

Name: Suhas Kamath
SR No: 06-18-01-10-51-25-1-25945
Email ID: suhaskamath@iisc.ac.in
Date: October 26, 2025

Assignment No: Assignment 5 - Question 1
Course Code: DS284
Course Name: Numerical Linear Algebra
Term: AUG 2025

Preparation and Algorithm

The first step in analysing the graph is to obtain the Markov transition graph. The Markov transition matrix for the graph is as follows:

$$\mathbf{M} = \begin{pmatrix} 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0.5 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 \end{pmatrix}$$

We need to run the following power iteration algorithm on the above Markov transition matrix:

```

Initialise a random vector  $\tilde{\mathbf{v}}^{(0)}$  such that  $\|\tilde{\mathbf{v}}^{(0)}\|_2 = 1$ 
for  $k = 1, 2, \dots$  do
     $\tilde{\mathbf{w}} = \mathbf{A}\tilde{\mathbf{v}}^{(k-1)}$ 
     $\tilde{\mathbf{v}}^{(k)} = \frac{\tilde{\mathbf{w}}}{\|\tilde{\mathbf{w}}\|_2}$ 
     $\lambda^{(k)} = (\tilde{\mathbf{v}}^{(k)})^T \mathbf{A}\tilde{\mathbf{v}}^{(k)}$ 
end for

```

The series of steps followed by the Python program is:

1. Build the Markov transition graph \mathbf{M} , using the nodes in the graph.
2. Run the power iteration for a maximum of 100 iterations. Stop the loop if the iteration converges before this. Convergence is measured by taking the 2-norm of the difference of the eigenvectors of successive iterates, i.e. $\|\mathbf{v}^{(k+1)} - \mathbf{v}^{(k)}\|_2$.¹
3. The highest and lowest PageRank and their respective nodes are identified and output.
4. The required plots are saved.

Results

Performing power iterations on this matrix, we obtain the following results:

Number of iterations required: 38
 Highest PageRank: Node 5 (PageRank: 0.4685)
 Lowest PageRank: Node 3 (PageRank: 0.0000)

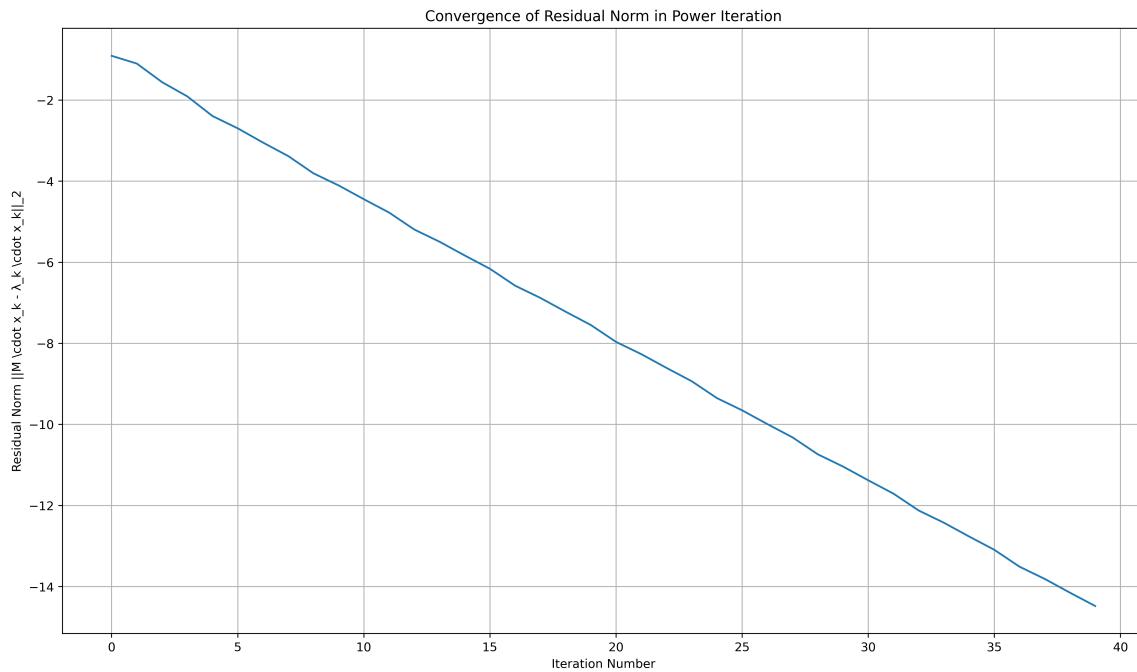
¹For this question, as no tolerance was specified, I decided to use 10^{-6} .

Name: Suhas Kamath
SR No: 06-18-01-10-51-25-1-25945
Email ID: suhaskamath@iisc.ac.in
Date: October 26, 2025

Assignment No: Assignment 5 - Question 1
Course Code: DS284
Course Name: Numerical Linear Algebra
Term: AUG 2025

Plots

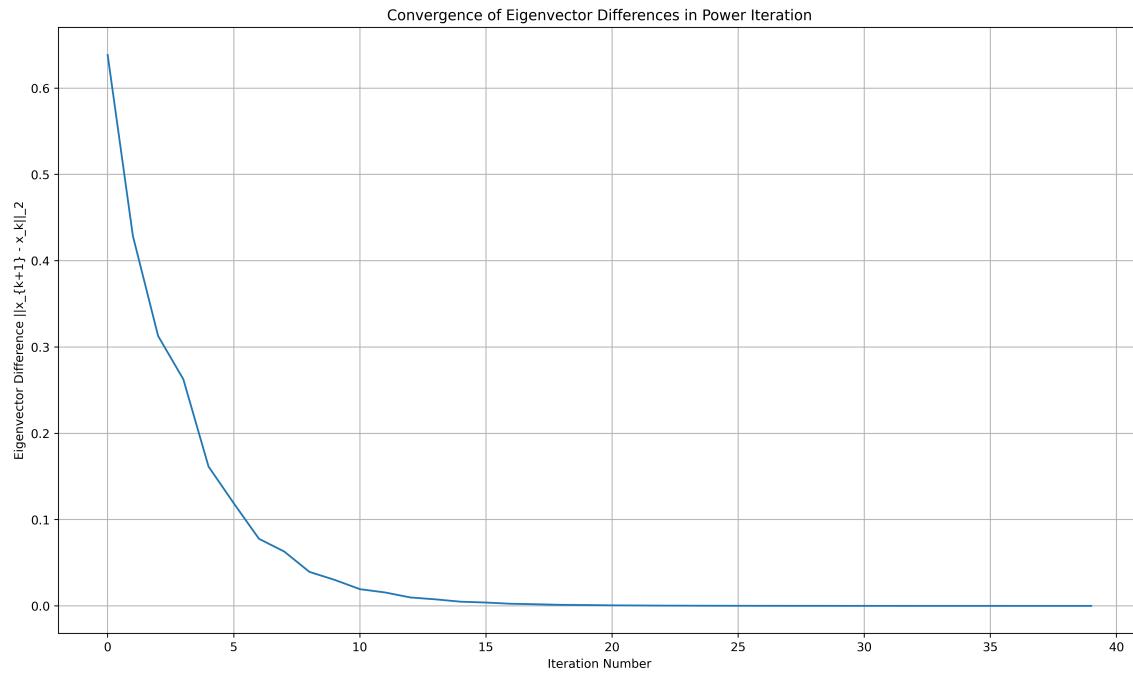
The log plot of the 2-norm of the residual of the eigenvalue problem, i.e. $\log \|\mathbf{M}\mathbf{x}^{(k+1)} - \lambda^{(k+1)}\mathbf{x}^{(k+1)}\|_2$, is as follows:



