Math 315: Numerical Analysis
Emory University
Spring 2024
Prof. Lucas Onisk

Matrix Splitting, Relaxation Methods

Mitchell Scott
(mtscot4)

1 Introduction

So far in class, only direct methods have been used. These methods solve the linear system $\mathbf{Au} = \mathbf{f}$ in a similar way – factor $\mathbf{A}$, then solve simpler problems. Examples are $\mathbf{LU}, \mathbf{QR}, \mathbf{U\Sigma V}^\top$. We know they are direct because as soon as you finish your computation, you have the correct answer.
Today we are going to introduce an "iterative method", which are ways that we can build up a sequence of progressively better approximations to the solution, that eventually converge to the true solution, or a solution that is close enough that we are happy with it.

2 Matrix Splitting

Suppose we have an linear system[1]

$$\mathbf{Au} = \mathbf{f}, \qquad \mathbf{A} \in \mathbb{R}^{n\times n}, \mathbf{f} \in \mathbb{R}^n. \tag{1}$$

We are going to split this matrix into $\mathbf{A} = \mathbf{M} - \mathbf{N}$. If we choose wisely and $\mathbf{M}$ is invertible, we can get a "stationary" iteration to solve the original linear system.

$$\mathbf{Mu}_{k+1} = \mathbf{Nu}_k + \mathbf{f}, \quad k = 0, 1, 2, ... \tag{2}$$

To choose a good matrix splitting, we need to pick an $\mathbf{M}$ such that it is cheap to invert, and it will let the stationary method converge quickly.

Definition 2.1 (Stationary Iteration, Iteration Matrix)**.** The reason this is called a *stationary iteration* is because $\mathbf{A} = \mathbf{M} - \mathbf{N}$, where $\mathbf{M}$ is invertible, stay constant from iteration to iteration. Once we have determined a matrix splitting, we stick to it for the rest of the problem. The *iteration matrix* is $\mathbf{G} := \mathbf{M}^{-1}\mathbf{N}$.

Example 2.1. *An example of a "bad" matrix splitting would be if we let* $\mathbf{M} = \mathbf{A}, \mathbf{N} = \mathbf{0}$. *Then we would have*

$$\begin{aligned} \mathbf{Mu}_{k+1} &= \mathbf{Nu}_k + \mathbf{f} \\ \mathbf{Au}_{k+1} &= \mathbf{0u}_k + \mathbf{f} \\ \mathbf{u}_{k+1} &= \mathbf{A}^{-1}\mathbf{f}. \end{aligned}$$

[1] In numerical PDEs, we often write the linear system as such where u is the unknown solution to the PDE, and f is the function that the PDE equals. This is just a different way to write $\mathbf{A}x = b$.

While this does converge in one iteration, it is essentially the same as solving the linear system. This isn't a good iterative method because $\mathbf{M}$ *was* not *easy to invert.*

Because we only have approximations, we now need to define how close the approximation is to the real solution.

Definition 2.2 (Error, Residual, and Difference)**.** Let $\mathbf{u}$ be the true solution of $\mathbf{Au} = \mathbf{f}$. Additionally, we have $\mathbf{u}_k$ is the solution vector after k iterations. We define $\mathbf{e}_k := \mathbf{u} - \mathbf{u}_k$ as the error between the true solution and the approximate solution after k iterations, the residual $\mathbf{r}_k := \mathbf{f} - \mathbf{Au}_k$, and the difference $\mathbf{d}_k := \mathbf{u}_{k+1} - \mathbf{u}_k$.

Equivalently, we can rewrite the generic matrix splitting using the definition of $\mathbf{r}$.

$$\begin{aligned} \mathbf{Mu}_{k+1} &= \mathbf{Nu}_k + \mathbf{f} \\ \mathbf{u}_{k+1} &= \mathbf{M}^{-1}\mathbf{Nu}_k + \mathbf{M}^{-1}\mathbf{f} \\ &= \mathbf{M}^{-1}\left(\mathbf{M} - \mathbf{A}\right)\mathbf{u}_k + \mathbf{M}^{-1}\mathbf{f} \\ &= \mathbf{u}_k + \mathbf{M}^{-1}\underbrace{\left(\mathbf{f} - \mathbf{Au}_k\right)}_{:=\mathbf{r}_k} \end{aligned}$$

