PMAI LAB

Linear Algebra: Theory, Computation, and Modern Applications

Lecture 06 • Inner Products, Orthogonality, and Projections

Course lens
Understand → Model → Compute → Apply → Code → Interpret

Audience
Undergraduate BS Mathematics students

Instructor
Dr. Hira Benish

Inner products measure alignment; orthogonality separates directions; projections extract the part of a vector explained by a chosen subspace.

1. Inner Products: Measuring Alignment

From multiplication of components to length, distance, and angle

For vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$, the standard inner product is

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \boldsymbol{u}^\top \boldsymbol{v} = u_1 v_1 + u_2 v_2 + \cdots + u_n v_n.$$

In Euclidean space this is also called the dot product.

Learning outcomes

By the end of this lecture, students should be able to:

- calculate inner products, norms, distances, and angles;
- determine whether vectors or subspaces are orthogonal;
- construct and use orthonormal sets;
- apply the Gram-Schmidt process to a small set of vectors;
- project a vector onto a vector or subspace;
- interpret projection matrices and orthogonal residuals; and
- solve a document-similarity problem using cosine similarity in Python.

Core properties

For vectors $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}$ and scalar c:

$$\begin{aligned} \langle \boldsymbol{u}, \boldsymbol{v} \rangle &= \langle \boldsymbol{v}, \boldsymbol{u} \rangle, \\ \langle \boldsymbol{u} + \boldsymbol{w}, \boldsymbol{v} \rangle &= \langle \boldsymbol{u}, \boldsymbol{v} \rangle + \langle \boldsymbol{w}, \boldsymbol{v} \rangle, \\ \langle c\boldsymbol{u}, \boldsymbol{v} \rangle &= c \langle \boldsymbol{u}, \boldsymbol{v} \rangle, \\ \langle \boldsymbol{u}, \boldsymbol{u} \rangle &\geq 0, \quad \text{with equality only when } \boldsymbol{u} = \boldsymbol{0}. \end{aligned}$$

Norm, distance, and angle

$$\|\boldsymbol{u}\| = \sqrt{\langle \boldsymbol{u}, \boldsymbol{u} \rangle}, \qquad d(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|,$$

and for nonzero vectors,

$$\cos\theta = \frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{\|\boldsymbol{u}\| \|\boldsymbol{v}\|}.$$

Interpretation of the sign

A positive inner product indicates broadly similar direction, zero indicates perpendicular directions, and a negative value indicates broadly opposing direction. Magnitude and scaling still matter, so cosine similarity is often used when direction is more important than size.

2. Orthogonality and Geometric Structure

Perpendicular directions contain no component along one another

Two vectors are **orthogonal** if

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = 0.$$

For example,

$$\boldsymbol{u} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \qquad \boldsymbol{v} = \begin{pmatrix} 1 \\ -2 \end{pmatrix}, \qquad \langle \boldsymbol{u}, \boldsymbol{v} \rangle = 2 - 2 = 0.$$

Pythagorean theorem

If $\boldsymbol{u} \perp \boldsymbol{v}$, then

$$\|\boldsymbol{u} + \boldsymbol{v}\|^2 = \|\boldsymbol{u}\|^2 + \|\boldsymbol{v}\|^2.$$

For the vectors above,

$$\|\boldsymbol{u}\|^2 = 5, \quad \|\boldsymbol{v}\|^2 = 5, \quad \|\boldsymbol{u} + \boldsymbol{v}\|^2 = \left\| \begin{pmatrix} 3 \\ -1 \end{pmatrix} \right\|^2 = 10.$$

Orthogonal and orthonormal sets

A set $\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ is orthogonal if every pair of distinct vectors has inner product zero. It is **orthonormal** if, in addition,

$$\|\boldsymbol{v}_i\| = 1 \quad \text{for every } i.$$

Every orthogonal set of nonzero vectors is linearly independent. To normalize a nonzero vector, use

$$\widehat{\boldsymbol{v}} = \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}.$$