Name: Suhas Kamath
SR No: 06-18-01-10-51-25-1-25945
Email ID: suhaskamath@iisc.ac.in
Date: October 26, 2025

Assignment No: Assignment 5 - Question 1
Course Code: DS284
Course Name: Numerical Linear Algebra
Term: AUG 2025

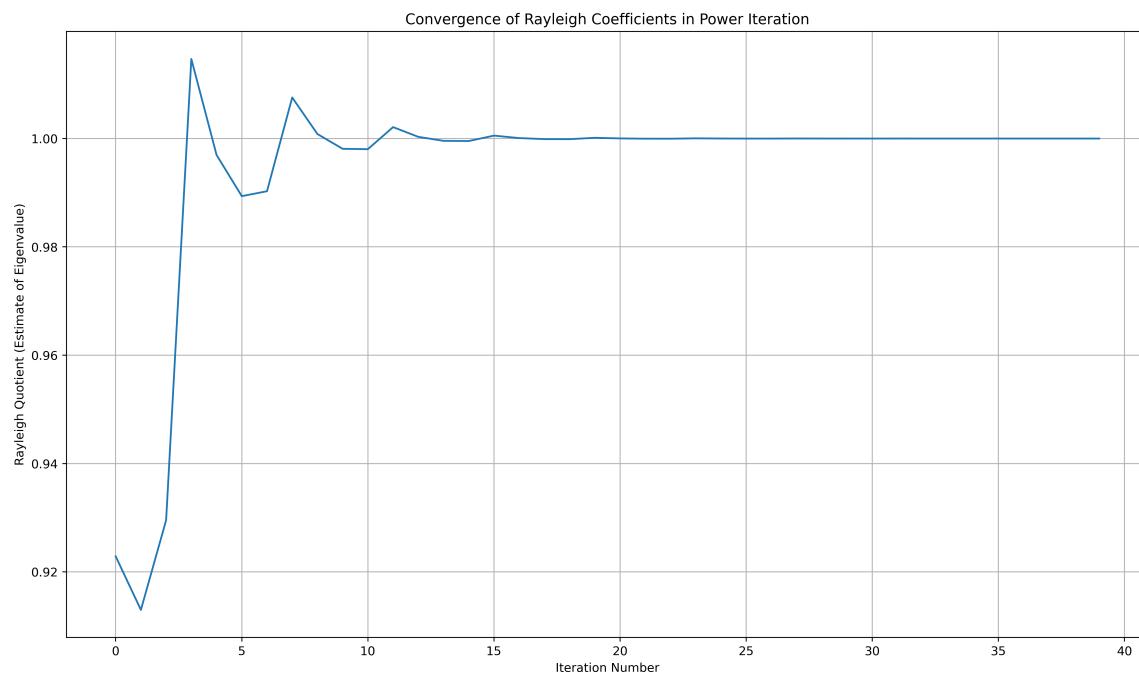
The 2-norm plot of the difference between the vectors of successive iterates, i.e. $\|x^{(k+1)} - x^{(k)}\|_2$, is as follows:



Name: Suhas Kamath
SR No: 06-18-01-10-51-25-1-25945
Email ID: suhaskamath@iisc.ac.in
Date: October 26, 2025

Assignment No: Assignment 5 - Question 1
Course Code: DS284
Course Name: Numerical Linear Algebra
Term: AUG 2025

The Rayleigh quotient calculated at each stage of the iteration is as follows:



Name: Suhas Kamath
SR No: 06-18-01-10-51-25-1-25945
Email ID: suhaskamath@iisc.ac.in
Date: October 26, 2025

Assignment No: Assignment 5 - Question 1
Course Code: DS284
Course Name: Numerical Linear Algebra
Term: AUG 2025

Code

The Python code is as follows:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import os
4
5 # Multiple indexes may have the same max value
6 # In other words, a PageRank tie is possible
7 # Returns tuple: (array of max indexes, max value)
8 def find_max_indexes(vector):
9     max_value = np.max(vector)
10    max_indexes = np.where(vector == max_value)[0]
11    return (max_indexes, max_value)
12
13 # Multiple indexes may have the same min value
14 # In other words, a PageRank tie is possible
15 # Returns tuple: (array of min indexes, min value)
16 def find_min_indexes(vector):
17     min_value = np.min(vector)
18     min_indexes = np.where(vector == min_value)[0]
19     return (min_indexes, min_value)
20
21 # --- Build the Markov transition matrix M ---
22 M = np.zeros((10, 10))
23
24 # Outgoing links dictionary: for each node j, list nodes i that j links to.
25 # Interpretation: j -> i for each i in links[j].
26 links = {
27     0: [1, 2], 1: [2, 5], 2: [0, 1], 3: [4], 4: [5, 7],
28     5: [6, 9], 6: [2, 7], 7: [4, 5], 8: [1, 7], 9: [8]
29 }
30
31 # Fill M so each nonzero column j distributes mass uniformly over the
32 # outgoing links of j. For example, if node j points to k_j targets,
33 # each target i receives 1/k_j probability from column j.
34 for j, i_list in links.items():
35     k_j = len(i_list)
36     for i in i_list:
37         M[i, j] = 1.0 / k_j
38
39 # --- Power iteration parameters and initialization ---
40 max_iterations = 100          # hard cap to avoid infinite loops
41 tolerance = 1e-6             # stopping criterion on eigenvector change
42
43 # Random initial vector (nonzero), normalized to unit 2-norm.
44 x_k = np.random.rand(M.shape[0])
45 x_k /= np.linalg.norm(x_k)
46
47 # Containers to store convergence diagnostics for plotting/analysis
48 residuals = []                # ||M x_k - lambda_k x_k||_2
49 eigenvector_differences = []    # ||x_{k+1} - x_k||_2
50 eigenvalues = []                # Rayleigh quotient estimates
51
52 print("Starting power iteration...")
```

Name: Suhas Kamath
SR No: 06-18-01-10-51-25-1-25945
Email ID: suhaskamath@iisc.ac.in
Date: October 26, 2025

Assignment No: Assignment 5 - Question 1
Course Code: DS284
Course Name: Numerical Linear Algebra
Term: AUG 2025