Remark. Pick your favorite vector norm and corresponding induced matrix norm. A sufficient but not necessary condition for the convergence of the matrix splitting $\mathbf{A} = \mathbf{M} - \mathbf{N}$ is if $\|\mathbf{M}^{-1}\mathbf{N}\| < 1$. This is because

$$\begin{aligned} \|\mathbf{e}_{k+1}\| &\leqslant \left\|\mathbf{M}^{-1}\mathbf{N}\right\|\|\mathbf{e}_k\| \\ &\leqslant \left\|\mathbf{M}^{-1}\mathbf{N}\right\|^2\|\mathbf{e}_{k-1}\| \\ &\leqslant \left\|\mathbf{M}^{-1}\mathbf{N}\right\|^{k+1}\|\mathbf{e}_0\| \end{aligned}$$

Therefore if $\|\mathbf{M}^{-1}\mathbf{N}\| < 1 \implies \lim_{k\to\infty} \|\mathbf{M}^{-1}\mathbf{N}\|^k \searrow 0$. However, this is not a necessary condition. Consider a matrix $\mathbf{X}$ with all zeros except a 10 in the upper right corner. This has $\|\cdot\| > 1$ but $\mathbf{X}^2$ is the zero matrix.

Theorem 2.2 (Golub, Van Loan)**.** *Let* $\mathbf{A} \in \mathbb{R}^{n\times n}$ *be invertible with* $\mathbf{A} = \mathbf{M} - \mathbf{N}$*, where* $\mathbf{M}$ *is invertible. Additionally* $\mathbf{f} \in \mathbb{R}^n$*. The stationary method*

$$\mathbf{Mu}_{k+1} = \mathbf{Nu}_k + \mathbf{f}$$

converges for any initial vector $\mathbf{u}_0 \in \mathbb{R}^n$ *to the true solution* $\mathbf{u}$ *of the linear system* $\mathbf{Au} = \mathbf{f}$ *if and only if* $\rho(\mathbf{M}^{-1}\mathbf{N}) < 1$*.*

Proof sketch. Assume for the sake of contradiction $|\lambda_{\max}| = \rho(\mathbf{M}^{-1}\mathbf{N}) \geqslant 1$. Let $\mathbf{u}_0$ be an initial guess such that $\mathbf{e}_0 = \mathbf{u} - \mathbf{u}_0$ is an eigenvector. Now observing the error at the k^{th} step, and using recursion, we see

$$\begin{aligned} \mathbf{e}_{k+1} &= \mathbf{u} - \mathbf{u}_{k+1} \\ &= \mathbf{u} - \left(\mathbf{M}^{-1}\mathbf{Nu}_k + \mathbf{M}^{-1}\mathbf{f}\right) \\ &= \mathbf{M}^{-1}\mathbf{Nu} + \mathbf{M}^{-1}\mathbf{f} - \left(\mathbf{M}^{-1}\mathbf{Nu}_k + \mathbf{M}^{-1}\mathbf{f}\right) \\ &= \mathbf{M}^{-1}\mathbf{N}\left(\mathbf{u} - \mathbf{u}_k\right) \\ &= \mathbf{M}^{-1}\mathbf{Ne}_k \\ &= \left(\mathbf{M}^{-1}\mathbf{N}\right)^{k+1}\mathbf{e}_0 \\ &= |\lambda_{\max}|^{k+1}\,\mathbf{e}_0 \end{aligned}$$

But since $|\lambda_{\max}| > 1$, $\mathbf{e}_{k+1} \to \infty$ even for relatively small $\mathbf{e}_0$. Even if $|\lambda_{\max}| = 1$, we still don't get a reduction in error, so we have no hope of converging, or $\mathbf{e}_k \to 0$, as $k \to \infty$. □

3 Jacobi Iteration

Jacobi (1845) proposed this method, where you approximate a matrix by just looking at the diagonal. Let's look at our approximate solution at k iterations, $\mathbf{u}_k$. Since we don't have the true solution, the error vector isn't enlightening, but the residual vector $\mathbf{r}_k = f - \mathbf{A}\mathbf{u}_k$ we can get. We want to set one of the components of the residual vector to zero.

$$\begin{aligned}\mathbf{r}_k(i) &= \mathbf{f}(i) - \mathbf{A}(i,:)\mathbf{u}_k \\ &= \mathbf{f}(i) - \mathbf{A}(i,1:i-1)\mathbf{u}_k(1:i-1) - \mathbf{A}(i,i+1:n)\mathbf{u}_k(i+1:n) - \mathbf{A}(i,i)\mathbf{u}_k(i) \\ &= 0.\end{aligned}$$

