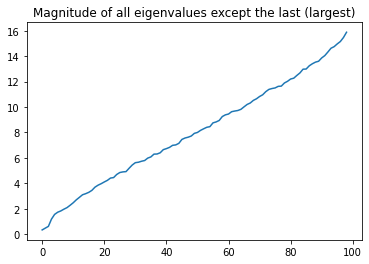
# Matrix Creation

* We generate a 100x100 symmetric random matrix, and add a suitable multiple of the identity matrix to make sure its smallest eigenvalue is postitive. Call this matrix B
* We add a small perturbation to B to create an almost symmetric matrix C, and verify that its smallest eigenvalues are still positive



Above: Eigenvalues for the B matrix plotted above in ascending order, except the last (largest) eigenvalue. Ordinate – Eigenvalue, Abscissa - Index

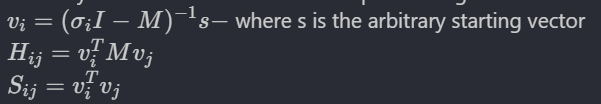
# Creating a Generalized Eigenvalue Equation

* We identify that the eigenvalues in B and C extend to almost 100
* We choose 5 equidistant sigma values in the window from 3 to 10



Above: The sigma values for this problem

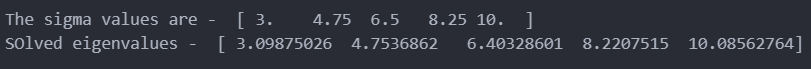
* We create an arbitrary starting vector *s* – it has unit magnitude, 1D vector with 100 entries
* We create vectors V­I , and the 5x5 matrices H and S in the following way –



* Then, we solve the generalized eigenvalue equation shown below to obtain the eigenvalues, and compare them with the sigma values. We use the standard eigenvalue solver in python (numpy)

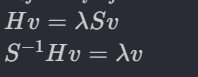


Above: lambda is the eigenvalue corresponding to eigenvector v



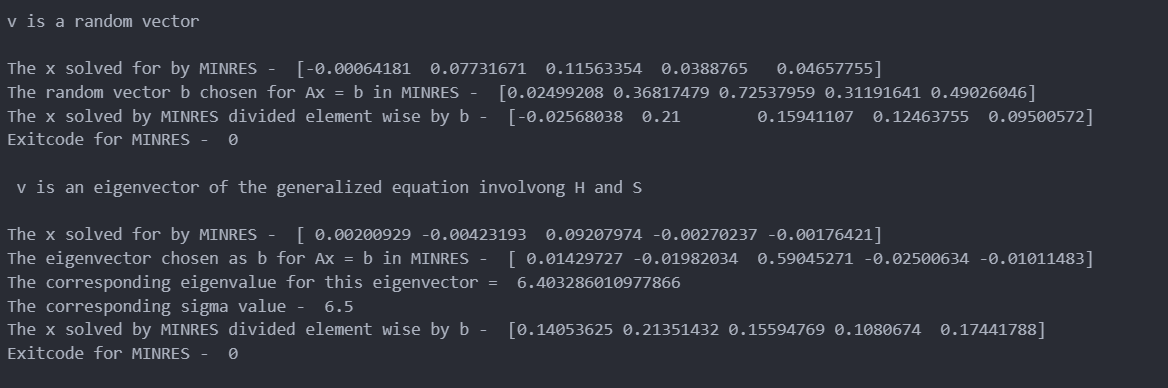
# Using MINRES

I’m not really sure how to use it here, so I try the following –



* Since I know S is invertible here, I multiply both sides with inv(S), replace the vector v on the RHS with a vector of my own choosing, and then solve for what should be v/lambda on the left hand side.
* I take 2 cases – One wherein v is an arbitrary vector, and the second case wherein v is an eigenvector from the generalised eigenvalue problem involving H and S

## B – The symmetric matrix was used to create H and S



## B – The asymmetric matrix was used to create H and S

## 