity is always $\leq$ Alg. Mult
- A defective matrix has GM<AM for some λ (called defective eigenvalue): It will not have n linearly independednt eigenvalues.

Finding eigenvalues for given eigenvectors

- Straightforward: All components are multiplied by the factor λ. Ratio of the L_1, L_2, or L_∞ norm of Ax and x could be used.

- $Ax=\lambda x \Rightarrow x^T Ax= \lambda x^T x$

- Therefore: $\lambda= x^T Ax/(x^T x)$

- Known as Rayleigh's quotient

Iterative methods for linear equations

- What are the conditions of convergence for the iterative methods?
- Rate of convergence? Can we make them converge faster?

Iterative Methods

$$\underset{\boldsymbol{A}}{\begin{bmatrix} a_{11} & a_{12} & \cdot & \cdot & a_{1n} \\ a_{21} & a_{22} & \cdot & \cdot & a_{2n} \\ a_{31} & a_{32} & \cdot & \cdot & a_{3n} \\ & \cdot\ \cdot & \cdot & \cdot\ \cdot & \\ a_{n1} & a_{n2} & \cdot & \cdot & a_{nn} \end{bmatrix}} = \underset{\boldsymbol{L}}{\begin{bmatrix} 0 & 0\cdots & \cdot & \cdot & \cdots\cdot 0 \\ a_{21} & 0 & 0 & \cdot & \cdots 0 \\ a_{31} & a_{32} & 0 & 0 & 0 \\ & \cdot\ \cdot & \cdot & \cdot\ \cdot & \\ a_{n1} & a_{n2} & a_{n3} & \cdot & 0 \end{bmatrix}} +$$

$$\underset{\boldsymbol{D}}{\begin{bmatrix} a_{11} & 0\cdots & 0 & \cdot & 0 \\ 0 & a_{22} & 0 & 0 & \cdot 0 \\ 0 & 0 & a_{33} & 0 & \cdot 0 \\ & \cdot\ \cdot & \cdot & \cdot\ \cdot & \\ 0 & 0 & 0 & \cdot & \cdot\cdot a_{nn} \end{bmatrix}} + \underset{\boldsymbol{U}}{\begin{bmatrix} 0 & a_{12} & a_{13} & \cdot & a_{1n} \\ 0 & 0 & a_{23} & \cdot & a_{2n} \\ 0 & 0 & 0 & \cdot & a_{3n} \\ & \cdot\ \cdot & \cdot & \cdot\ \cdot & \\ 0 & 0 & \cdot & \cdot & 0 \end{bmatrix}}$$

Iterative Methods

- $\boldsymbol{A = L + D + U}$
- $\boldsymbol{Ax = b}$ translates to $\boldsymbol{(L + D + U)x = b}$
- **Jacobi:** for an iteration counter k

$$Dx^{(k+1)} = -(U + L)x^{(k)} + b$$

$$x^{(k+1)} = -D^{-1}(U + L)x^{(k)} + D^{-1}b$$

- **Gauss Seidel:** for an iteration counter k

$$(L + D)x^{(k+1)} = -Ux^{(k)} + b$$

$$x^{(k+1)} = -(L + D)^{-1}Ux^{(k)} + (L + D)^{-1}b$$

Iterative Methods: Convergence

- All iterative methods: $x^{(k+1)} = Sx^{(k)} + c$
- *Jacobi*: $S = -D^{-1}(U+L) \qquad c = D^{-1}b$
- *Gauss Seidel*: $S = -(L+D)^{-1}U \qquad c = (L+D)^{-1}b$
- For true solution vector (ξ): $\xi = S\,\xi + c$
- True error: $e^{(k)} = \xi - x^{(k)}$
- $e^{(k+1)} = Se^{(k)}$ or $e^{(k)} = S^k e^{(0)}$
- Methods will converge if:

$$\lim_{k \to \infty} e^{(k)} = 0; \quad \text{i.e.,} \quad \lim_{k \to \infty} S^k = 0$$