Now solving for $\mathbf{u}_k$, we see the update

$$\mathbf{u}_{k+1}(i) = \frac{1}{\mathbf{A}(i,i)}\left[\mathbf{f}(i) - \mathbf{A}(i,:)\mathbf{u}_k\right] \tag{3}$$

will make sure $r_{k+1}(i) = 0$. This idea can be performed for all $i = 1 : n$ in parallel. In fact, this motivates a specific matrix splitting where for a matrix $\mathbf{A}$, we assign

$$\begin{aligned}\mathbf{A} &= \begin{pmatrix} \ddots & & \mathbf{U} \\ & \mathbf{D} & \\ \mathbf{L} & & \ddots \end{pmatrix} \qquad (4)\\ &= \mathbf{L} + \mathbf{D} + \mathbf{U}, \qquad (5)\end{aligned}$$

where $\mathbf{L}$ is a strict lower triangular matrix, $\mathbf{D}$ is a diagonal matrix, and $\mathbf{U}$ is a strict upper triangular matrix. Using this, we can use our matrix splitting $\mathbf{A} = \mathbf{M} - \mathbf{N}$, where $\mathbf{M} = \mathbf{D}, \mathbf{N} = -\mathbf{L} - \mathbf{U}$ to get

$$\mathbf{u}_{k+1} = \mathbf{D}^{-1}\left[-(\mathbf{L}+\mathbf{U})\mathbf{u}_k + \mathbf{f}\right] \tag{6}$$

One of the requirements of the matrix splitting was to pick an invertible $\mathbf{M}$ that was easy to compute. Did we accomplish our job using this Jacobi Method? Yes, the diagonal matrix is a basic structure that can be inverted in $\mathcal{O}(n)$ operations. Additionally, for a certain class of matrices, approximating it as a diagonal matrix is a reasonable assumption.

Definition 3.1 (Strictly Diagonally Dominant)**.** A matrix $\mathbf{A}$ is said to be *strictly diagonally dominant* if

$$|a_{ii}| > \sum_{j \neq i} |a_{ij}|, \quad \forall i = 1, ..., n \tag{7}$$

Theorem 3.1. *(Convergence of Jacobi Method for diagonally dominant matrices) If the matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is strictly diagonally dominant, then the Jacobi Iteration converges.*

Proof. The iteration matrix for the Jacobi Method is $\mathbf{G}_J := \mathbf{M}^{-1}\mathbf{N} = -\mathbf{D}\left(\mathbf{L}+\mathbf{U}\right)$. We then use diagonal dominance to observe

$$\begin{aligned}\|\mathbf{G}_j\|_\infty &= \max_{i \in \{1:n\}} \frac{1}{|a_{ii}|}\sum_{j \neq i}|a_{ij}| \\ &< 1\end{aligned}$$

Then we see $\rho(\mathbf{G}_J) \leqslant \|\mathbf{G}\|_\infty < 1$, so Jacobi Iteration will converge. □

4 Gauss-Seidel

Gauss (1823) actually theorized this (better) method years before Jacobi. Recall in Jacobi iteration, we can write it as

$$\begin{aligned}\mathbf{u}_{k+1}(i) &= \frac{1}{\mathbf{A}(i,i)}\left[\mathbf{f}(i) - \sum_{j=1,j\neq i}^{n} \mathbf{A}(i,j)\mathbf{u}_k(j)\right] \\ &= \frac{1}{\mathbf{A}(i,i)}\left[\mathbf{f}(i) - \sum_{j=1}^{i-1} \mathbf{A}(i,j)\mathbf{u}_k(j) - \sum_{j=i+1}^{n} \mathbf{A}(i,j)\mathbf{u}_k(j)\right]\end{aligned}$$

But notice that we do this for all i, and we have already have computed $\mathbf{u}_k(i)$ for $i < j$. We could use these updated values in our computation, so we don't have to start using these computed values once we have all of them available.
This method would look like:

$$\mathbf{u}_{k+1}(i) = \frac{1}{\mathbf{A}(i,i)}\left[\mathbf{f}(i) - \sum_{j=1}^{i-1} \mathbf{A}(i,j)\mathbf{u}_{k+1}(j) - \sum_{j=i+1}^{n} \mathbf{A}(i,j)\mathbf{u}_k(j)\right] \tag{8}$$

While we can no longer do these computations on the i^{th} component in parallel, we can rewrite this in terms of matrices to speed up computations.
Can anyone guess what this would look like? Let $\mathbf{M} = \mathbf{D} + \mathbf{L}, \mathbf{N} = -\mathbf{U}$. This implies

$$\mathbf{u}_{k+1} = -\left(\mathbf{D} + \mathbf{L}\right)^{-1}\left[\mathbf{U}\mathbf{u}_k + \mathbf{f}\right] \tag{9}$$

Remark. The above algorithm is called forward Gauss-Seidel. There is also a backwards Gauss-Seidel, which could be written as

$$\mathbf{u}_{k+1} = -\left(\mathbf{D} + \mathbf{U}\right)^{-1}\left[\mathbf{L}\mathbf{u}_k + \mathbf{f}\right].$$

This is accomplished from running from $i = N : -1 : 1$.

