RESEARCH UNIVERSITY
EXCELLENCE INITIATIVE



Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie

AGH University of Krakow

Numerical Methods in Physics

Project 05

Matrix Diagonalization Using the Power Method and Gram–Schmidt Orthogonalization

Hamzeh Khanpour

hamzeh.khanpour@fis.agh.edu.pl

Faculty of Physics and Applied Computer Science
AGH University of Krakow

Two questions

How do we calculate the eigenvalues and eigenvectors of a real symmetric matrix in C++ using the power method and Gram–Schmidt orthogonalization?

How do we verify the accuracy of the computed eigenpairs and reconstruct the original matrix and its inverse?

Scientific workflow

Physical model $\longrightarrow$ discretize $\longrightarrow$ power iteration $\longrightarrow$ orthogonalize $\longrightarrow$ verify $\longrightarrow$ interpret.

Course sources

S01: Tomasz Chwiej, Numerical Methods in Physics course page.
S02: Laboratory handout: Diagonalization of matrix with power method and Gram–Schmidt orthogonalization.
S03: Lecture: Diagonalization with iterative methods.

Technical sources

Netlib: Power Method – iterative calculation of dominant eigenpairs.
Netlib: Templates for the Solution of Algebraic Eigenvalue Problems.

AGH

After reviewing the mathematics, we can

Explain $A\mathbf{x} = \lambda\mathbf{x}$; calculate dot products and norms; execute one power step; obtain eigenvalue estimates; remove vector projections with Gram–Schmidt.

Our assessed responsibility

We independently construct the assigned 6×6 matrix, determine its six eigenpairs, reconstruct its matrix and inverse, save numerical histories, and investigate the report questions.

Teaching examples versus assessed work

E01–E09 solve complete *small independent* exercises. They do not assemble and solve the whole $n = 6$ assignment.

AGH

From physical model to verified result

$$f''(x) \quad \longrightarrow \quad A \quad \longrightarrow \quad (\lambda_k, \mathbf{q}_k) \quad \longrightarrow \quad A_{\text{num}}, A^{-1}_{\text{num}}.$$

We verify at each stage

Check the stencil coefficients first; compare hand-computable toy outputs with C++; inspect eigenpair residuals and orthogonality; check whether reconstructed matrices match known identities.

Recommended working rhythm

Predict a tiny numerical result on paper → compile one example → compare → implement the next part of our own program.

What is a matrix?

A matrix is a rectangular arrangement of numbers. In numerical physics, we often use a matrix to represent a mathematical operation acting on a vector.

For example, $A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$, $\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$.

How do we multiply a matrix by a vector?

We multiply each row of the matrix by the vector and add the corresponding terms:

$$A\mathbf{x} = \begin{pmatrix} 2x_1 + x_2 \\ x_1 + 2x_2 \end{pmatrix}.$$

For $\mathbf{x} = (1, 0)^T$, we obtain $A\mathbf{x} = \begin{pmatrix} 2(1) + 1(0) \\ 1(1) + 2(0) \end{pmatrix} = \boxed{\begin{pmatrix} 2 \\ 1 \end{pmatrix}}$.

Why is this important?

The power method repeatedly multiplies a matrix by a vector. Therefore, matrix-vector multiplication is the first mathematical operation we must understand.

What is an eigenvector?

Usually, multiplying a matrix by a vector changes its direction and length.
However, some special nonzero vectors keep their direction (up to a possible reversal). These are called **eigenvectors**. Mathematically, $\boxed{A\mathbf{x} = \lambda\mathbf{x}}$, where $\mathbf{x}$ is an eigenvector and λ is its corresponding eigenvalue.

A simple numerical example

Consider the same matrix and a different vector: $A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$, $\mathbf{x} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$.

Multiplying gives

$$A\mathbf{x} = \begin{pmatrix} 2(1) + 1(1) \\ 1(1) + 2(1) \end{pmatrix} = \begin{pmatrix} 3 \\ 3 \end{pmatrix} = 3 \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Therefore, $\boxed{\lambda = 3, \qquad \mathbf{x} = (1, 1)^T}$.

