MINRES and Lanczos in perturbed symmetric systems

# Overview

Broadly, four types of matrices have been investigated:

* A simple diagonal matrix (6x6 for readability)
* Perturbations of increasing magnitude added to the above diagonal matrix
* A random symmetric matrix
* Perturbations of increasing magnitude added to the above symmetric matrix

Essentially, I began with evaluating how well MINRES performs in its role of solving Ax = b. I tried to be general, so the results are when

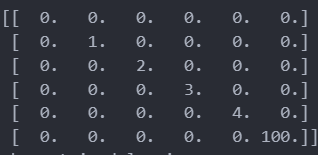
* b is an Arbitrary Vector
* b is an Eigenvector

After these results, I decided to try and have a closer look at the Lanczos algorithm, and how it behaves when increasing perturbations are introduced.

# MINRES

## B is an arbitrary vector

### A is a simple diagonal matrix

A = 

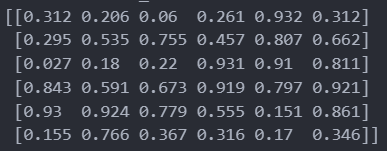
b = 

x = 

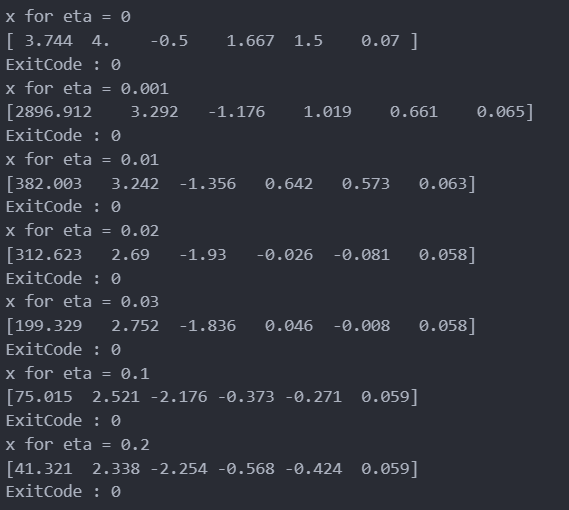
As expected, the algorithm converges normally. These are the baseline results, to compare the enxt section against.

### Perturbations are introduced to the Diagonal A

Perturbations are of the form eta\*del\_A, where

del\_A = 

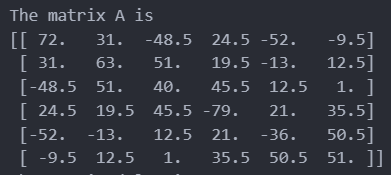
Now, as eta is varied, we’re in for a rude shock.



At absolutely **tiny** perturbations, our x changes dramatically. I’m not sure if this is because of this matrix, but it’s still an interesting observation – this is, after all, an almost diagonal matrix.

The exitcode = 0 refers to the fact that the algorithm converged normally, or at least to the extent possible.

### A is a symmetric matrix (random)

A = 

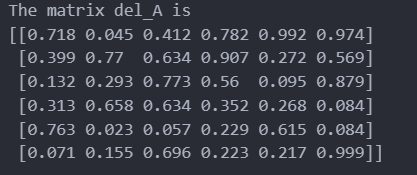
b = 

x = This thing keeps varying every time I refresh my code!