Orthogonal complements

For a subspace $W \subseteq \mathbb{R}^n$,

$$W^{\perp} = \{\boldsymbol{x} \in \mathbb{R}^n : \langle \boldsymbol{x}, \boldsymbol{w} \rangle = 0 \text{ for every } \boldsymbol{w} \in W\}.$$

Every vector $\boldsymbol{y} \in \mathbb{R}^n$ can be decomposed uniquely as

$$\boldsymbol{y} = \boldsymbol{y}_W + \boldsymbol{y}_{W^\perp},$$

where $\boldsymbol{y}_W \in W$ and $\boldsymbol{y}_{W^\perp} \in W^\perp$.

Why orthogonality is useful
Orthogonal directions do not interfere with one another. This simplifies coordinates, separates signals, supports stable computation, and makes projection errors geometrically interpretable.

3. Orthonormal Bases and Gram-Schmidt

Convert an independent set into clean perpendicular directions

If $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_k\}$ is an orthonormal basis for a subspace W, then every $\boldsymbol{y} \in W$ has the simple expansion

$$\boldsymbol{y} = \langle \boldsymbol{y}, \boldsymbol{e}_1 \rangle \boldsymbol{e}_1 + \cdots + \langle \boldsymbol{y}, \boldsymbol{e}_k \rangle \boldsymbol{e}_k.$$

The coefficients are obtained directly through inner products.

Gram-Schmidt process

Given independent vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$, define

$$\boldsymbol{u}_1 = \boldsymbol{v}_1,$$

and remove from each later vector the components already explained by earlier orthogonal directions:

$$\boldsymbol{u}_j = \boldsymbol{v}_j - \sum_{i=1}^{j-1} \frac{\langle \boldsymbol{v}_j, \boldsymbol{u}_i \rangle}{\langle \boldsymbol{u}_i, \boldsymbol{u}_i \rangle} \boldsymbol{u}_i.$$

Finally normalize: $\boldsymbol{e}_i = \boldsymbol{u}_i / \|\boldsymbol{u}_i\|$.

Worked example in $\mathbb{R}^3$

Let

$$\boldsymbol{v}_1 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \qquad \boldsymbol{v}_2 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}.$$

First,

$$\boldsymbol{u}_1 = \boldsymbol{v}_1, \qquad \boldsymbol{e}_1 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}.$$

Next,

$$\boldsymbol{u}_2 = \boldsymbol{v}_2 - \frac{\langle \boldsymbol{v}_2, \boldsymbol{u}_1 \rangle}{\langle \boldsymbol{u}_1, \boldsymbol{u}_1 \rangle} \boldsymbol{u}_1 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} - \frac{1}{2} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1/2 \\ -1/2 \\ 1 \end{pmatrix}.$$

Therefore

$$\boldsymbol{e}_2 = \frac{1}{\sqrt{6}} \begin{pmatrix} 1 \\ -1 \\ 2 \end{pmatrix}.$$

The vectors $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ are orthonormal and span the same plane as $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$.

> **Computational connection**
> Gram-Schmidt leads to QR factorization $\mathbf{A} = \mathbf{QR}$, where the columns of $\mathbf{Q}$ are orthonormal. QR methods are central to least squares and numerical linear algebra.

4. Projection onto a Vector

Extract the component of one vector along a chosen direction

For a nonzero vector $\boldsymbol{a}$, the projection of $\boldsymbol{b}$ onto $\boldsymbol{a}$ is

$$\operatorname{proj}_{\boldsymbol{a}}(\boldsymbol{b}) = \frac{\langle \boldsymbol{b}, \boldsymbol{a} \rangle}{\langle \boldsymbol{a}, \boldsymbol{a} \rangle} \boldsymbol{a}.$$