Connection to our laboratory

Our goal is to calculate these special numbers and vectors numerically. We use the power method to find eigenpairs and Gram-Schmidt orthogonalization to help us find different eigenvectors.

AGH

The mathematical question is the same

For a real symmetric matrix, find scalars λ_k and nonzero vectors $\mathbf{q}_k$ satisfying $A\mathbf{q}_k = \lambda_k \mathbf{q}_k$.

The numerical strategy is different

Direct diagonalization transforms or factorizes a matrix. The **power method** repeatedly multiplies a trial vector by A; Gram–Schmidt separates the additional eigenvectors.

Important limitation

An iterative method is not automatically faster or more accurate than a direct solver. Here we study its algorithm and convergence.

AGH

Where eigenvalues occur

Normal-mode frequencies, quantum-state energies, and discretized differential operators all lead to eigenvalue problems. Their eigenvectors describe modes or states.

Our simplified operator

$$\frac{d^2 f}{dx^2} \quad \longrightarrow \quad A\mathbf{f}, \qquad A\mathbf{q}_k = \lambda_k \mathbf{q}_k.$$

Sign convention

The assignment uses a discretization of $+d^2/dx^2$, with negative eigenvalues. The operator $-d^2/dx^2$ would have the opposite sign.

AGH

Taylor-expand around a point x_i

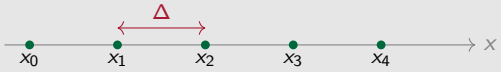
$$f(x_i \pm \Delta) = f_i \pm \Delta f_i' + \frac{\Delta^2}{2} f_i'' \pm \frac{\Delta^3}{6} f_i^{(3)} + O(\Delta^4).$$

Add the two expansions

The odd terms cancel, so

$$\boxed{f_i'' \approx \frac{f_{i-1} - 2f_i + f_{i+1}}{\Delta^2}} \qquad \text{error } O(\Delta^2).$$

Grid sketch



Three unknown interior values

Let $\mathbf{f} = (f_0, f_1, f_2)^T$. To close the centered difference, prescribe zero *outside* values $f_{-1} = f_3 = 0$.

For $\Delta = 1$

$$T_3 = \begin{pmatrix} -2 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -2 \end{pmatrix}.$$

Why this boundary convention?

At the two ends there is only one interior neighbor. Our stored endpoint values f_0, f_2 are still unknowns.

AGH

The nonzero entries are local

$$a_{ij} = \begin{cases} -2, & i = j, \\ 1, & |i-j| = 1, \\ 0, & \text{otherwise.} \end{cases}$$

Interpret each row

The diagonal coefficient acts on the value at point i. The two adjacent coefficients act on neighboring grid points. The matrix is symmetric, $A^T = A$.

Implementation caution

Tridiagonal does not require CSR here. The handout explicitly asks for two-dimensional matrix storage.

Required assignment input

$$A_6 = \begin{pmatrix} -2 & 1 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 1 & -2 \end{pmatrix}.$$

Numerical settings

$n = 6$, $\Delta = 1$, $\mathrm{itmax} = 150$, $\varepsilon = 10^{-8}$. We must display the matrix to confirm its entries.

What not to copy

Our teaching examples use 2×2 and 3×3 matrices. We write the full 6×6 diagonalization driver ourselves.

Definition

$$\boxed{A\mathbf{x} = \lambda\mathbf{x}}, \qquad \mathbf{x} \neq \mathbf{0}.$$

The scalar λ is an eigenvalue. The nonzero vector $\mathbf{x}$ is a corresponding eigenvector.

Interpretation

Multiplying an eigenvector by A changes its scale (and possibly direction sign), not its line through the origin.

A direct numerical check

Compute both $A\mathbf{x}$ and $\lambda\mathbf{x}$: their componentwise difference should be zero for an exact eigenpair.

AGH

An eigenvector is a direction

If $A\mathbf{q} = \lambda\mathbf{q}$, then $A(c\mathbf{q}) = \lambda(c\mathbf{q})$ for every nonzero scalar c. Therefore an eigenvector has no unique length.