Iterative Methods: Convergence

- For the solution to exist, the matrix should have full rank (= n)
- The iteration matrix S will have n eigenvalues $\{\lambda_j\}_{j=1}^{n}$ and n independent eigenvectors $\{v_j\}_{j=1}^{n}$
- Initial error vector: $e^{(0)} = \sum_{j=1}^{n} C_j v_j$
- From the definition of eigenvalues: $e^{(k)} = \sum_{j=1}^{n} C_j \lambda_j^k v_j$
- Necessary condition: $\rho(S) < 1$
- Sufficient condition: $\|S\| < 1$ because $\rho(A) \le \|A\|$. Why?

$$Ax = \lambda x \Rightarrow \lambda\|x\| = \|Ax\| \Rightarrow \lambda\|x\| \le \|A\|.\|x\| \Rightarrow \lambda \le \|A\|$$

Jacobi Convergence

$$\boldsymbol{S} = -\boldsymbol{D}^{-1}(\boldsymbol{L}+\boldsymbol{U}) \qquad s_{ij} = \begin{cases} -\dfrac{a_{ij}}{a_{ii}} & \text{for } i \neq j \\ 0 & \text{for } i = j \end{cases}$$

If we use infinity (row-sum) norm:

$$\|S\|_\infty = \max_{1 \le i \le n} \sum_{j=1}^{n} |s_{ij}| = \max_{1 \le i \le n} \sum_{j=1,\, j \ne i}^{n} \left| \frac{a_{ij}}{a_{ii}} \right|$$

$$|a_{ii}| > \sum_{j=1,\, j\neq i}^{n} |a_{ij}| \,, \quad i = 1,2,\cdots n$$

Iterative Methods: Convergence

Using the **definition of *S*** and using ***row-sum norm*** for matrix S, we obtain the following as the **sufficient condition for convergence** for Jacobi

$$|a_{ii}| > \sum_{j=1,\, j\neq i}^{n} |a_{ij}|, \quad i = 1,2,\cdots n$$

If the original matrix is diagonally dominant, Jacobi method will always converge!

(Gauss Seidel convergence is a little harder to prove, but diagonal dominance is sufficient for that also)

Rate of Convergence

For large k:

$$\frac{\left|e^{(k+1)}\right|}{\left|e^{(k)}\right|} \cong \rho(S) \qquad \text{or} \qquad \frac{\left|e^{(k)}\right|}{\left|e^{(0)}\right|} \cong \left[\rho(S)\right]^k$$

Rate of Convergence

Number of iteration (k) required to decrease the initial error by a factor of 10^{-m} is then given by:

$$\frac{\left|e^{(k)}\right|}{\left|e^{(0)}\right|} \cong \left[\rho(S)\right]^k = 10^{-m}$$

or

$$k \geq -\frac{m}{\log_{10} \rho(S)}$$

Improving Convergence

Denoting: $\rho(S) = |\lambda|_{\max}$

$$e^{(k+1)} \cong |\lambda|_{\max} e^{(k)} \ \text{ or } \ e^{(k+1)} - e^{(k)} \cong |\lambda|_{\max}(e^{(k)} - e^{(k-1)})$$

For any iterative method: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \boldsymbol{d}^{(k)}$

$$d^{(k)} \cong \lambda_{\max} d^{(k-1)}$$

Improving Convergence

Recall Gauss Seidel:

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}}{a_{ii}}, \quad i = 1,\ 2,\ \cdots\ n$$

Rewrite as:

$$x_i^{(k+1)} = x_i^{(k)} + \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)}}{a_{ii}}, \quad i = 1,\ \ 2,\ \ \cdots\ n$$

$$x_i^{(k+1)} = x_i^{(k)} + d_i^{(k)}, \quad i = 1,\ \ 2,\ \ \cdots\ n$$

Successive Over/Under Relaxation

$$x_i^{(k+1)} = x_i^{(k)} + \omega d_i^{(k)}, \quad i = 1, \; 2, \; \cdots \; n, \quad \omega > 0$$

$0 < \omega < 1$: Under relaxation
$\omega = 1$: Gauss Seidel
$1 < \omega < 2$: Over Relaxation

Non-convergent if ω is outside the range (0,2)