Theorem 4.1. *(Convergence of Gauss-Seidel Method on diagonally dominant matrices) If the matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ is strictly diagonally dominant, then the Gauss-Seidel Iteratino converges.*

Proof. Let λ be an eigenvalue of $\mathbf{G}_{GS} = -\left(\mathbf{D} + \mathbf{U}\right)^{-1}\mathbf{L}$ with eigenvector $\|\mathbf{x}\|_\infty = 1$. Then $-\mathbf{U}\mathbf{x} = \lambda\left(\mathbf{D} + \mathbf{L}\right)\mathbf{x}$.

$$-\sum_{j=i+1}^{n} \mathbf{A}(i,j)\mathbf{x}_j = \lambda\mathbf{A}(i,i)\mathbf{x}_i - \lambda\sum_{j=1}^{i-1}\mathbf{A}(i,j)\mathbf{x}_j, \qquad \forall i = 1:n$$

Solving for λ with a specific index i

$$\begin{aligned}
|\lambda| &\leqslant \frac{\sum_{j=i+1}^{n} |\mathbf{A}(i,j)||\mathbf{x}_j|}{|\mathbf{A}(i,i)||x_i| - \sum_{j=1}^{i-1} |\mathbf{A}(i,j)||\mathbf{x}_j|} \\
&= \frac{\sum_{j=i+1}^{n} |\mathbf{A}(i,j)|}{|\mathbf{A}(i,i)| - \sum_{j=1}^{i-1} |\mathbf{A}(i,j)|} \\
&= \frac{\sum_{j=i+1}^{n} |\mathbf{A}(i,j)|}{|\mathbf{A}(i,i)| - \sum_{j=1}^{i-1} |\mathbf{A}(i,j)| + \sum_{j=i+1}^{n} |\mathbf{A}(i,j)| - \sum_{j=i+1}^{n} |\mathbf{A}(i,j)|} \\
&= \frac{\sum_{j=i+1}^{n} |\mathbf{A}(i,j)|}{\sum_{j=i+1}^{n} |\mathbf{A}(i,j)| + |\mathbf{A}(i,i)| - \sum_{j=1}^{i-1} |\mathbf{A}(i,j)| - \sum_{j=i+1}^{n} |\mathbf{A}(i,j)|} \\
&= \frac{\sum_{j=i+1}^{n} |\mathbf{A}(i,j)|}{\sum_{j=i+1}^{n} |\mathbf{A}(i,j)| + |\mathbf{A}(i,i)| - \sum_{j\neq i} |\mathbf{A}(i,j)|} \\
&< 1
\end{aligned}$$

Since λ was an arbitrary eigenvalue, this means this holds for all λ, namely $\rho(\mathbf{G}_{GS}) < 1$, so it will converge. □

Theorem 4.2. *(Convergence of Gauss-Seidel Method on SPD matrices.) If* $\mathbf{A}$ *is SPD, then the Gauss–Seidel iterates converge to the true solution, given any initial guess* $\mathbf{u}_0$.

Proof. □

Example 4.3. *Clearly we are getting a better approximation to* $\mathbf{A}$ *by using GS then Jacobi, but is this always true? Consider the matrix*

$$\mathbf{A} = \begin{pmatrix} -1 & 0 & -1 \\ -1 & 1 & 0 \\ 1 & 2 & -3 \end{pmatrix}$$

Quick check: is it strictly diagonally dominant? SPD? The Jacobi iteration matrix is

$$\mathbf{G}_J = -\mathbf{D}\left(\mathbf{L} + \mathbf{U}\right) = \begin{pmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ \frac{1}{3} & \frac{2}{3} & 0 \end{pmatrix}$$

The spectral radius $\rho(\mathbf{G}_J) \approx 0.944 < 1$*, so it will converge. The GS iteration matrix is*

$$\mathbf{G}_{GS} = -\left(\mathbf{D} + \mathbf{L}\right)^{-1}\mathbf{U} = \begin{pmatrix} 0 & 0 & -1 \\ 0 & 0 & -1 \\ 0 & 0 & -1 \end{pmatrix},$$

which has $\rho(\mathbf{G}_{GS}) = 1$*, so it will not converge.*

Remark. If both GS and Jacobi converge, then GS is faster.

