```
julia> n=5
5

julia> Z=rand(n,n).-0.5
5×5 Matrix{Float64}:
 -0.0919428   0.127042    0.333714    0.297024    0.23492
 -0.418973   -0.252085    0.0732716  -0.293201    0.09378
 -0.296016   -0.0947318  -0.398581    0.395704   -0.00208211
  0.0422706  -0.260992    0.0280384   0.447972    0.258772
 -0.183211    0.461886   -0.111315    0.0892178  -0.456926
```

```
julia> mycholesky(Z)
ERROR: input is not a symmetric matrix
```

```
julia> Y=Z+Z'
5×5 Matrix{Float64}:
 -0.183886   -0.29193     0.0376982   0.339294   0.0517092
 -0.29193    -0.50417    -0.0214602  -0.554193   0.555666
  0.0376982  -0.0214602  -0.797162    0.423743  -0.113397
  0.339294   -0.554193    0.423743    0.895944   0.34799
  0.0517092   0.555666   -0.113397    0.34799   -0.913852
```

```
julia> mycholesky(Y)
ERROR: input is not positive definite
```

```
julia> using LinearAlgebra

julia> W=Y+16*I
5×5 Matrix{Float64}:
 15.8161     -0.29193     0.0376982   0.339294   0.0517092
 -0.29193    15.4958     -0.0214602  -0.554193   0.555666
  0.0376982  -0.0214602  15.2028      0.423743  -0.113397
  0.339294   -0.554193    0.423743   16.8959     0.34799
  0.0517092   0.555666   -0.113397    0.34799   15.0861
```

← large positive.

```
julia> mycholesky(W)
5×5 Matrix{Float64}:
  3.97695      0.0          0.0         0.0         0.0
 -0.0734057    3.93579      0.0         0.0         0.0
  0.00947918  -0.00527578   3.89907     0.0         0.0
  0.0853153   -0.139217     0.108282    4.1058      0.0
  0.0130022    0.141425    -0.0289234   0.0900437   3.88034
```

If it is positive def. then Cholesky exists

```
julia> W=Y-16*I
5×5 Matrix{Float64}:
 -16.1839      -0.29193    …    0.339294    0.0517092
  -0.29193    -16.5042          -0.554193    0.555666
   0.0376982   -0.0214602        0.423743   -0.113397
   0.339294    -0.554193       -15.1041     0.34799
   0.0517092    0.555666         0.34799   -16.9139
```

```
julia> mycholesky(W)
ERROR: input is not positive definite
```

```
julia> mycholesky(-W)
5×5 Matrix{Float64}:
  4.02292      0.0          0.0         0.0         0.0
  0.0725668    4.06188      0.0         0.0         0.0
 -0.00937086   0.00545072   4.09842     0.0         0.0
 -0.0843403    0.137944    -0.103768    3.88164     0.0
 -0.0128536   -0.13657      0.0278207  -0.0843322   4.1094
```

```
julia> Wbad=Y+diagm([16,16,16,-16,-16])
5×5 Matrix{Float64}:
 15.8161      -0.29193     0.0376982    0.339294    0.0517092
 -0.29193     15.4958     -0.0214602   -0.554193    0.555666
  0.0376982   -0.0214602  15.2028       0.423743   -0.113397
  0.339294    -0.554193    0.423743   -15.1041     0.34799
  0.0517092    0.555666   -0.113397     0.34799   -16.9139
```

```
julia> cholesky(Wbad)
ERROR: PosDefException: matrix is not positive definite; Cholesky factorization failed.
```

```
julia> cholesky(-Wbad)
ERROR: PosDefException: matrix is not positive definite; Cholesk
y factorization failed.
```

$A = M - N$ ← what's left

↑ easy to invert

Gauss-Seidel.

$M = \text{LowerTriangular}(A)$ $\quad N = M - A$

```
julia> M=LowerTriangular(Wbad)
5×5 LowerTriangular{Float64, Matrix{Float64}}:
 15.8161       .          .          .          .
 -0.29193    15.4958      .          .          .
  0.0376982  -0.0214602  15.2028      .          .
  0.339294   -0.554193    0.423743  -15.1041     .
  0.0517092   0.555666   -0.113397    0.34799  -16.9139

julia> N=M-Wbad
5×5 Matrix{Float64}:
 0.0  0.29193  -0.0376982  -0.339294  -0.0517092
 0.0  0.0       0.0214602   0.554193  -0.555666
 0.0  0.0       0.0        -0.423743   0.113397
 0.0  0.0       0.0         0.0       -0.34799
 0.0  0.0       0.0         0.0        0.0
```