Choose unit length

$$\|\mathbf{q}\|_2 = 1, \qquad \mathbf{q} = \frac{\mathbf{x}}{\sqrt{\mathbf{x}^T\mathbf{x}}}.$$

The sign is not unique

$\mathbf{q}$ and $-\mathbf{q}$ represent the same eigenvector direction. Compare eigenvectors up to sign.

AGH

Our independent teaching matrix

$$B = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \quad \det(B - \lambda I) = \begin{vmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{vmatrix}.$$

Calculate the characteristic polynomial

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

Therefore $\boxed{\lambda_1 = 3,\ \lambda_2 = 1}$.

Why this example?

Its arithmetic fits on one line, so we can check every C++ step by hand.

AGH

For $\lambda_1 = 3$

$(B - 3I)\mathbf{x} = 0$ implies $x_1 = x_2$. Choose $\mathbf{q}_1 = (1, 1)^T/\sqrt{2}$.

For $\lambda_2 = 1$

$(B - I)\mathbf{x} = 0$ implies $x_1 = -x_2$. Choose $\mathbf{q}_2 = (1, -1)^T/\sqrt{2}$.

Substitute to verify

$$B\mathbf{q}_1 = 3\mathbf{q}_1, \qquad B\mathbf{q}_2 = 1\mathbf{q}_2.$$

AGH

Start from the same scalar in two ways

$$\mathbf{q}_i^T A\mathbf{q}_j = \lambda_j \mathbf{q}_i^T \mathbf{q}_j, \qquad (A\mathbf{q}_i)^T \mathbf{q}_j = \lambda_i \mathbf{q}_i^T \mathbf{q}_j.$$

Symmetry gives the conclusion

Because $A^T = A$, their left sides agree. Thus $(\lambda_i - \lambda_j)\mathbf{q}_i^T \mathbf{q}_j = 0$. For distinct eigenvalues, $\mathbf{q}_i^T \mathbf{q}_j = 0$.

Unit eigenvectors

For an orthonormal basis: $\boxed{\mathbf{q}_i^T \mathbf{q}_j = \delta_{ij}}$. Repeated eigenvalues require choosing an orthogonal basis.

Source file

`examples/01_eigenpairs_2x2.cpp`. We calculate both $B\mathbf{q} - \lambda\mathbf{q}$ residuals and the dot product of the two vectors.

Compile and run

```
make bin/01_eigenpairs_2x2  ./bin/01_eigenpairs_2x2
```

Expected: $\lambda_1 = 3$, $\lambda_2 = 1$, both residuals approximately zero;
`PASS E01`.

Our project connection

For every computed eigenpair we should be able to perform the same residual check.

Teaching link: E01

AGH

Build this small known matrix

$$T_3 = \begin{pmatrix} -2 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -2 \end{pmatrix}.$$

What to run

`make bin/02_tridiagonal_3x3  ./bin/02_tridiagonal_3x3`
Expected: all three rows, then $T_3(1,2,3)^T = (0,0,-4)^T$.

Project connection

We generalize the indexing idea to the assigned matrix, with $n = 6$ and two-dimensional storage.

Teaching link: E02

Mathematical versus C++ indices

For a C++ 2D array or nested vector, $A[i][j]$ denotes the matrix entry in row i and column j, with indices starting at zero.

A safe conceptual rule

Set $A[i][i] = -2$; set adjacent entries to 1 only when the index is within the matrix bounds; leave the rest zero.

Check before iterating

A visually incorrect first or last row will produce an entirely different spectrum. Print all rows and confirm $A_{ij} = A_{ji}$.

Teaching link: E02

AGH

Definition

$$\mathbf{u} \cdot \mathbf{w} = \sum_{j=0}^{n-1} u_j w_j.$$

Three-component numerical example

$$\mathbf{u} = (1, 2, 3), \quad \mathbf{w} = (4, -1, 2), \quad \mathbf{u} \cdot \mathbf{w} = 4 - 2 + 6 = \boxed{8}.$$

Why we need it

Dot products calculate lengths, Gram–Schmidt projections, Rayleigh quotients, and the eigenvector orthogonality checks.