The orthogonal residual is

$$\boldsymbol{r} = \boldsymbol{b} - \operatorname{proj}_{\boldsymbol{a}}(\boldsymbol{b}), \qquad \langle \boldsymbol{r}, \boldsymbol{a} \rangle = 0.$$

Worked example

Let

$$\boldsymbol{a} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \qquad \boldsymbol{b} = \begin{pmatrix} 3 \\ 1 \end{pmatrix}.$$

The projection coefficient is

$$\frac{\langle \boldsymbol{b}, \boldsymbol{a} \rangle}{\langle \boldsymbol{a}, \boldsymbol{a} \rangle} = \frac{3+2}{1+4} = 1.$$

Hence

$$\operatorname{proj}_{\boldsymbol{a}}(\boldsymbol{b}) = \begin{pmatrix} 1 \\ 2 \end{pmatrix}, \qquad \boldsymbol{r} = \begin{pmatrix} 3 \\ 1 \end{pmatrix} - \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \end{pmatrix}.$$

Verification:

$$\left\langle \begin{pmatrix} 2 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \end{pmatrix} \right\rangle = 2 - 2 = 0.$$

Projection matrix

Writing $\boldsymbol{a}$ as a column vector,

$$\mathbf{P} = \frac{\boldsymbol{a}\boldsymbol{a}^{\mathsf{T}}}{\boldsymbol{a}^{\mathsf{T}}\boldsymbol{a}} = \frac{1}{5}\begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}, \qquad \operatorname{proj}_{\boldsymbol{a}}(\boldsymbol{b}) = \mathbf{P}\boldsymbol{b}.$$

Projection matrices satisfy

$$\mathbf{P}^{\mathsf{T}} = \mathbf{P}, \qquad \mathbf{P}^2 = \mathbf{P}.$$

The second property means that projecting an already projected vector does not change it.

Closest-vector interpretation
Among all vectors on the line $\operatorname{span}\{\boldsymbol{a}\}$, the projection is the one closest to $\boldsymbol{b}$. The residual gives the minimum distance from $\boldsymbol{b}$ to that line.

5. Projection onto a Subspace

Approximate a vector using several independent directions

Let the columns of **A** form a basis for a subspace $W \subseteq \mathbb{R}^m$. The projection of b onto W has the form

$$\widehat{b} = \mathbf{A}\widehat{x},$$

and the residual $b - \mathbf{A}\widehat{x}$ must be orthogonal to every column of **A**. Therefore

$$\mathbf{A}^\top(b - \mathbf{A}\widehat{x}) = \mathbf{0},$$

which gives the normal equations

$$\mathbf{A}^\top\mathbf{A}\widehat{x} = \mathbf{A}^\top b.$$

If the columns are independent,

$$\widehat{x} = (\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top b, \qquad \widehat{b} = \mathbf{P}b,$$

where

$$\mathbf{P} = \mathbf{A}(\mathbf{A}^\top\mathbf{A})^{-1}\mathbf{A}^\top.$$

Special case: orthonormal columns

If $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$, then

$$\widehat{b} = \mathbf{Q}\mathbf{Q}^\top b.$$

No matrix inverse is required, which is one reason orthonormal bases are computationally valuable.

Orthogonal matrices

A square matrix **Q** is orthogonal when

$$\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}, \qquad \mathbf{Q}^{-1} = \mathbf{Q}^\top.$$

Orthogonal transformations preserve inner products, norms, distances, and angles. Rotation and reflection matrices are important examples.

Object	Key property	Interpretation
Orthonormal columns	$\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$	Independent unit directions with no overlap
Orthogonal matrix	$\mathbf{Q}^{-1} = \mathbf{Q}^\top$	Length- and angle-preserving transformation
Projection matrix	$\mathbf{P}^2 = \mathbf{P}$	Repeating the projection has no further effect
Residual	$\mathbf{A}^\top r = 0$	Unexplained component is orthogonal to the model space

Bridge to Lecture 7
Least squares uses orthogonal projection when an exact solution to $\mathbf{A}x = b$ does not exist. The projected vector is the closest achievable output.