```
54 # --- Power iteration loop ---
55 for k in range(max_iterations):
56     # Apply the matrix and renormalize to keep vector magnitude bounded
57     x_k_plus_1 = M @ x_k
58     x_k_plus_1 /= np.linalg.norm(x_k_plus_1)
59
60     # Rayleigh quotient provides an estimate of the eigenvalue for x_{k+1}
61     eigenval = x_k_plus_1.T @ M @ x_k_plus_1
62     eigenvalues.append(eigenval)
63
64     # Residual norm: measures how well (lambda, x) satisfies the eigen eqn
65     residuals.append(np.log(np.linalg.norm(M @ x_k_plus_1 - eigenval * x_k_plus_1)))
66
67     # Change in eigenvector between iterations (useful stopping metric)
68     eigenvector_differences.append(np.linalg.norm(x_k_plus_1 - x_k))
69
70     x_k = x_k_plus_1
71
72     # Check convergence
73     if eigenvector_differences[-1] < tolerance:
74         print(f"\nConverged after {k+1} iterations.")
75         break
76
77     # PageRank interpretation: normalize so components sum to 1 (probability)
78     pagerank_vector = x_k / np.linalg.norm(x_k)
79
80     for i in range(len(pagerank_vector)):
81         print(f"Node {i}: PageRank = {pagerank_vector[i]}")
82
83     print(f"Highest PageRank: Node(s) {find_max_indexes(pagerank_vector)[0]} (Rank:
84     → {find_max_indexes(pagerank_vector)[1]:.4f})")
85     print(f"Lowest PageRank: Node(s) {find_min_indexes(pagerank_vector)[0]} (Rank:
86     → {find_min_indexes(pagerank_vector)[1]:.4f})")
87
87 save_dir = r"H:\My Drive\Numerical Linear Algebra\Assignments\Assignment 5"
88 print(f"\nAttempting to save plots to: {save_dir}")
89
90 #Plot 1
91 plt.figure(figsize=(16, 9))
92 plt.plot(residuals)
93 plt.title('Convergence of Residual Norm in Power Iteration')
94 plt.xlabel('Iteration Number')
95 plt.ylabel('Residual Norm ||M x_k - lambda_k x_k||_2')
96 plt.grid(True)
97 plt.savefig(os.path.join(save_dir, 'residual_norm.png'), dpi=400)
98
99 #Plot 2
100 plt.figure(figsize=(16, 9))
101 plt.plot(eigenvector_differences)
102 plt.title('Convergence of Eigenvector Differences in Power Iteration')
103 plt.xlabel('Iteration Number')
104 plt.ylabel('Eigenvector Difference ||x_{k+1} - x_k||_2')
105 plt.grid(True)
106 plt.savefig(os.path.join(save_dir, 'eigenvector_differences.png'), dpi=400)
```

Name: Suhas Kamath
SR No: 06-18-01-10-51-25-1-25945
Email ID: suhaskamath@iisc.ac.in
Date: October 26, 2025

Assignment No: Assignment 5 - Question 1
Course Code: DS284
Course Name: Numerical Linear Algebra
Term: AUG 2025

```
108
109 #Plot 3
110 plt.figure(figsize=(16,9))
111 plt.plot(eigenvalues)
112 plt.title('Convergence of Rayleigh Coefficients in Power Iteration')
113 plt.xlabel('Iteration Number')
114 plt.ylabel('Rayleigh Quotient (Estimate of Eigenvalue)')
115 plt.grid(True)
116 plt.savefig(os.path.join(save_dir, 'rayleigh_coefficients.png'), dpi=400)
117
118 print("\nAll plots saved.")
```

Output

The output of the code is as follows:

```
1 Starting power iteration...
2
3 Converged after 38 iterations.
4 Node 0: PageRank = 0.23426049470201327
5 Node 1: PageRank = 0.4685210514425084
6 Node 2: PageRank = 0.46852134899268866
7 Node 3: PageRank = 0.0
8 Node 4: PageRank = 0.15617386821293897
9 Node 5: PageRank = 0.46852161969991823
10 Node 6: PageRank = 0.23426071515052027
11 Node 7: PageRank = 0.3123473886175061
12 Node 8: PageRank = 0.23426042938272038
13 Node 9: PageRank = 0.23426071515052027
14 Highest PageRank: Node(s) [5] (Rank: 0.4685)
15 Lowest PageRank: Node(s) [3] (Rank: 0.0000)
16
17 Attempting to save plots to: H:\My Drive\Numerical Linear
   → Algebra\Assignments\Assignment 5
18
19 All plots saved.
```

Q) 2) a) FALSE. We know of only one definite method of solving for eigenvalues: solving characteristic equations. However, solving characteristic equations is unstable - Even if we have an infinite precision computer, we do not have a method to solve characteristic equations above degree 5. To conclude, we do not have a numerically stable direct method to calculate eigenvalues for $m \geq 5$ even if we possess an infinite precision computer.

Q) 2) b) FALSE. Most eigensolvers employ a two phase method: the first phase converts the matrix to an upper Hessenberg matrix. An upper Hessenberg matrix is a upper triangular matrix with an addition 'line' of non-zero elements parallel to the diagonal. This phase has a time complexity of $O(m^3)$. The second phase further reduces the upper Hessenberg matrix to an upper triangular matrix using $O(m)$ iterations, each with $O(m^2)$ iterations, thereby bringing the time complexity of phase 2 to $O(m^3)$.

If phase 1 is not performed, we shall have to reduce a full (dense) matrix in $O(m)$ iterations, with each iteration requiring $O(m^3)$ operations. This shall raise the time complexity to $O(m^4)$.

To conclude, if a matrix does not go through phase-1, we will not be able to get the eigenvalue in $O(m^3)$ time. We shall require $O(m^4)$ time.

2) c) FALSE. In power iteration, we need to provide an initial "guess" vector denoted by $\tilde{v}^{(0)}$. The entire power iteration depends on the premise that $\tilde{v}^{(0)}$ is not orthogonal to the eigenvector corresponding to the largest eigenvalue. In other words, if the eigenvector corresponding to the largest eigenvalue is denoted by \tilde{x}_1 , $\tilde{x}_1^T \tilde{v}^{(0)} \neq 0$ for $\tilde{v}^{(0)}$ to converge to \tilde{x}_1 . Hence, not all vectors which are non-zero can qualify as an initial guess.

Q2) d)

TRUE

We know that a householder matrix \tilde{F} can be constructed using the following formula:

$$\tilde{F} = \frac{I}{n} - \frac{2\tilde{u}\tilde{u}^T}{\tilde{u}^T\tilde{u}}$$

Let us check whether \tilde{F} is orthogonal. For \tilde{F} to be orthogonal, it should satisfy the following property:

$$\tilde{F} \cdot \tilde{F}^T = \frac{I}{n}$$

Let us first find \tilde{F}^T :

$$\tilde{F} = \frac{I}{n} - \frac{2\tilde{u}\tilde{u}^T}{\tilde{u}^T\tilde{u}}$$

$$\tilde{F}^T = \left(\frac{I}{n} - \frac{2\tilde{u}\tilde{u}^T}{\tilde{u}^T\tilde{u}} \right)^T$$

$$= \frac{I^T}{n} - \left(\frac{2}{\tilde{u}^T\tilde{u}} \cdot \tilde{u}\tilde{u}^T \right)^T \quad \begin{array}{l} \text{Transpose of sum is sum of} \\ \text{transposes} \end{array}$$

$$= \frac{I}{n} - (\tilde{u}^T)^T \cdot \frac{2}{\tilde{u}^T\tilde{u}} \cdot \tilde{u}^T \quad \begin{array}{l} \text{[Reversal law]} \\ \text{[} \frac{2}{\tilde{u}^T\tilde{u}} \text{ is a constant} \text{]} \end{array} \quad \left[\frac{I^T}{n} = \frac{I}{n} \right]$$

$$= \frac{I}{n} - \frac{\tilde{u} \cdot \tilde{u}^T \cdot 2}{\tilde{u}^T\tilde{u}}$$

$$= \frac{I}{n} - \frac{2}{\tilde{u}^T\tilde{u}} \tilde{u}\tilde{u}^T$$

$$= \tilde{F} \quad \text{(Hence proven, } \tilde{F} \text{ is symmetric, i.e. } \tilde{F} = \tilde{F}^T \text{)}$$

Hence, let us calculate $\underline{F} \cdot \underline{F}^T$:

$$\underline{F} \cdot \underline{F}^T = \underline{F}^2 \quad [\underline{F} = \underline{F}^T]$$

$$= \left(\underline{I} - \frac{2}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T \right) \left(\underline{I} - \frac{2}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T \right)$$

$$= \underline{I}^2 - \frac{2}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T \cdot \underline{I} - \frac{\underline{I} \cdot 2}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T +$$

$$\frac{2}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T \cdot \frac{2}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T$$

$$= \underline{I} - \frac{4}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T + \frac{4}{(\underline{u}^T \underline{u})^2} \cdot \underline{u} \underline{u}^T \cdot \underline{u} \underline{u}^T$$

$$= \underline{I} - \frac{4}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T + \frac{4}{(\underline{u}^T \underline{u})^2} \cdot \underline{u} (\underline{u}^T \underline{u}) \underline{u}^T$$

$$= \underline{I} - \frac{4}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T + \frac{4}{\underline{u}^T \underline{u}} \underline{u} \underline{u}^T$$

$$= \underline{I}$$

(Hence proven, \underline{F} is orthogonal)

Hence, because \underline{F} is orthogonal, we know that $\underline{F}^T = \underline{F}^{-1}$.

$$\text{Let } \underline{\underline{F}} \underline{\underline{A}} \underline{\underline{F}}^T = \underline{\underline{F}} \underline{\underline{A}} \underline{\underline{F}}^{-1} = \underline{\underline{B}}$$

Proof of $\underline{A} = \underline{F}^{-1} \underline{B} \underline{F}$:

$$\text{Then } \underline{A} = \underline{F}^{-1} \underline{B} \underline{F}$$

$$= \underline{F}^{-1} \underline{B} (\underline{F}^{-1})^{-1}$$