Teaching link: E03

AGH

Euclidean norm

$$\|\mathbf{u}\|_2 = \sqrt{\sum_j u_j^2}, \qquad \|(1,2,3)\|_2 = \sqrt{14}.$$

Normalize without changing direction

$$\widehat{\mathbf{u}} = \frac{(1,2,3)}{\sqrt{14}}, \qquad \|\widehat{\mathbf{u}}\|_2 = 1.$$

Guard against division by zero

Before dividing, require a finite norm strictly larger than a suitable numerical threshold.

Teaching link: E03

Multiply each matrix row with the vector

$$\mathbf{y} = A\mathbf{x}, \qquad y_i = \sum_{j=0}^{n-1} a_{ij} x_j.$$

Use T_3 and $(1, 2, 3)^T$

$$y_0 = -2(1) + 1(2) = 0,$$
$$y_1 = 1(1) - 2(2) + 1(3) = 0,$$
$$y_2 = 1(2) - 2(3) = -4.$$

Expected result

$\boxed{\mathbf{y} = (0, 0, -4)^T}$. We must check every component, not only the vector norm.

Teaching link: E03

AGH

Three small, reusable operations

`dot(u,w)` evaluates a scalar product; `norm(u)` uses $\sqrt{\mathbf{u}\cdot\mathbf{u}}$; a nested loop evaluates $y_i = \sum_j A[i][j]x_j$.

Compile / run / verify

```
make bin/03_vector_matrix_operations
./bin/03_vector_matrix_operations
```

Expected: `dot=8`, `norm_u=3.741657...`, and $\mathbf{y} = (0, 0, -4)$.

Connection to the project

These elementary functions are prerequisites for our own power-method loop and numerical diagnostics.

Teaching link: E03

AGH

One multiplication can improve our guess

We start with a nonzero vector $\mathbf{x}^{(0)}$ and repeatedly apply A. Each step changes the relative components along eigenvector directions.

The central idea

$$\mathbf{x}^{(0)} \rightarrow A\mathbf{x}^{(0)} \rightarrow A^2\mathbf{x}^{(0)} \rightarrow \cdots$$

We need safeguards

Large vector magnitudes may overflow. We normalize each new vector and verify whether the result is approaching a genuine eigenpair.

Because A is real symmetric

We can choose an orthonormal eigenbasis $\{\mathbf{q}_j\}$ and write $\mathbf{x}^{(0)} = \sum_j c_j \mathbf{q}_j$.

After m repeated multiplications

$$\boxed{A^m \mathbf{x}^{(0)} = \sum_j c_j \lambda_j^m \mathbf{q}_j}.$$

Each eigenvector component is multiplied by its eigenvalue raised to the mth power.

Why this matters

Components associated with eigenvalues of larger *absolute magnitude* become dominant, provided their initial coefficients are nonzero.

Teaching link: E04–E05

Assume a unique dominant eigenvalue

Let $|\lambda_1| > |\lambda_2| \geq \cdots$ and $c_1 \neq 0$. Factor out λ_1^m.

Normalize the expansion

$$\frac{A^m \mathbf{x}^{(0)}}{\lambda_1^m} = c_1 \mathbf{q}_1 + \sum_{j>1} c_j \left(\frac{\lambda_j}{\lambda_1}\right)^m \mathbf{q}_j.$$

The unwanted components shrink

Because $|\lambda_j/\lambda_1| < 1$, the ratios decay with m. This predicts the limiting eigenvector direction.

Teaching link: E05

AGH

A negative eigenvalue reverses orientation

If $A\mathbf{q} = -3\mathbf{q}$, multiplication reverses the vector sign while changing its length by a factor of three.

What the algorithm sees

$\mathbf{q} \to -\mathbf{q} \to \mathbf{q}$ is not a failure: the eigenvector line is unchanged. Compare $|\mathbf{q}^T\mathbf{x}|$, not the raw componentwise sign.

Limit of the argument

If two eigenvalues have the same absolute magnitude, a generic initial vector need not settle to one eigenvector. Optional E10 demonstrates this.

Teaching link: E05 / E10

AGH

Start with our 2*imes*2 matrix

$$B = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \qquad \mathbf{x}^{(0)} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Multiply

$$\mathbf{y} = B\mathbf{x}^{(0)} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

The second component becomes nonzero even though it was initially zero.