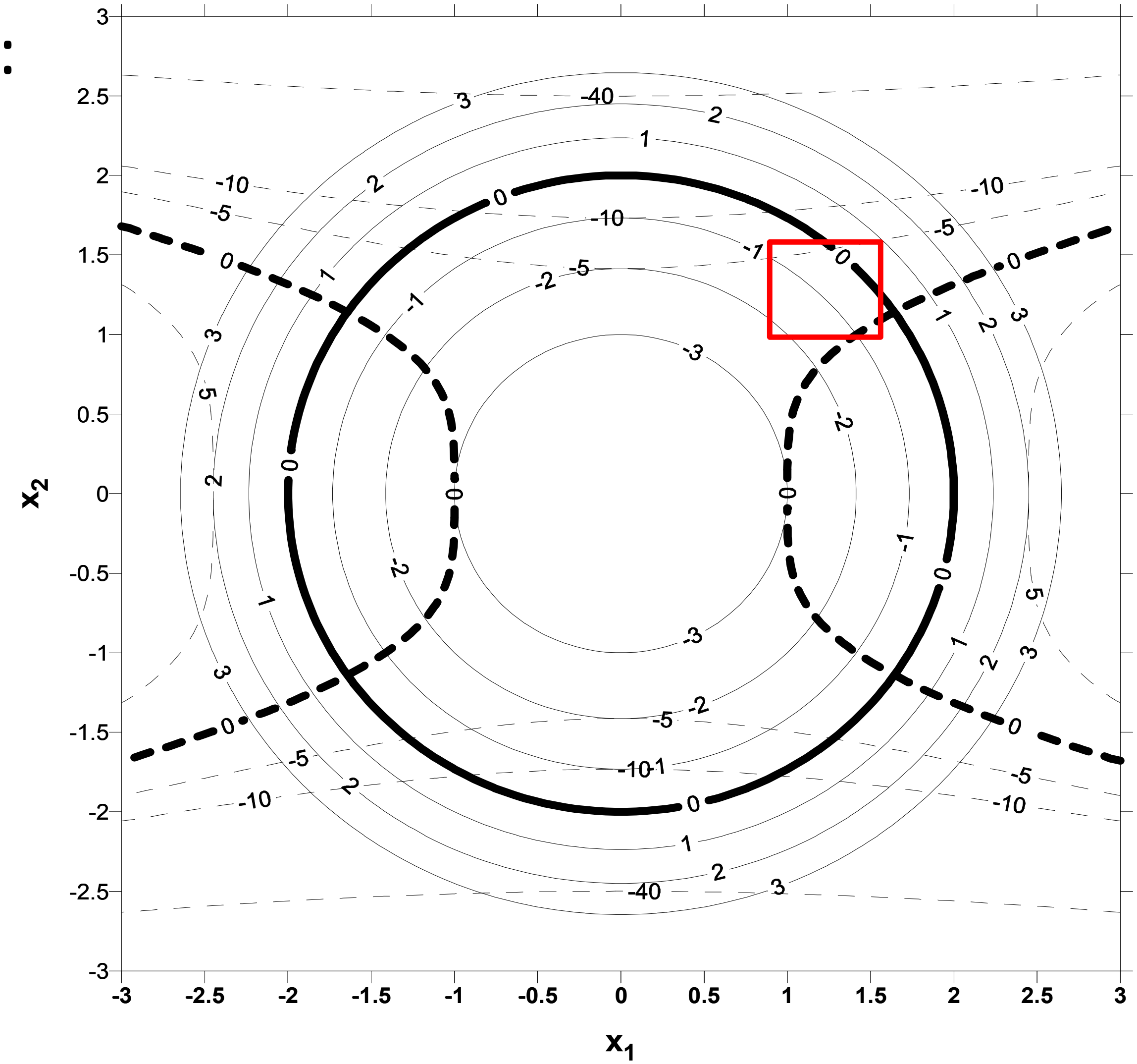
$$x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \omega \frac{b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}}{a_{ii}}, \quad i = 1, \; 2, \; \cdots \; n$$

System of nonlinear equations

- Example: $x_1^2 + x_2^2 = 2^2 \quad \Rightarrow f_1(x_1, x_2) = x_1^2 + x_2^2 - 4 = 0$