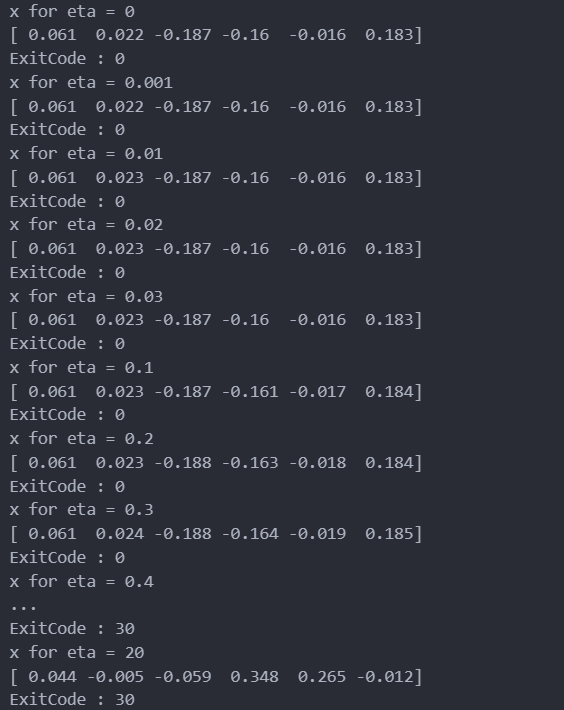


### Perturbations are introduced to A

Perturbations are of the form eta\*del\_A, where

del\_A = 

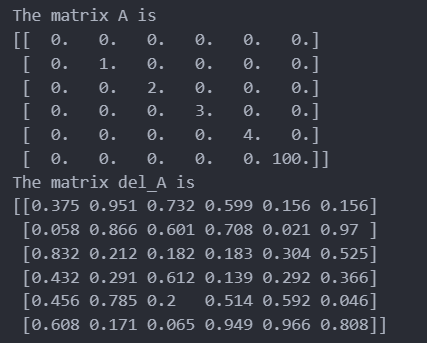
and the resulting x as we vary eta are –



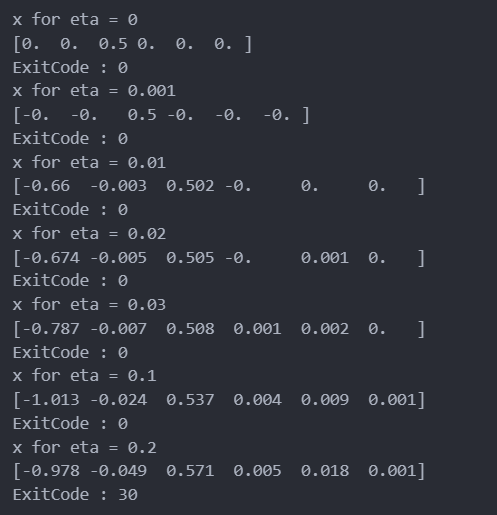
I am not sure what to make of this. Here, the solution is quite unchanged, with convergence only being interrupted at higher values of eta, and even then the solution is pretty good. I do not understand why this is the case, unless it has to do with the underlying distribution of the matrix elements. The matrix A here is populated via a uniform random distribution of integers from -100 to 100, whereas the diagonal matrix was a random set I chose. The “relative order of magnitude”, if this is a thing, seems pretty constant between matrix elements for the full matrix, whereas it is very skewed for the diagonal one (you have a few small numbers and then a MASSIVE diagonal entry) – maybe that does something as well?

## B is an eigenvector

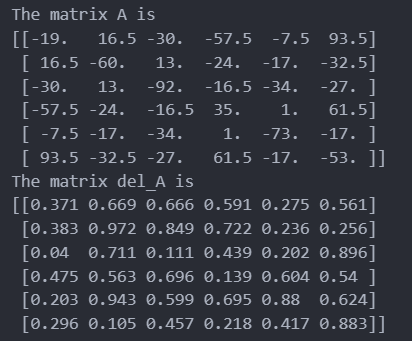
### Diagonal Matrix



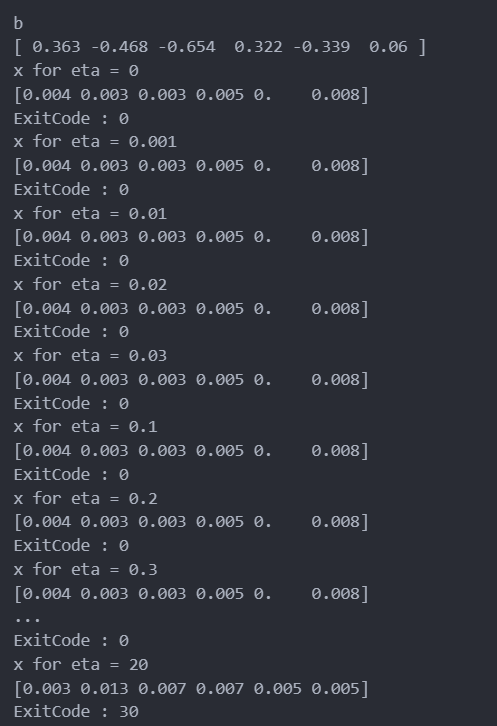




### Proper Symmetric Matrix



b is an eigenvector:

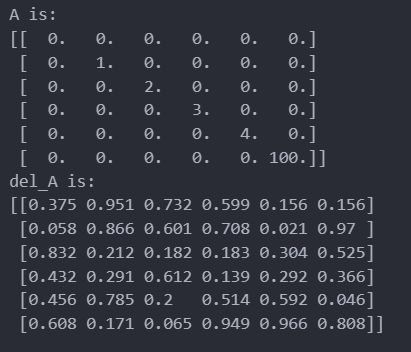


# Lanczos

Here, only 10 iterations were performed, and thus ghost eigenvalues avoided.