Next operation

Before repeating, we normalize $\mathbf{y}$ to unit length; this prevents uncontrolled growth.

Teaching link: E04

www.agh.edu.pl

Hand calculation

$$\|\mathbf{y}\|_2 = \sqrt{2^2 + 1^2} = \sqrt{5}, \qquad \mathbf{x}^{(1)} = \frac{1}{\sqrt{5}}(2, 1)^T.$$

Run the teaching example

`make bin/04_one_power_step  ./bin/04_one_power_step`
Expected: `y0=2`, `y1=1`, `norm_y=sqrt(5)`; then `PASS E04`.

Project connection

The same multiply-and-normalize operation appears inside every iteration of our assessed implementation.

Teaching link: E04

AGH

Core algorithm for one eigenpair

$$\mathbf{y}^{(k)} = A\mathbf{x}^{(k)}, \qquad \mathbf{x}^{(k+1)} = \frac{\mathbf{y}^{(k)}}{\|\mathbf{y}^{(k)}\|_2}.$$

Mini-pseudocode for the independent 2*imes*2 toy

Initialize a nonzero vector; repeatedly multiply; reject a zero/nonfinite norm; normalize; estimate the eigenvalue and check the residual; stop at a justified criterion.

Important distinction

This ordinary power loop finds one dominant eigenpair. It is not yet the complete Gram–Schmidt multi-eigenpair project.

Teaching link: E05

AGH

Known starting point

$\mathbf{x}^{(0)} = (1, 0)^T$. Its initial Rayleigh quotient is 2; after one power step, the quotient becomes 2.8.

The next iterations

$$R(\mathbf{x}^{(0)}) = 2, \quad R(\mathbf{x}^{(1)}) = 2.8, \quad R(\mathbf{x}^{(2)}) \approx 2.975610.$$

The true dominant eigenvalue is 3.

Test against known truth

E05 writes real CSV rows and checks the eigenpair residual and alignment up to sign; run `make bin/05_power_method_2x2_history`.

Teaching link: E05

AGH

Start from an exact eigenpair

If $A\mathbf{q} = \lambda\mathbf{q}$, multiply on the left by $\mathbf{q}^T$:

$$\mathbf{q}^T A\mathbf{q} = \lambda \mathbf{q}^T \mathbf{q}.$$

Solve for the eigenvalue

$$\boxed{R(\mathbf{x}) = \frac{\mathbf{x}^T A\mathbf{x}}{\mathbf{x}^T \mathbf{x}}}.$$

For an approximate eigenvector $\mathbf{x}$, $R(\mathbf{x})$ estimates its eigenvalue.

Normalization is optional in this formula

For unit length, $\mathbf{x}^T\mathbf{x} = 1$ and the formula simplifies to $R(\mathbf{x}) = \mathbf{x}^T A\mathbf{x}$.

Teaching link: E06

AGH

Take the unnormalized vector $(2,1)^T$

$$B\begin{pmatrix}2\\1\end{pmatrix} = \begin{pmatrix}5\\4\end{pmatrix}.$$

Compute numerator and denominator separately

$$\mathbf{x}^T B\mathbf{x} = (2,1)\cdot(5,4) = 14, \quad \mathbf{x}^T\mathbf{x} = 5, \quad \boxed{R = \frac{14}{5} = 2.8}.$$

Compare with the initial guess

For $(1,0)^T$, $R = 2$; the new vector yields an estimate closer to the true eigenvalue 3.

Teaching link: E06

AGH

Define an independent error diagnostic

$$\mathbf{r} = A\mathbf{x} - R(\mathbf{x})\mathbf{x}, \qquad \rho = \|\mathbf{r}\|_2.$$

A small ρ means that this pair nearly satisfies the eigenvalue equation.

Exact eigenpair versus approximate eigenpair

If $A\mathbf{q} = \lambda\mathbf{q}$, then $\rho = 0$ (up to roundoff). An eigenvalue estimate may change very little while ρ remains large.

Report both quantities

We must distinguish a small change between consecutive Rayleigh quotients from an actually accurate eigenpair.