  $x_1^2 - x_2^4 = 1 \quad \Rightarrow f_2(x_1, x_2) = x_1^2 - x_2^4 - 1 = 0$

- Plot the functions:
- Bracketing does not work



Solution:

$\pm 1.64, \pm 1.14$

System of nonlinear equations: Fixed Point

- Given *n* equations,

$$f_1(x_1, x_2, \ldots, x_n) = 0; f_2(x_1, x_2, \ldots, x_n) = 0; \ldots; f_n(x_1, x_2, \ldots, x_n) = 0$$

- Similar to the single equation method, we write:

$$x_1 = \phi_1(x_1, x_2, \ldots, x_n)$$

$$x_2 = \phi_2(x_1, x_2, \ldots, x_n)$$

$$.$$

$$.$$

$$x_n = \phi_n(x_1, x_2, \ldots, x_n)$$

- Iterations: $\{x^{(i+1)}\} = \{\phi(x^{(i)})\}$

  in which {x} and {ϕ} are vectors

Fixed Point Method: Example

$$x^x + y^y = 11.72;\ x^y + y^x = 6.71$$

$$x = \left(6.71 - y^x\right)^{1/y} \Rightarrow \phi_1(x, y) = \left(6.71 - y^x\right)^{1/y}$$
$$y = \left(11.72 - x^x\right)^{1/y} \Rightarrow \phi_2(x, y) = \left(11.72 - x^x\right)^{1/y}$$

- Iterations: $$x^{(i+1)} = \left(6.71 - y^x\right)^{1/y}\Big|_{(x^{(i)}, y^{(i)})}$$
  $$y^{(i+1)} = \left(11.72 - x^x\right)^{1/y}\Big|_{(x^{(i+1)}, y^{(i)})}$$

- We could use a Jacobi style scheme, but Seidel is preferred – update values as these are being computed
- Computations, starting with *x*=*y*=2

i	x	y	$\Phi_1(x,y)$	$\Phi_2(x,y)$	E1a(%)	E2a(%)	max(E1a,E2a)
0	2	2	1.646208	3.073785			
1	1.646208	3.073785	0.716856	2.177338	-21.4913	34.93364	34.933643
2	0.716856	2.177338	2.087117	2.456267	-129.643	-41.1717	129.6428016
3	2.087117	2.456267	0.503584	2.655634	65.65331	11.35583	65.65330821
4	0.503584	2.655634	1.843432	2.251655	-314.453	7.507304	314.452503
5	1.843432	2.251655	1.432151	2.786309	72.68224	-17.9414	72.68223984
10	1.739954	2.445766	1.319329	2.592682	34.87487	-0.94789	34.87487135
15	1.605152	2.520952	1.391191	2.506223	12.05857	3.04661	12.05857347
20	1.521199	2.525046	1.46397	2.486363	1.01676	2.296381	2.296380772
25	1.492373	2.512254	1.497092	2.489679	-1.89271	0.927234	1.892713526
31	1.505144	2.49584	1.49943	2.504048	0.953095	-0.29328	0.953095181

$$\phi_1(x,y) = \left(6.71 - y^x\right)^{1/y}; \phi_2(x,y) = \left(11.72 - x^x\right)^{1/y}$$

Fixed Point Method: Convergence

$$e^{(i+1)} = \xi - x^{(i+1)} = \phi(\xi) - \phi(x^{(i)})$$

- ξ is also a vector

$$e_1^{(i+1)} = \left.\frac{\partial \phi_1}{\partial x_1}\right|_{\hat{x}_1} e_1^{(i)} + \left.\frac{\partial \phi_1}{\partial x_2}\right|_{\hat{x}_2} e_2^{(i)} + \ldots + \left.\frac{\partial \phi_1}{\partial x_n}\right|_{\hat{x}_n} e_n^{(i)}$$

- If $\left|\frac{\partial \phi_j}{\partial x_1}\right| + \left|\frac{\partial \phi_j}{\partial x_2}\right| + \ldots \left|\frac{\partial \phi_j}{\partial x_n}\right| < 1 \quad \forall j \text{ from } 1 \text{ to } n$

convergence is guaranteed

System of nonlinear equations: Newton method

- Given *n* equations,

$$f_1(x_1, x_2, \ldots, x_n) = 0; f_2(x_1, x_2, \ldots, x_n) = 0; \ldots; f_n(x_1, x_2, \ldots, x_n) = 0$$