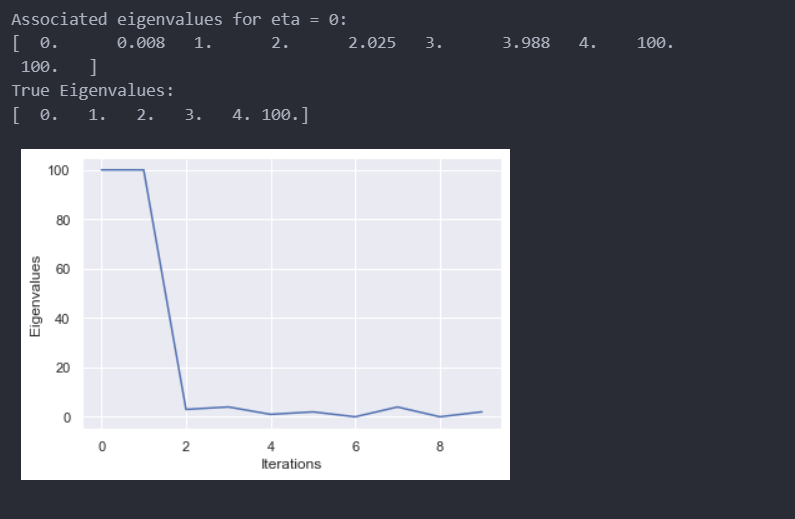
## Diagonal Matrix

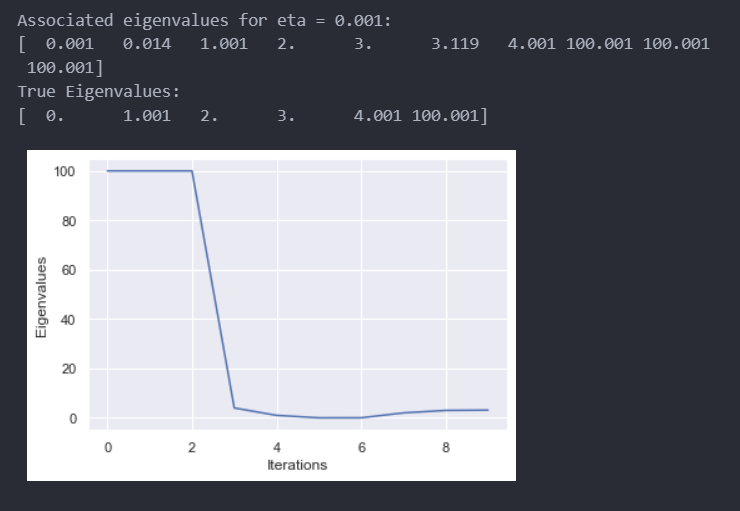
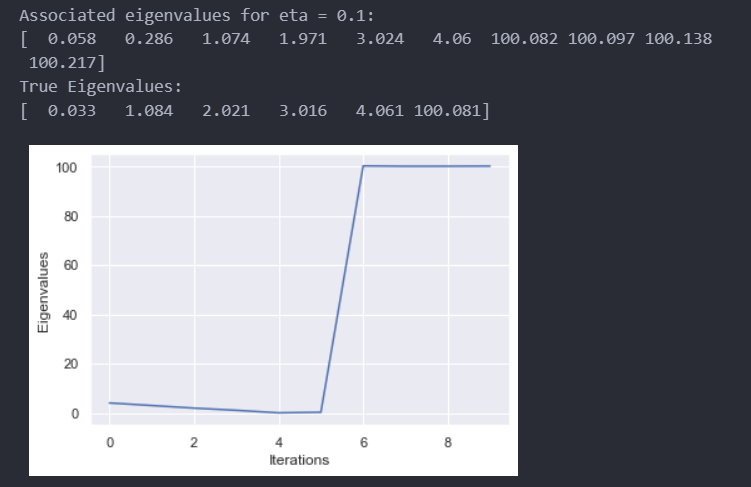
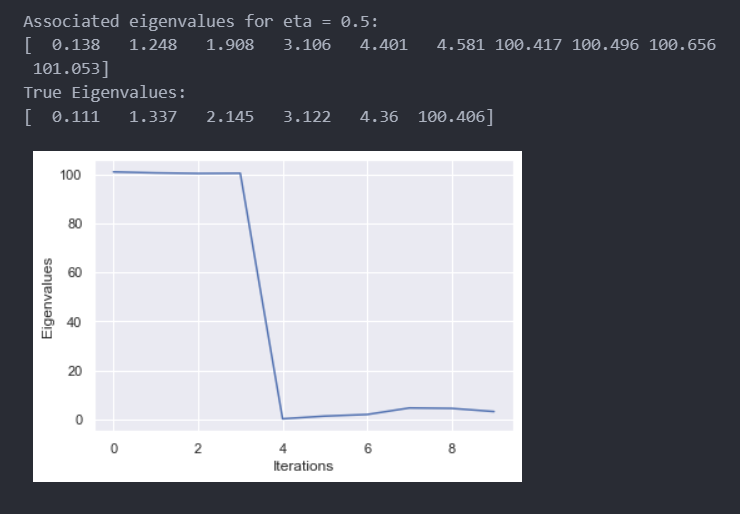
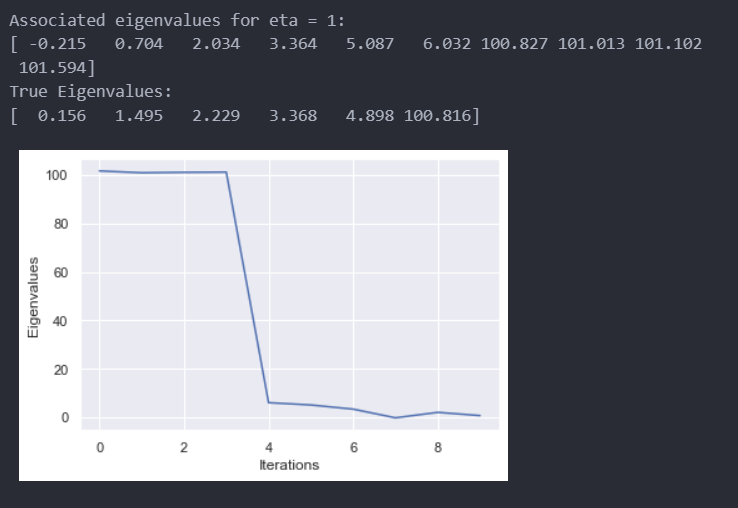
Here, we set up the following. The perturbations will be in the form A +eta\*del\_A