Teaching link: E06 / E10

Hand-check the normalized vector

For $\mathbf{q} = (2,1)^T/\sqrt{5}$, $R = 2.8$ and

$$B\mathbf{q} - 2.8\mathbf{q} = \frac{1}{\sqrt{5}}(-0.6, 1.2)^T.$$

Compute the norm

$$\rho = \sqrt{(0.6)^2 + (1.2)^2}/\sqrt{5} = \boxed{0.6}.$$

Run E06

`make bin/06_rayleigh_residual  ./bin/06_rayleigh_residual`
Expected: `rayleigh=2.8 residual=0.6`, then `PASS E06`.

Teaching link: E06

The official handout

Uses $\varepsilon = 10^{-8}$ and $\mathrm{itmax} = 150$, with a relative change in consecutive eigenvalue estimates (when the previous estimate is safely nonzero).

A stronger diagnostic

$$\delta_k = \frac{|\lambda_k - \lambda_{k-1}|}{|\lambda_{k-1}|}, \qquad \rho_k = \|A\mathbf{x}^{(k)} - \lambda_k \mathbf{x}^{(k)}\|_2.$$

Safe program behavior

Handle an undefined initial value, small denominator, zero projected norm, nonfinite numbers and iteration-limit termination. Do not label `itmax` as automatic convergence.

Teaching link: E05 / E10

The same dominant component reappears

If we restart from another generic vector, its component parallel to the dominant eigenvector will again grow relative to the others.

What we need

We must remove contributions parallel to the eigenvectors already discovered. This is a problem of vector *projection and orthogonalization*.

Next topic

Gram–Schmidt provides a simple, mathematically transparent way to subtract those contributions during every iteration.

Teaching link: E07–E08

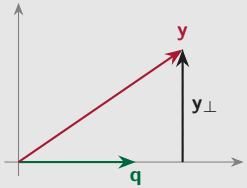
Project a vector onto a known unit direction

For normalized $\mathbf{q}$, the parallel component of $\mathbf{y}$ is $(\mathbf{q}^T\mathbf{y})\mathbf{q}$.

Remove that component

$$\boxed{\mathbf{y}_\perp = \mathbf{y} - (\mathbf{q}^T\mathbf{y})\mathbf{q}}, \qquad \mathbf{q}^T\mathbf{y}_\perp = 0.$$

Geometry



Teaching link: E07

AGH

Use exactly computable values

$$\mathbf{q}_1 = \frac{1}{\sqrt{2}}(1,1)^T, \quad \mathbf{y} = (2,0)^T, \quad c = \mathbf{q}_1^T \mathbf{y} = \sqrt{2}.$$

Subtract the projection

$$\mathbf{z} = \mathbf{y} - c\mathbf{q}_1 = (2,0)^T - (1,1)^T = \boxed{(1,-1)^T}.$$

The new vector is perpendicular

$\mathbf{q}_1^T \mathbf{z} = (1-1)/\sqrt{2} = 0$. Normalizing $\mathbf{z}$ yields $\mathbf{q}_2 = (1,-1)^T/\sqrt{2}$.

Teaching link: E07

Suppose we already found k unit eigenvectors

$\mathbf{q}_0, \ldots, \mathbf{q}_{k-1}$ are orthonormal. To seek a new direction, remove its components along every known vector.

General projector

$$\mathbf{y} \leftarrow \mathbf{y} - \sum_{m=0}^{k-1} (\mathbf{q}_m^T \mathbf{y}) \mathbf{q}_m .$$

Index discipline

The sum uses *previous eigenvector index* m, not the power-iteration counter. For $k = 0$, the sum is empty.

Teaching link: E07–E08

AGH

The projector in matrix form

$$P_k = I - \sum_{m=0}^{k-1} \mathbf{q}_m \mathbf{q}_m^T, \qquad \mathbf{y}_\perp = P_k \mathbf{y}.$$

Why use it in an iterative eigensolver?

Without projection, the largest-magnitude eigenpair already found can dominate again. Projection forces our trial vector to remain in the orthogonal complement.

Implementation and stability

Apply this during *each iteration*. In floating-point arithmetic, a second orthogonalization pass may improve accuracy, but it is an optional safeguard.