- Similar to the single equation method, we write:

$$f_1\left(x^{(k+1)}\right) \approx f_1\left(x^{(k)}\right) + \sum_{j=1}^{n} \left.\frac{\partial f_1}{\partial x_j}\right|_{x^{(k)}} \left(x_j^{(k+1)} - x_j^{(k)}\right) = 0$$

- Iterations: $J\left(x^{(k)}\right)\Delta x^{(k+1)} = -f\left(x^{(k)}\right)$

J is called the Jacobian matrix, given by $J_{i,j} = \dfrac{\partial f_i(x)}{\partial x_j}$

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k+1)} = x^{(k)} - J^{-1}\left(x^{(k)}\right) f\left(x^{(k)}\right)$$

(If J is easily invertible, else solve linear system JΔx=-f)

Newton Method: Example

$$x^x + y^y = 11.72; x^y + y^x = 6.71$$

$$f_1(x,y) = x^x + y^y - 11.72 = 0; f_2(x,y) = x^y + y^x - 6.71 = 0$$

$$f_{1x}^{'} = x^x(1+\ln x); f_{1y}^{'} = y^y(1+\ln y); f_{2x}^{'} = yx^{y-1} + y^x \ln y; f_{2y}^{'} = x^y \ln x + xy^{x-1}$$

- Iterations:

$$\begin{bmatrix} x^x(1+\ln x) & y^y(1+\ln y) \\ yx^{y-1} + y^x \ln y & x^y \ln x + xy^{x-1} \end{bmatrix}_{(x^{(k)},y^{(k)})} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}^{(k+1)} = -\begin{Bmatrix} x^x + y^y - 11.72 \\ x^y + y^x - 6.71 \end{Bmatrix}_{(x^{(k)},y^{(k)})}$$

- Computations, starting with *x*=1, *y*=2
- (Will not converge, if we start with 2,2. Fixed-point would not have converged if we started with 1,2!)

$$\begin{bmatrix} x^x(1+\ln x) & y^y(1+\ln y) \\ yx^{y-1}+y^x\ln y & x^y\ln x + xy^{x-1}\end{bmatrix}_{(x^{(k)},y^{(k)})} \begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix}^{(k+1)} = -\begin{Bmatrix} x^x+y^y-11.72 \\ x^y+y^x-6.71\end{Bmatrix}_{(x^{(k)},y^{(k)})}$$

i	x1	x2	$f_1(x_1,x_2)$	$f_2(x_1,x_2)$	f_{1x1}\`	f_{1x2}\`	f_{2x1}\`	f_{2x2}\`	$J(x_1,x_2)$		$J^{-1}(x_1,x_2)$		Δx
0	1.0000	2.0000	-6.7200	-3.7100	1.0000	6.7726	3.3863	1.0000	1.0000	6.7726	-0.0456	0.3088	0.8392
									3.3863	1.0000	0.1544	-0.0456	0.8683
1	1.8392	2.8683	11.8882	5.9762	4.9354	42.1865	16.2721	7.9513	4.9354	42.1865	-0.0123	0.0652	-0.2435
									16.2721	7.9513	0.0251	-0.0076	-0.2533
2	1.5957	2.6150	2.7388	1.3201	3.0928	24.2235	10.0186	4.4150	3.0928	24.2235	-0.0193	0.1058	-0.0868
									10.0186	4.4150	0.0437	-0.0135	-0.1020
3	1.5089	2.5130	0.2726	0.1180	2.6254	19.4694	8.3839	3.5681	2.6254	19.4694	-0.0232	0.1265	-0.0086
									8.3839	3.5681	0.0545	-0.0171	-0.0128
4	**1.5002**	**2.5002**	0.0036	0.0012	2.5832	18.9449	8.2181	3.4910	2.5832	18.9449	-0.0238	0.1292	-0.0001
									8.2181	3.4910	0.0560	-0.0176	-0.0002