And, upon varying the results, the obtained eigenvalues from the lanczos iteration are as follows:

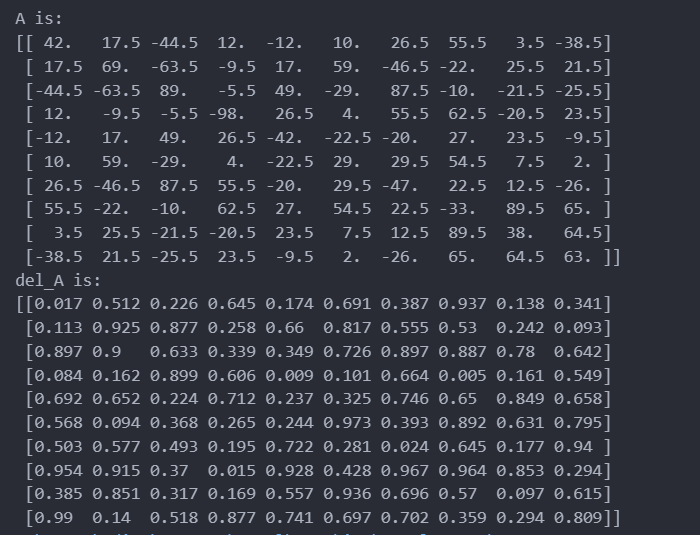
(The plot shows the convergence of the iteration unto various eigenvalues)