Teaching link: E08

Expected exact arithmetic

$$\mathbf{y} = (2, 0), \quad \mathbf{y}_\perp = (1, -1), \quad \mathbf{q}_2 = (1, -1)/\sqrt{2}.$$

Compile / run

```
make bin/07_gram_schmidt_2d
./bin/07_gram_schmidt_2d
```

Expected: `z0=1`, `z1=-1`, `orth=0`, and `PASS E07`.

Connection

We use this operation to remove previously determined eigenvector components in the assessed program.

Teaching link: E07

AGH

Annotated mathematical skeleton

$$\mathbf{y} \leftarrow A\mathbf{x}; \quad \mathbf{y} \leftarrow \mathbf{y} - \sum_{m<k} (\mathbf{q}_m^T \mathbf{y})\mathbf{q}_m;$$
$$\mathbf{x} \leftarrow \mathbf{y}/\|\mathbf{y}\|_2; \quad \lambda \leftarrow \mathbf{x}^T A\mathbf{x}.$$

Order matters

The Rayleigh quotient at the end belongs to the *current projected and normalized* vector. This is a recommended consistency improvement over quotient-before-projection ordering in the handout.

We still need the project loop

The official $n = 6$ code must initialize new trial vectors, save pairs and convergence histories, and implement safe stopping. Students build that integration themselves.

Teaching link: E08

AGH

Floating-point vectors are never perfectly orthogonal

Rounding may leave a tiny component along a previously determined dominant eigenvector. After multiplication by A, it can grow again.

Fix the problem where it arises

After *each* matrix–vector product, subtract the known-direction contributions before normalizing and estimating the current Rayleigh quotient.

Additional safeguard

Modified Gram–Schmidt and reorthogonalization can reduce drift. Check $|\mathbf{q}_m^T \mathbf{x}|$ to detect loss of orthogonality.

Teaching link: E08

AGH

Known toy 3*imes*3 matrix

$$C = \begin{pmatrix} 3 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{q}_1 = \frac{1}{\sqrt{2}}(1,1,0)^T, \quad \lambda_1 = 4.$$

Remove that first direction every time

With a suitable start, iterates in the orthogonal complement approach $\mathbf{q}_2 = (1,-1,0)^T/\sqrt{2}$, whose eigenvalue is $\lambda_2 = 2$.

Run it

```
make bin/08_projected_power_toy  ./bin/08_projected_power_toy.
```

Teaching link: E08

AGH

Know the expected outcome

$$C\mathbf{q}_2 = 2\mathbf{q}_2, \quad \mathbf{q}_1^T\mathbf{q}_2 = 0.$$

The program reports a quotient near 2 and a small eigenpair residual.

Three checks to inspect

Eigenvalue error $|\lambda - 2|$; eigenpair residual $\|C\mathbf{x} - \lambda\mathbf{x}\|_2$; orthogonality $|\mathbf{q}_1^T\mathbf{x}|$.

The independent-work boundary

E08 is a *fixed* teaching toy with a known first eigenvector, not a general six-eigenpair eigensolver.

Teaching link: E08

www.agh.edu.pl

AGH

Collect normalized eigenvectors into Q

$$Q = (\mathbf{q}_0\ \mathbf{q}_1\ \cdots), \qquad Q^T Q = I, \quad \Lambda = \mathrm{diag}(\lambda_0, \lambda_1, \ldots).$$

Matrix diagonalization

$$AQ = Q\Lambda \quad \Longrightarrow \quad \boxed{A = Q\Lambda Q^T}.$$

Why we care

This representation lets us reconstruct the original matrix and, if every eigenvalue is nonzero, calculate its inverse.

Teaching link: E09

AGH

Two eigenpairs of our small example

$\lambda_1 = 3$, $\mathbf{q}_1 = (1,1)/\sqrt{2}$; $\lambda_2 = 1$, $\mathbf{q}_2 = (1,-1)/\sqrt{2}$.

Add the outer products

$$B = 3\mathbf{q}_1\mathbf{q}_1^T + 1\mathbf{q}_2\mathbf{q}_2^T$$
$$= \tfrac{3}{2}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} + \tfrac{1}{2}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} = \boxed{\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}}.$$

One sum per matrix entry

$A_{ij} = \sum_k \lambda_k (q_k)_i (q_k)_j$. For the project we use all the eigenpairs we calculate ourselves.