5 Successive Over Relaxiation (SOR)

This idea is a weighted version of GS, with relaxation parameter ω. It computes the GS iterate, but then averages it with weight ω with the last iterate.

$$\mathbf{u}_{k+1}(i) = (1-\omega)\,\mathbf{u}_k(i) + \frac{\omega}{\mathbf{A}(i,i)}\left[\mathbf{f}(i) - \sum_{j=1}^{i-1} \mathbf{A}(i,j)\mathbf{u}_{k+1}(j) - \sum_{j=i+1}^{n} \mathbf{A}(i,j)\mathbf{u}_k(j)\right] \tag{10}$$

Similarly, we can write it in the matrix splitting scheme, where

$$\mathbf{M} = \frac{1}{\omega}\mathbf{D} + L, \mathbf{N} = -\mathbf{U} + \left(\frac{1}{\omega} - 1\right)\mathbf{D},$$

leading to the SOR update

$$\mathbf{u}_{k+1} = (\mathbf{D} + \omega\mathbf{L})^{-1}\big(-\omega\mathbf{U} + (1-\omega)\,\mathbf{D}\big)\mathbf{u}_k + \omega\mathbf{f} \tag{11}$$

Incorporating a weighting parameter ω might help in some cases, but will it always converge?

Theorem 5.1. *(Kahan, 1958) Let* $\mathbf{A} \in \mathbb{R}^{n\times n}$*, then*

$$\rho(\mathbf{G}_{SOR}) \geqslant |\omega - 1|, \qquad \forall\omega \in \mathbb{R}.$$

Proof. We judiciously multiply by the identity bfI.

$$\begin{aligned}\mathbf{G}_{SOR} &= (\mathbf{D} + \omega\mathbf{L})^{-1}\big(-\omega\mathbf{U} + (1-\omega)\,\mathbf{D}\big)\\ &= (\mathbf{D} + \omega\mathbf{L})^{-1}\,\mathbf{D}\mathbf{D}^{-1}\big(-\omega\mathbf{U} + (1-\omega)\,\mathbf{D}\big)\\ &= \big(\mathbf{I} + \omega\mathbf{D}^{-1}\mathbf{L}\big)^{-1}\big(-\omega\mathbf{D}^{-1}\mathbf{U} + (1-\omega)\mathbf{I}\big)\end{aligned}$$

Since $(\mathbf{I} + \omega\mathbf{D}^{-1}\mathbf{L})$ is a unit lower triangular matrix, we know that its determinant equals 1, and so does it's inverse.

$$\begin{aligned}\det(\mathbf{G}_{SOR}) &= \det\big(-\omega\mathbf{D}^{-1}\mathbf{U} + (1-\omega)\mathbf{I}\big)\\ &= (1-\omega)^n\end{aligned}$$

By definition of the determinant, we have

$$\prod_{j=1}^{n}\lambda_j(\mathbf{G}_{SOR}) = (1-\omega)^n$$

but if we substitute λ_j with $\lambda_{\max}$, then clearly,

$$\begin{aligned}|1-\omega|^n &\leqslant \left(\max_j|\lambda_j(\mathbf{G}_{SOR})|\right)^n\\ &= \rho(\mathbf{G}_{SOR})^n\end{aligned}$$

□

Corollary 5.2. *SOR will converge for all* $0 < \omega < 2$.

Proof. Since we need $\rho(\mathbf{G}_{SOR}) < 1$ for SOR to converge, we need $|\omega - 1| < 1 \implies 0 < \omega < 2$. □

6 Summary

For a stationary iteration scheme

$$\mathbf{u}_{k+1} = \mathbf{G}\mathbf{u}_k + \mathbf{f},$$

we get the following methods:

Method	Iteration Matrix
Jacobi	$\mathbf{G}_J = -\mathbf{D}^{-1}(\mathbf{L}+\mathbf{U}) = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A}$
Gauss-Seidel	$\mathbf{G}_{GS} = -(\mathbf{D}+\mathbf{L})^{-1}\mathbf{U} = \mathbf{I} - (\mathbf{D}+\mathbf{L})^{-1}\mathbf{A}$
SOR	$\mathbf{G}_{SOR} = (\mathbf{D}+\omega\mathbf{L})^{-1}\big(-\omega\mathbf{U} + (1-\omega)\,\mathbf{D}\big)$
SSOR	

Table 1