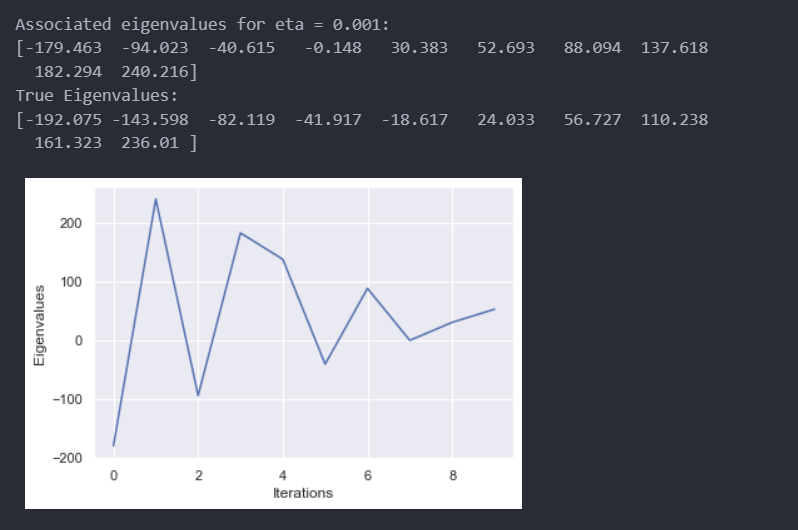
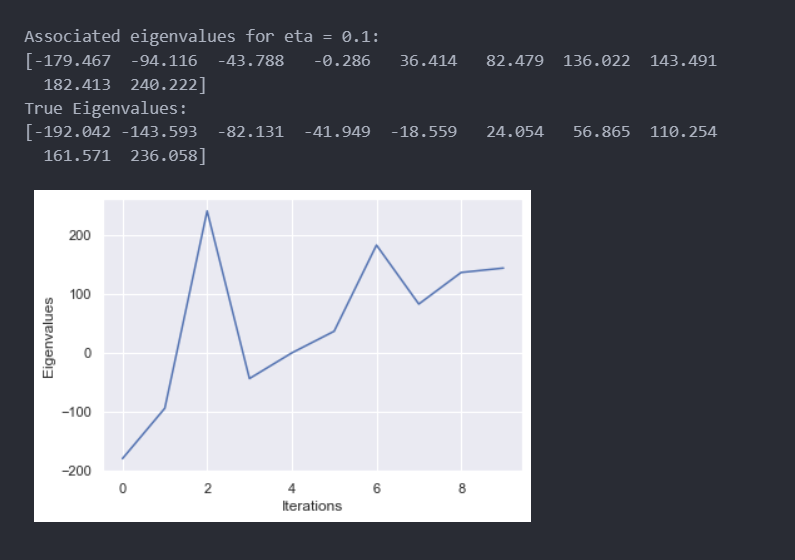
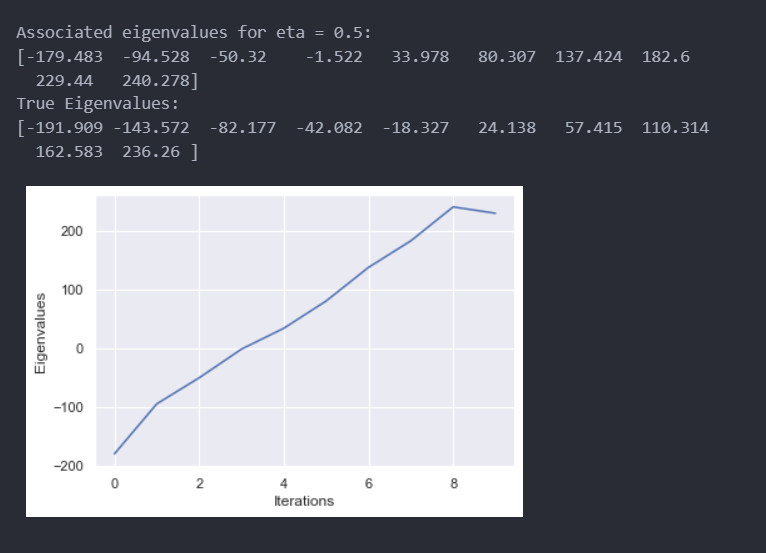
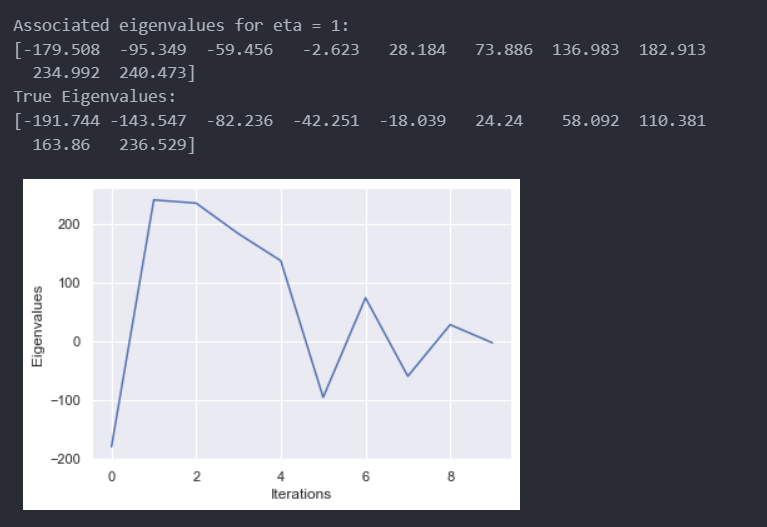
   

Pretty good stuff! Bear in mind the order of magnitude difference between terms in a and del\_A is at least 10^2

## Proper Symmetric Matrix



After setting this up, we proceed as before.

Once again, the numbers say it all. The convergence is less accurate here than before, but still good (?enough)