$$\underline{A} = \underline{F}^{-1} \cdot \underline{\underline{F}} \underline{\underline{A}} \underline{\underline{F}}^{-1} \cdot \underline{F}$$

$$= \underline{I} \cdot \underline{A} \cdot \underline{I}$$

$$= \underline{A}$$

Hence, because a similarity transformation exists between \underline{A} and $\underline{B} = \underline{\underline{F}} \underline{\underline{A}} \underline{\underline{F}}^T$, \underline{A} and $\underline{\underline{F}} \underline{\underline{A}} \underline{\underline{F}}^T$ have the same eigenvalues. However, they may differ in eigen vectors.

3) a) We are given that $\tilde{A}\underline{u}_i = \lambda_i \tilde{B}\underline{u}_i$.

Because \tilde{B} is a symmetric positive definite matrix, we can perform a Cholesky decomposition on \tilde{B} such that $\tilde{B} = \tilde{R}\tilde{R}^T$.

Hence, the original equation can be re-written as:

$$\tilde{A}\underline{u}_i = \lambda_i (\tilde{R}\tilde{R}^T)\underline{u}_i \quad [\text{Substitute } \tilde{B} = \tilde{R}\tilde{R}^T]$$

$$\tilde{R}^{-1}\tilde{A}\underline{u}_i = \lambda_i (\tilde{R}^{-1}\tilde{R}\tilde{R}^T\underline{u}_i) \quad \begin{bmatrix} \text{Pre-multiply with } \tilde{R}^{-1}, \text{ we know } \tilde{R} \text{ is non-singular} \\ \text{due to properties of Cholesky decomposition} \end{bmatrix}$$

$$\tilde{R}^{-1}\tilde{A}\underline{u}_i = \lambda_i \tilde{R}^T\underline{u}_i$$

$$\tilde{R}^{-1}\tilde{A} \cdot (\tilde{R}^T)^{-1}\underline{v}_i = \lambda_i \underline{v}_i \quad \begin{bmatrix} \text{Let } \tilde{R}^T\underline{u}_i = \underline{v}_i, \text{ then } \underline{u}_i = (\tilde{R}^T)^{-1}\underline{v}_i \end{bmatrix}$$

$$\therefore \underline{H}\underline{v}_i = \lambda_i \underline{v}_i \quad \begin{bmatrix} \text{Let } \underline{H} = \tilde{R}^{-1}\tilde{A}(\tilde{R}^T)^{-1} \end{bmatrix}$$

Let us also check if \underline{H} is symmetric:

$$\begin{aligned} \underline{H}^T &= (\tilde{R}^{-1}\tilde{A}(\tilde{R}^T)^{-1})^T = ((\tilde{R}^T)^{-1})^T \tilde{A}^T (\tilde{R}^{-1})^T \\ &= ((\tilde{R}^T)^T)^{-1} \tilde{A}^T (\tilde{R}^T)^{-1} \quad [(\tilde{R}^T)^{-1} = (\tilde{R}^{-1})^T] \\ &= \tilde{R}^{-1} \tilde{A}^T (\tilde{R}^T)^{-1} \quad [(\tilde{R}^T)^T = \tilde{R}] \\ &= \underline{H} \quad (\text{shown}) \end{aligned}$$

Q3) b)

Because we are given a value 2.0 and are told that this value is close to an eigenvalue λ_i , we can use inverse power iterations. The algorithm for this iteration is as follows:

1 Initialize $\mu = 2.0$ // value close to eigenvalue

2 Initialize $\tilde{v}^{(0)} = \text{some random vector}$

3 For $k = 1 \rightarrow \infty$:

4 $\tilde{w} = (\tilde{H} - \mu \tilde{I})^{-1} \tilde{v}^{(k-1)}$ // We can solve for \tilde{w} in

$$(\tilde{H} - \mu \tilde{I})\tilde{w} = \tilde{v}^{(k-1)} \text{ to get } \tilde{w}.$$

This avoids the calculation of matrix inverse.

5 $\tilde{v}^{(k)} = \frac{\tilde{w}}{\|\tilde{w}\|}$ // Normalize \tilde{w}

6 $\lambda^{(k)} = (\tilde{v}^{(k)})^\top \tilde{v}^{(k)}$

$\tilde{u}_i = (\tilde{L}^\top)^{-1} \tilde{v}^{(k)}$ // $\tilde{v}^{(k)}$ is an eigenvector of \tilde{H} , we need to transform it back to the eigenvector in the original problem.

For this algorithm to converge to λ_i as $k \rightarrow \infty$, the following conditions must be fulfilled:

i) There must be a unique eigenvalue, i.e. an eigenvalue with unitary algebraic multiplicity close to the initial value of 2.0.

ii) The random vector chosen in the second step must not be orthogonal to the eigenvector corresponding to the eigenvalue λ_i .

Q3)c)

The computationally dominant step is step 4, which is obtaining the solution to the linear system of equations $(\tilde{H} - \mu \tilde{I})\tilde{w} = \tilde{v}^{(k-1)}$.

Because $A \in \mathbb{R}$ and B are large sparse matrices, it will be easier for us to work with them.

Let us convert this linear system of equations to use the original sparse matrices:

$$(\underline{H} - \mu \underline{I}) \underline{w} = \underline{v}^{(k-1)}$$

$$(\underline{R}^{-1} \underline{A} (\underline{R}^T)^{-1} - \mu \underline{I}) \underline{w} = \underline{v}^{(k-1)} \quad [\text{Substitute } \underline{H} = \underline{R}^{-1} \underline{A} (\underline{R}^T)^{-1}]$$

$$(\underline{R} \underline{R}^{-1} \underline{A} (\underline{R}^T)^{-1} - \underline{R} (\mu \underline{I})) \underline{w} = \underline{R} \underline{v}^{(k-1)} \quad [\text{Left multiply } \underline{R}]$$

$$(\underline{A} (\underline{R}^T)^{-1} - \mu \underline{R}) \underline{w} = \underline{R} \underline{v}^{(k-1)}$$

$$(\underline{A} - \mu \underline{R} \underline{R}^T) (\underline{R}^T)^{-1} \underline{w} = \underline{R} \underline{v}^{(k-1)} \quad [\text{Factorise } (\underline{R}^T)^{-1}]$$

$$(\underline{A} - \mu \underline{B}) (\underline{R}^T)^{-1} \underline{w} = \underline{R} \underline{v}^{(k-1)} \quad [\text{Substitute } \underline{R} \underline{R}^T = \underline{B}]$$

$$(\underline{A} - \mu \underline{B}) \underline{t} = \underline{R} \underline{v}^{(k-1)} \quad [\text{Let } \underline{t} = (\underline{R}^T)^{-1} \underline{w}]$$

We can solve this equation in a two-step process:

1) Solve for \underline{t} in the linear system $(\underline{A} - \mu \underline{B}) \underline{t} = \underline{R} \underline{v}^{(k-1)}$

2) Solve for \underline{w} in the linear system $(\underline{R}^T)^{-1} \underline{w} = \underline{t}$

Because the matrices \underline{A} & \underline{B} are sparse, this method shall be more computationally efficient than solving directly for \underline{w} in the linear system $(\underline{H} - \mu \underline{I}) \underline{w} = \underline{v}^{(k-1)}$.