Teaching link: E09

AGH

Invert the diagonal matrix, not the entries of A

If $\lambda_k \neq 0$ for every k, then $A^{-1} = Q\Lambda^{-1}Q^T$, with $\Lambda^{-1}_{kk} = 1/\lambda_k$.

The explicit spectral formula

$$\boxed{A^{-1} = \sum_k \frac{1}{\lambda_k}\mathbf{q}_k\mathbf{q}_k^T}.$$

Special conditions

This formula needs a complete orthonormal eigenbasis and nonzero eigenvalues. The inverse of a singular matrix does not exist.

Teaching link: E09

AGH

Exact inverse of the toy matrix

$$B^{-1} = \frac{1}{3}\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

Verify multiplication

$$BB^{-1} = \frac{1}{3}\begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix} = I.$$

Compile / run

`make bin/09_spectral_reconstruction_2x2`
Expect reconstruction and identity errors approximately zero, then `PASS E09`.

Teaching link: E09

AGH

Eigenpairs, norm, orthogonality

$$\rho_k = \|A\mathbf{q}_k - \lambda_k \mathbf{q}_k\|_2, \qquad \|Q^T Q - I\|_F.$$

Reconstruction and inverse

$$\|A - A_{\text{num}}\|_F, \qquad \|AA_{\text{num}}^{-1} - I\|_F.$$

Interpret these numbers

Errors should be small relative to the matrix scale and the requested tolerance; no single test alone proves every part of the implementation correct.

AGH

Milestone 1: construct the exact assigned A_6

$A_{ii} = -2$, $A_{i,i\pm1} = 1$ (within bounds), all remaining entries zero, using a 2D matrix representation. Display all six rows.

Milestone 2: implement basic operations

Dot product, Euclidean norm, dense matrix–vector product, vector normalization, and Gram–Schmidt projection. Test each using E02–E07 sized inputs.

AGH

Milestone 3: implement the full iterative sequence

With $n = 6$, $\mathrm{itmax} = 150$ and $\varepsilon = 10^{-8}$, find consecutive eigenvectors while projecting away all previously determined ones in every iteration.

Output requirements

Write successive estimates of *each* eigenvalue to a separate file; identify the eigenpair index and iteration number so histories can later be plotted together.

Log honest termination status

Record whether tolerance was met or the iteration limit was reached. Relative eigenvalue change alone does not guarantee small eigenpair residual.

AGH

Milestone 4: save the reconstructed original matrix

$$A_{\text{num}} = \sum_{k=0}^{5} \lambda_k \mathbf{q}_k \mathbf{q}_k^T .$$

Save the inverse matrix too

$$A_{\text{num}}^{-1} = \sum_{k=0}^{5} \lambda_k^{-1} \mathbf{q}_k \mathbf{q}_k^T .$$

Verification output

Compare entries of A_{num} with A_6, and check $AA_{\text{num}}^{-1} \approx I$. Files and formatting should make entries straightforward to inspect.

AGH

First report figure

Plot the iteration histories of all six eigenvalue estimates together, with meaningful legend labels, iteration counters, and eigenvalue axis.

Discuss numerical sensitivity

Compare convergence by eigenvalue index. Reconstruct A with $\varepsilon = 10^{-8}$ and again with $\varepsilon = 10^{-4}$; explain changes in its entries.

Prove inverse symmetry

Because $A^T = A$ and A is invertible, $(A^{-1})^T = (A^T)^{-1} = A^{-1}$. Explain why numerical errors can cause small departures.

AGH

Larger matrix experiment

Find ten leading eigenvalues for $n = 200$. Explain that the ordinary power method selects leading eigenvalues by *magnitude*, and can converge slowly when those magnitudes are close.

Disable orthogonalization

Repeat without Gram–Schmidt. Can we obtain ten distinct eigenvectors? Explain the repeated domination by the largest-magnitude eigenpair.

Final checklist and sources

Use our own project data, save convergence histories and both matrices, report actual stopping status.
Official assignment Official theory lecture.


THANK
YOU!