$$Mx^{(k+1)} = Nx^{(k)} + b$$ defines iterations

$$x^{(k+1)} = \underbrace{M^{-1}N}_{T} x^{(k)} + M^{-1}b$$

$$\rho(T) = \rho(M^{-1}N) = \max\{|\lambda| : \lambda \text{ is an eigenvalue of } T\}$$

```
julia> T=M^-1*N
5×5 Matrix{Float64}:
 0.0   0.0184578    -0.00238353   -0.0214524   -0.0032694
 0.0   0.000347732   0.00134       0.0353598   -0.0359207
 0.0  -4.52786e-5    7.80193e-6   -0.0277695    0.00741634
 0.0   0.000400603  -0.000102491  -0.00255838   0.0244921
 0.0   7.63989e-5    3.45746e-5    0.00122962  -0.000735903

julia> eigvals(T)
5-element Vector{Float64}:
 -0.008153399814561105
 -0.0010997572891949016
  0.0
  0.0010664730265540511
  0.005247930957076534
```

all less than 1 in magnitude.

```
julia> rho(T)=maximum(abs,eigvals(T))
rho (generic function with 1 method)

julia> rho(T)
0.008153399814561105
```

```
julia> A=Z'*Z
5×5 Matrix{Float64}:
  0.30697     0.0263234   …  -0.0190106    0.0343782
  0.0263234   0.370116       -0.00154842  -0.272184
  0.0781843   0.00295091     -0.0774533    0.144216
 -0.0190106  -0.00154842      0.53941      0.116613
  0.0343782  -0.272184        0.116613     0.339731
```

Test that it really is positive definite...

```
julia> L=mycholesky(A)
5×5 Matrix{Float64}:
  0.554049    0.0          0.0        0.0        0.0
  0.0475109   0.606514     0.0        0.0        0.0
  0.141114   -0.00618876   0.518485   0.0        0.0
 -0.0343121   0.000134842 -0.140044   0.720153   0.0
  0.062049   -0.453628     0.255846   0.214722   0.13616
```

```
julia> opnorm(L*L'-A)
1.1273073621278 58e-16
```

← Gives back the same matrix...

Try Gauss-Seidel iteration: solve $Ax=b$

$$Mx^{(k+1)} = Nx^{(k)} + b$$

$$x^{(0)} = 0$$

$$x^{(k+1)} = M \backslash (Nx^{(k)} + b)$$

```
julia> b=rand(n).-0.5
5-element Vector{Float64}:
 -0.3486054517753O6
  0.13187311144352 73
 -0.43146139757772983
 -0.024487685600632858
 -0.042816370531560954
```

```
M=LowerTriangular(A); N=M-A
xk=zeros(n)
for k=1:20
    global xk=M\(N*xk+b)
    println("|Ax-b|=",norm(A*xk-b))
end
```

```
julia> for k=1:20
           global xk=M\(N*xk+b)
           println("|Ax-b|=",norm(A*xk-b))
       end
|Ax-b|=0.3218423166986515
|Ax-b|=0.2992902644599371
|Ax-b|=0.2835593039442102
|Ax-b|=0.2686942814873009
|Ax-b|=0.2546107556430864
|Ax-b|=0.24126557667970464
|Ax-b|=0.22861988556357107
|Ax-b|=0.2166370065991067
|Ax-b|=0.20528219814153567
|Ax-b|=0.1945225404321386
```

getting smaller

```
julia> for k=1:20
           global xk=M\(N*xk+b)
           println("|Ax-b|=",norm(A*xk-b))
       end
|Ax-b|=1.2851760658783268e-15
|Ax-b|=1.2851760658783268e-15
|Ax-b|=1.2851760658783268e-15
|Ax-b|=1.2851760658783268e-15
|Ax-b|=1.2851760658783268e-15
|Ax-b|=1.2851760658783268e-15
|Ax-b|=1.2851760658783268e-15
```

can't converge any more because of rounding error...

Over-relaxation: For any splitting $A=M-N$

$$Mx^{(k+1)} = Nx^{(k)} + b$$

Now do this...

$$Mz = Nx^{(k)} + b$$

$$x^{(k+1)} = wz + (1-w)x^{(k)}$$

usually think of this as an interpolation with $w \in (0,1)$

But if $w>1$ this extrapolation.

$$Mz = Nx^{(k)} + b$$

$$x^{(k+1)} = wz + (1-w)x^{(k)}$$

$$Mx^{(k+1)} = wMz + (1-w)Mx^{(k)}$$

$$Mx^{(k+1)} = w\left(Nx^{(k)} + b\right) + (1-w)Mx^{(k)}$$

Solve for $x^{(k+1)}$

$$x^{(k+1)} = wM^{-1}\left(Nx^{(k)} + b\right) + (1-w)x^{(k)}$$

$$= \left(wM^{-1}N + (1-w)I\right)x^{(k)} + wM^{-1}b$$

$$T$$

I want to know $\rho\left(wM^{-1}N + (1-w)I\right)$ and choose w to minimize the spectral radius...