Q3)d) Because we are given 2.0 is close to λ_1 , and λ_1 is the smallest eigenvalue (by magnitude), we can further optimise by tuning a value of μ such that $\mu < \lambda_1$. In this case, we choose $\mu = 1.5$.

When we tune a value of $\mu < \lambda_1$, we can guarantee that $(\underline{A} - \mu \underline{B})$ is symmetric, positive & definite (S.P.D.)

This can be easily proven as follows:

S.P.D's must follow $\underset{\sim}{x}^T \underset{\sim}{M} \underset{\sim}{x} = 0 \quad \forall \underset{\sim}{x} \in \mathbb{R}^m$

$$\begin{aligned}\underset{\sim}{x}^T (\underset{\sim}{A} - \mu \underset{\sim}{B}) \underset{\sim}{x} &= \underset{\sim}{x}^T \underset{\sim}{A} \underset{\sim}{x} - \mu \underset{\sim}{x}^T \underset{\sim}{B} \underset{\sim}{x} \\ &\geq (\lambda_{\min} - \mu) \underset{\sim}{x}^T \underset{\sim}{B} \underset{\sim}{x} \quad [A \text{ is given as S.P.D.}] \\ &\geq 0 \quad [\text{B is positive definite}]\end{aligned}$$

Hence, because we know $(A - \mu B)$ is S.P.D., we can perform an L.V. decomposition on this and hence solve the first linear system of equations faster.

Q4) a) From the definition of backward stability, we can state that:

$$(\underline{A} + \underline{\delta A}) \tilde{\underline{w}} = \underline{v}, \text{ where } \|\underline{\delta A}\| = O(\epsilon_m) \cdot \|\underline{A}\|$$

$$(\underline{A} + \underline{\delta A})(\underline{w} + \underline{\delta w}) = \underline{v} \quad [\tilde{\underline{w}} = \underline{w} + \underline{\delta w}]$$

$$\cancel{\underline{A}\underline{w} + \underline{\delta A}\underline{w} + \underline{A}\underline{\delta w} + \underline{\delta A}\underline{\delta w}} = \cancel{\underline{A}\underline{w}} \quad [\underline{A}\underline{w} = \underline{v}]$$

$$\underline{\delta A}\underline{w} + \underline{A}\underline{\delta w} + \underline{\delta A}\underline{\delta w} = 0$$

$$(\underline{A} + \underline{\delta A})\underline{\delta w} = -\underline{\delta A}\underline{w}$$

$$\underline{\delta w} = (\underline{A} + \underline{\delta A})^{-1}(-\underline{\delta A}\underline{w})$$

$$= -(\underline{A} + \underline{\delta A})^{-1}\underline{\delta A}\underline{w} \quad (\text{shown})$$

Q4)b) Because \underline{A} is symmetric, we know that \underline{A} is non-degenerate and hence has an eigen-decomposition. Furthermore, we also know that the set of vectors $\{q_1, \dots, q_m\}$ forms a basis for \mathbb{R}^m .

Because it is given that \underline{v} has components from all eigenvectors of \underline{A} , we can state:

$$\underline{v} = \sum_{i=1}^m c_i \underline{q}_i$$

Let us try to solve for $\underline{A}\underline{w} = \underline{v}$:

$$\underline{A}\underline{w} = \underline{v}$$

$$\underline{w} = \underline{A}^{-1}\underline{v} \quad [A \text{ is known to be invertible}]$$

$$\underline{w} = \underline{A}^{-1} \left(\sum_{i=1}^m c_i \underline{q}_i \right) \quad \left[\text{Substitute } \underline{v} = \sum_{i=1}^m c_i \underline{q}_i \right]$$

$$= \sum_{i=1}^m c_i \underline{A}^{-1} \underline{q}_i$$

$$= \sum_{i=1}^m c_i \cdot \frac{\underline{q}_i}{\lambda_i} \quad \left[\text{Because } \underline{A} \underline{q}_i = \lambda_i \underline{q}_i, \quad \underline{A}^{-1} \underline{q}_i = \frac{\underline{q}_i}{\lambda_i} \right]$$

Because $|\lambda_1| < |\lambda_2| \leq |\lambda_3| \leq \dots \leq |\lambda_m|$, we conclude that:

$$\left| \frac{1}{\lambda_1} \right| >> \left| \frac{1}{\lambda_2} \right| \geq \left| \frac{1}{\lambda_3} \right| \geq \dots \geq \left| \frac{1}{\lambda_m} \right|$$

Hence, the coefficient of \underline{q}_1 , i.e. $\frac{c_1}{\lambda_1}$ shall 'dominate' over all of the other eigenvectors' coefficients.

$$\text{Therefore, } \underline{w} \approx \frac{c_1}{\lambda_1} \underline{q}_1$$

Normalizing \underline{w} , we get:

$$\frac{\underline{w}}{\|\underline{w}\|} = \frac{\frac{c_1}{\lambda_1} \underline{q}_1}{\left\| \frac{c_1}{\lambda_1} \underline{q}_1 \right\|}$$

$$= \frac{\frac{c_1}{\lambda_1} \underline{q}_1}{\left| \frac{c_1}{\lambda_1} \right| \|\underline{q}_1\|_2} \quad \left[\|\alpha \underline{x}\| = \alpha \|\underline{x}\| \right]$$

$$= \pm 1 \cdot \frac{\underline{q}_1}{\|\underline{q}_1\|_2} \quad \left[\frac{\alpha}{|\alpha|} = \pm 1 \right]$$

$$= \pm \underline{q}_1 \quad (\text{shown}) \quad \left[\|\underline{q}_1\| = 1 \right]$$

(Q4)c) We are given that $(\underline{A} + \delta\underline{A})^{-1} \simeq \underline{A}^{-1} + \underline{A}^{-1} (\delta\underline{A}) \underline{A}^{-1}$

From a), we know that $\delta w = -(\underline{A} + \delta\underline{A})^{-1} (\delta\underline{A}) \underline{w}$

Substituting the Taylor series expansion for $(\underline{A} + \delta\underline{A})^{-1}$, we get:

$$\delta \underline{w} = -(\underline{A} + \delta\underline{A})^{-1} (\delta\underline{A}) \underline{w}$$

$$\delta \underline{w} \simeq -\left(\underline{A}^{-1} + \underline{A}^{-1} (\delta\underline{A}) \underline{A}^{-1}\right) (\delta\underline{A}) \underline{w}$$

$$\simeq \underline{A}^{-1} (\delta\underline{A}) \underline{A}^{-1} (\delta\underline{A}) \underline{w} - \underline{A}^{-1} (\delta\underline{A}) \underline{w}$$

Because $\|\delta A\| = O(\epsilon_m)$, $\underline{A}^{-1} (\delta\underline{A}) \underline{A}^{-1} (\delta\underline{A}) \underline{w} \simeq 0$

$$\text{Hence, } \delta \underline{w} \simeq -\underline{A}^{-1} (\delta\underline{A}) \underline{w}$$

$$\simeq -\underline{A}^{-1} \underline{z} \quad [\text{Let } \underline{z} = (\delta\underline{A}) \underline{w}]$$

Because $\{q_1, \dots, q_m\}$ form a basis for \mathbb{R}^m , we can express \underline{z} as a linear combination of these vectors:

$$\underline{z} = \sum_{j=0}^m d_j q_j$$

Substituting this, we get:

$$\delta \underline{w} \simeq -\underline{A}^{-1} \sum_{j=0}^m d_j q_j$$

$$\simeq -\sum_{j=0}^m d_j \underline{A}^{-1} q_j$$

$$\simeq -\sum_{j=0}^m \frac{d_j}{\lambda_j} q_j \quad [\underline{A}^{-1} q_j = \frac{1}{\lambda_j} q_j]$$

Because we know from part b) that $\left| \frac{1}{\lambda_1} \right| > \left| \frac{1}{\lambda_2} \right| \geq \left| \frac{1}{\lambda_3} \right| \geq \cdots \geq \left| \frac{1}{\lambda_m} \right|$

we can state that $\underset{\sim}{\delta w} = - \frac{d_1}{\lambda_1} \underset{\sim}{g_1}$

Because $\underset{\sim}{\delta w} \approx K \cdot \underset{\sim}{g_1}$, $\underset{\sim}{\delta w}$ is approximately in the direction of $\underset{\sim}{g_1}$.

(proven)

Q) 5)a) Because ε_i is said to be the exact eigenvalue of A , we write:

$$\hat{A} \tilde{x}_i = \varepsilon_i \tilde{x}_i$$

It is given that the exact eigenvector is the sum of the current approximation, i.e. $\tilde{\tilde{x}}_i^{(0)}$ and the corrector, i.e. \tilde{t}_i . Hence we can state that:

$$\tilde{x}_i = \tilde{\tilde{x}}_i^{(0)} + \tilde{t}_i$$

Substituting this into the previous equation, we get:

$$\hat{A}(\tilde{\tilde{x}}_i^{(0)} + \tilde{t}_i) = \varepsilon_i (\tilde{\tilde{x}}_i^{(0)} + \tilde{t}_i)$$

$$\hat{A} \tilde{\tilde{x}}_i^{(0)} + \hat{A} \tilde{t}_i = \varepsilon_i \tilde{\tilde{x}}_i^{(0)} + \varepsilon_i \tilde{t}_i$$

$$\hat{A} \tilde{t}_i - \varepsilon_i \tilde{t}_i = \varepsilon_i \tilde{\tilde{x}}_i^{(0)} - \hat{A} \tilde{\tilde{x}}_i^{(0)}$$

$$(\hat{A} - \varepsilon_i I) \tilde{t}_i = (I - \hat{A}) \tilde{\tilde{x}}_i^{(0)} \quad (\text{shown})$$

From the above expression, we see that it is actually a linear system of equations. However, we know that $\det(\hat{A} - \varepsilon_i \tilde{t}_i)$ is zero, which makes $(\hat{A} - \varepsilon_i \tilde{t}_i)$ singular. Hence, there does not exist a unique solution for this equation.

Q) b)

Given an approximate eigenvector, i.e. $\tilde{\tilde{x}}_i^{(0)}$, the best possible estimate for its corresponding eigenvalue is the Rayleigh quotient. This quotient can be expressed as:

$$\tilde{q}_i^{(0)} = \frac{(\tilde{\tilde{x}}_i^{(0)})^T \hat{A} (\tilde{\tilde{x}}_i^{(0)})}{(\tilde{\tilde{x}}_i^{(0)})^T (\tilde{\tilde{x}}_i^{(0)})}$$

Substituting $\varepsilon_i \approx \tilde{\varepsilon}_i^{(0)}$ and replacing \tilde{A} with $\tilde{D} = \text{diag}(\tilde{A})$, we get :

$$(\tilde{D} - \tilde{\varepsilon}_i^{(0)} \tilde{I}) \tilde{f}_i = (\varepsilon_i^{(0)} \tilde{I} - \tilde{A}) \tilde{x}_i^{(0)}$$

We also know that $(\tilde{D} - \tilde{\varepsilon}_i^{(0)} \tilde{I})$ shall be a diagonal matrix, and shall be invertible, as long as $\varepsilon_i^{(0)} \neq D_{ii} \forall i \in \mathbb{N}$. Finally, we can derive an expression for \tilde{f}_i :

$$\tilde{f}_i = (\tilde{D} - \tilde{\varepsilon}_i^{(0)} \tilde{I})^{-1} (\varepsilon_i^{(0)} \tilde{I} - \tilde{A}) \tilde{x}_i^{(0)}$$

Q5) c)

In the current method, we are keeping track only of the approximate eigenvectors. The new space shall store the corrector values as well as the approximate eigenvectors.

If we do such a thing, we can write more efficient algorithms to essentially "point" the eigenvectors in the correct direction. We can also express the next iteration (or approximation) of eigenvalues as a linear combination of the current approximation as well as the correction factor. This shall speeden up the convergence greatly.

Q5) d)

The balskin condition states that : for an approximate eigenpair $(\tilde{x}_i^{(1)}, \varepsilon_i^{(1)})$, the residual must be orthogonal to the entire subspace.

$$(\tilde{V}_{2n}^{(0)})^T (\tilde{A} \tilde{x}_i^{(1)} - \varepsilon_i^{(1)} \tilde{x}_i^{(1)}) = 0$$

Let us construct a matrix \tilde{U} s.t. \tilde{U} forms an orthonormal basis for $\tilde{V}_{2n}^{(0)}$. This can be done by any one of the orthogonalisation methods. In the interest of stability, we may like to use Householder orthogonalisation.

Vary the tolerance condition, we can yield:

$$\underline{H} \underline{y}_i = \tilde{\epsilon}_i^{(1)} \underline{y}_i, \text{ where } \underline{H} = \underline{V}^T \underline{A} \underline{V}$$

The Ritz pair will be (μ, \underline{v}_y)

The consequence of performing this iteration is that we reduce the problem from a large $m \times m$ eigenvalue problem to a relatively small $2n \times 2n$ eigenvalue problem.

The procedure to obtain the next approximation is as follows:

- i) Construct the orthonormal basis for $\underline{V}_{2n}^{(0)}$. Construct a matrix \underline{V} with these orthonormal basis vectors as its columns.
- ii) Form the matrix $\underline{H} = \underline{V}^T \underline{A} \underline{V}$.
- iii) Solve the "small" problem $\underline{H} \underline{y}_i = \tilde{\epsilon}_i^{(1)} \underline{y}_i$ to obtain an eigenpair.
- iv) Select the n smallest eigenvalues (and their corresponding eigenvectors). This shall represent the next iteration.
- v) Map these eigenvectors back to the m -dimensional space, so that we can start the next iteration. This can be done by:

$$\underline{x}_i^{(1)} = \underline{Q} \underline{y}_i$$

Q6) a) The characteristic equation is given by $\det(\underline{A} - \lambda \underline{I}) = 0$

$$\det \left[\begin{bmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 2 & 3 & 3 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right] = 0$$

$$\det \begin{bmatrix} 1-\lambda & 0 & 0 \\ 1 & 2-\lambda & 0 \\ 2 & 3 & 3-\lambda \end{bmatrix} = 0$$

$$(1-\lambda) \cdot \det \begin{bmatrix} 2-\lambda & 0 \\ 3 & 3-\lambda \end{bmatrix} = 0$$