Need the eigenvalues of $T_w = wT + (1-w)I$

Turnout the eigenvectors of T_w are the same as T.

$$Tv = \lambda v$$

λ: eigenvalue of T. v: eigenvector of T

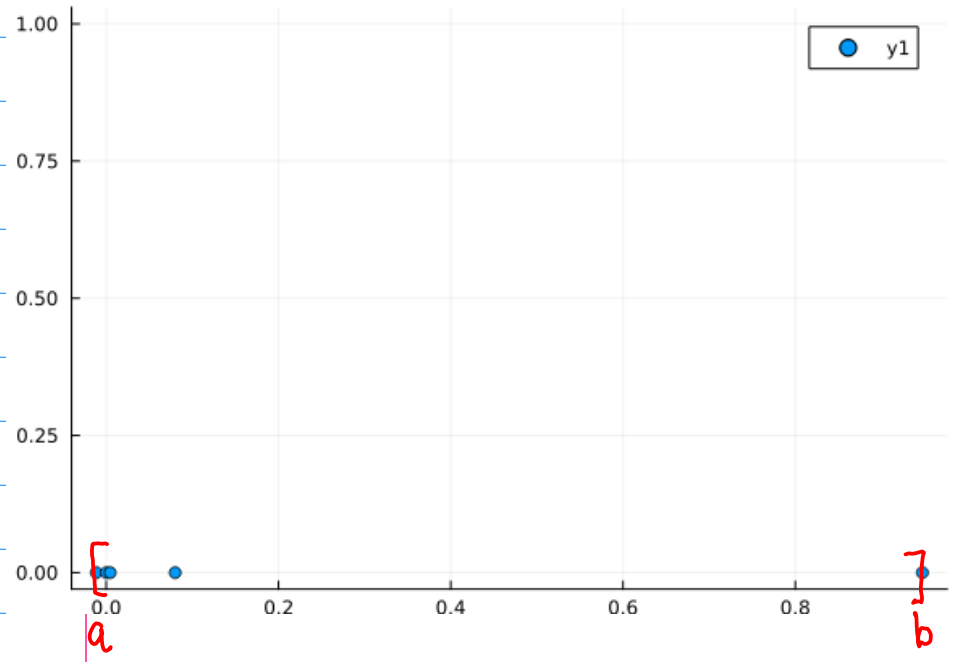
$$T_w v = \left(wT + (1-w)I\right)v = wTv + (1-w)v$$

$$= w\lambda v + (1-w)v = \left[w\lambda + (1-w)\right]v$$

v: also eigenvector of T_w

$w\lambda + (1-w)$: the eigenvalue of T_w

```
julia> T=M^-1*N
5×5 Matrix{Float64}:
 0.0  -0.0857522    -0.254697      0.0619297    -0.111992
 0.0   0.00609886    0.0101416    -0.000220967   0.743365
 0.0   0.0231544     0.0688534     0.251446     -0.476675
 0.0   0.000320031   0.000939349   0.0382868    -0.286445
 0.0   0.00362484    0.00434786   -0.126325      0.907569
```




```
julia> lambdas=eigvals(T)
5-element Vector{Float64}:
 -0.011456622415626O4
  0.0
  0.004615348410780209
  0.08006362609149038
  0.9475860166824607

julia> a=minimum(real,lambdas)
-0.011456622415626O4

julia>ba=maximum(real,lambdas)
0.9475860166824607
```

Changed to ba so doesn't overwrite the b from Ax=b.

minimize $|w\lambda + (1-w)|$

for all w between a, b

$$wa + (1-w) = -(wb + (1-w))$$

$$w(a+b) = -2(1-w) = 2w - 2$$

$$w(a+b-2) = -2$$

$$w(2-a-b) = 2$$

$$w_{opt} = \frac{2}{2-a-b}$$

new variable

```
julia> wopt=2/(2-a-b2)
1.8799278683159986

julia> Tw(w)=w*T+(1-w)*I
Tw (generic function with 1 method)

julia> eigvals(Tw(1))
5-element Vector{Float64}:
 -0.011456622415626O4
  0.0
  0.0046153484107802O9
  0.08006362609149038
  0.9475860166824607

julia> eigvals(Tw(wopt))
5-element Vector{Float64}:
 -0.9014654920719078
 -0.8799278683159986
 -0.8712513462165855
 -0.7294140263881734
  0.901465492071908
```

maximum

maximum is less than before

```
julia> rho(Tw(wopt))
0.901465492071908
```

```
julia> rho(Tw(1))
0.9475860166824607
```

so the over relaxed version converges faster

Test this next time.