$$-0 \cdot \det \begin{bmatrix} 1 & 0 \\ 2 & 3-\lambda \end{bmatrix} + 0 \cdot \det \begin{bmatrix} 1 & 2-\lambda \\ 2 & 3 \end{bmatrix}$$

$$(1-\lambda) \cdot \det \begin{bmatrix} 2-\lambda & 0 \\ 3 & 3-\lambda \end{bmatrix} = 0$$

$$(1-\lambda) [(2-\lambda)(3-\lambda) - 0(3)] = 0$$

$$(1-\lambda)(2-\lambda)(3-\lambda) = 0$$

$$\lambda = 1, 2, 3,$$

Q6) b) For this question, we need to find an orthonormal basis $\{\underline{q}_1, \underline{q}_2\}$ for the 2-D Krylov subspace $K_2 = \text{span}\{\underline{b}, \underline{A}\underline{b}\}$, where:

$$\underline{b} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\text{Computation of } \underline{\underline{q}}_1 : \quad \underline{\underline{q}}_1 = \frac{\underline{\underline{b}}}{\|\underline{\underline{b}}\|} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Computation of $\underline{\underline{q}}_2$:-

$$\text{Compute } \underline{\underline{A}} \underline{\underline{q}}_1 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 2 & 3 & 3 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}$$

Orthogonalise $\underline{\underline{A}} \underline{\underline{q}}_1$ from $\underline{\underline{q}}_1$:

$$\underline{\underline{v}}_2 = \underline{\underline{A}} \underline{\underline{q}}_1 - \underline{\underline{q}}_1^T (\underline{\underline{A}} \underline{\underline{q}}_1) \underline{\underline{q}}_1$$

$$= \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} - (1 \ 0 \ 0) \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix} - \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix}$$

Normalize $\underline{\underline{v}}_2$ to get $\underline{\underline{q}}_2$:

$$\underline{\underline{q}}_2 = \frac{\underline{\underline{v}}_2}{\|\underline{\underline{v}}_2\|} = \frac{1}{\sqrt{1^2+2^2}} \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} = \frac{1}{\sqrt{5}} \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} = \begin{bmatrix} 0 \\ \frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{bmatrix}$$

Hence, the basis vectors for K_2 are: $\underline{\underline{q}}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$ and $\underline{\underline{q}}_2 = \begin{bmatrix} 0 \\ \frac{1}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} \end{bmatrix}$

Q6) c) To project \underline{A} onto K , a 2×2 projection matrix \underline{H} shall be used.

From Q6)b), we calculated $h_{11} = 1$ and $h_{21} = \sqrt{5}$.

Let us calculate h_{12} :

$$\begin{aligned} h_{12} &= \underbrace{\underline{q}_1^T}_{\sim} \underbrace{\underline{A} \underline{q}_2}_{\sim} = \frac{1}{\sqrt{5}} (1 \ 0 \ 0) \begin{pmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 2 & 3 & 3 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} \\ &= \frac{1}{\sqrt{5}} (1 \ 0 \ 0) \begin{pmatrix} 0 \\ 2 \\ 9 \end{pmatrix} \\ &= \frac{1}{\sqrt{5}} (0) \\ &= 0 \end{aligned}$$

Let us calculate h_{22} :

$$\begin{aligned} h_{22} &= \underbrace{\underline{q}_2^T}_{\sim} \underbrace{\underline{A} \underline{q}_2}_{\sim} = \frac{1}{\sqrt{5}} \cdot \frac{1}{\sqrt{5}} \cdot (0 \ 1 \ 2) \begin{pmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 2 & 3 & 3 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} \\ &= \frac{1}{5} (0 \ 1 \ 2) \begin{pmatrix} 0 \\ 2 \\ 9 \end{pmatrix} \\ &= 20/5 \\ &= 4 \end{aligned}$$

$$\text{Hence, } \underline{H} = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \sqrt{5} & 4 \end{pmatrix}$$

Because \underline{H} is a lower triangular matrix, the eigenvalues of \underline{H} are 1 & 4. These are the "Ritz" values



The error in the largest eigenvalue = $|1 - 1| = 0$

The error in the smallest eigenvalue = $|4 - 3| = 1$

Q6) d) Let us find the solution to $\tilde{A}\tilde{x}^* = \tilde{b}$, $\tilde{x}^* \in \mathbb{R}^3$

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 2 & 3 & 3 \end{pmatrix} \tilde{x}^* = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \Rightarrow \tilde{x}^* = \begin{pmatrix} 1 \\ -\frac{1}{2} \\ -\frac{1}{6} \end{pmatrix}$$

Now, we want to find a vector \hat{x} such that $\|\tilde{A}\hat{x} - \tilde{b}\|$ is minimal, where $\hat{x} \in K_2$. We cannot solve this as a simple least squares problem due to the restriction on \hat{x} .

However, we can get a solution: $\hat{x} = Q\hat{y}$, where:

$$Q = \begin{bmatrix} 1 & 1 \\ q_1 & q_2 \\ q_3 & q_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\sqrt{5}} \\ 0 & \frac{2}{\sqrt{5}} \end{bmatrix} \text{ and } \hat{y} \text{ is the solution to } \tilde{A}\hat{y} = \|\tilde{b}\|_2 e_1.$$

Let us obtain \hat{y} :

$$\begin{pmatrix} 1 & 0 \\ \sqrt{5} & 4 \end{pmatrix} \hat{y} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \Rightarrow \hat{y} = \begin{pmatrix} 1 \\ -\frac{\sqrt{5}}{4} \end{pmatrix}$$

Hence, let us obtain \hat{x} :

$$\hat{x} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\sqrt{5}} \\ 0 & \frac{2}{\sqrt{5}} \end{bmatrix} \begin{pmatrix} 1 \\ -\frac{\sqrt{5}}{4} \end{pmatrix} = \begin{bmatrix} 1(1) + 0(-\frac{\sqrt{5}}{4}) \\ 0(1) + \frac{1}{\sqrt{5}}(-\frac{\sqrt{5}}{4}) \\ 0(1) + \frac{2}{\sqrt{5}}(\frac{\sqrt{5}}{4}) \end{bmatrix} = \begin{bmatrix} 1 \\ -\frac{1}{4} \\ -\frac{1}{2} \end{bmatrix}$$

Finally, let's calculate the error in $\underline{\underline{x}}^*$ and $\underline{\underline{x}}$:

$$\text{Error} = \|\underline{\underline{x}}^* - \underline{\underline{x}}\|_2$$

$$= \left\| \begin{pmatrix} 1 \\ -\frac{1}{2} \\ -\frac{1}{6} \end{pmatrix} - \begin{pmatrix} 1 \\ -\frac{1}{4} \\ -\frac{1}{2} \end{pmatrix} \right\|_2$$

$$= \left\| \begin{pmatrix} 0 \\ -\frac{1}{4} \\ \frac{1}{3} \end{pmatrix} \right\|$$

$$= \sqrt{(-\frac{1}{4})^2 + (\frac{1}{3})^2}$$

$$= \sqrt{\frac{1}{16} + \frac{1}{9}}$$

$$= \sqrt{\frac{25}{144}}$$

$$= \frac{5}{12}$$