6. Application Case Study: Document Similarity

Rank documents by directional similarity in a term-feature space

An information-retrieval system represents text using four term features:

$$\text{linear}, \quad \text{algebra}, \quad \text{Python}, \quad \text{AI}.$$

The query and three candidate documents are represented by vectors:

$$\boldsymbol{q} = \begin{pmatrix} 2 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \quad \boldsymbol{d}_A = \begin{pmatrix} 3 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \quad \boldsymbol{d}_B = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \end{pmatrix}, \quad \boldsymbol{d}_C = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 2 \end{pmatrix}.$$

Why cosine similarity?

Raw dot products increase with document length. Cosine similarity normalizes both vectors and compares direction:

$$\text{cosim}(\boldsymbol{q}, \boldsymbol{d}) = \frac{\boldsymbol{q}^\top \boldsymbol{d}}{\|\boldsymbol{q}\|\|\boldsymbol{d}\|}.$$

Manual calculation

Since $\|\boldsymbol{q}\| = \sqrt{6}$,

$$\begin{aligned} \text{cosim}(\boldsymbol{q}, \boldsymbol{d}_A) &= \frac{8}{\sqrt{6}\sqrt{11}} \approx 0.9847, \\ \text{cosim}(\boldsymbol{q}, \boldsymbol{d}_B) &= \frac{3}{\sqrt{6}\sqrt{14}} \approx 0.3273, \\ \text{cosim}(\boldsymbol{q}, \boldsymbol{d}_C) &= \frac{4}{\sqrt{6}\sqrt{7}} \approx 0.6172. \end{aligned}$$

Document A is most aligned with the query and should rank first.

Projection interpretation

The part of $\boldsymbol{d}_A$ aligned with the query is

$$\text{proj}_{\boldsymbol{q}}(\boldsymbol{d}_A) = \frac{\boldsymbol{d}_A^\top \boldsymbol{q}}{\boldsymbol{q}^\top \boldsymbol{q}} \boldsymbol{q} = \frac{8}{6} \boldsymbol{q} = \begin{pmatrix} 8/3 \\ 4/3 \\ 4/3 \\ 0 \end{pmatrix}.$$

The residual is orthogonal to $\boldsymbol{q}$ and represents features of Document A not explained by the query direction.

Application decision
Rank the documents A, C, B. Document A has the smallest angular separation from the query and therefore the highest cosine similarity.

Responsible limitation
Term-count vectors ignore word order, context, synonyms, and meaning. Modern retrieval systems use richer embeddings, but inner products and cosine similarity remain central to comparing those representations.

7. Python Laboratory

Rank documents, compute a projection, verify orthogonality, and visualize scores

```python
import numpy as np
import matplotlib.pyplot as plt

terms = ["linear", "algebra", "Python", "AI"]
query = np.array([2.0, 1.0, 1.0, 0.0])
documents = np.array([
    [3.0, 1.0, 1.0, 0.0],  # Document A
    [0.0, 1.0, 2.0, 3.0],  # Document B
    [1.0, 1.0, 1.0, 2.0]   # Document C
])
names = np.array(["A", "B", "C"])

def cosine_similarity(vector, reference):
    denominator = np.linalg.norm(vector) * np.linalg.norm(reference)
    if denominator == 0:
        return 0.0
    return float(vector @ reference / denominator)

scores = np.array([
    cosine_similarity(document, query)
    for document in documents
])
ranking = np.argsort(scores)[::-1]

best_document = documents[ranking[0]]
projection = (best_document @ query) / (query @ query) * query
residual = best_document - projection

print("Cosine scores:", np.round(scores, 4))
print("Ranking:", names[ranking])
print("Best document:", names[ranking[0]])
print("Projection:", np.round(projection, 4))
print("Residual dot query:", round(float(residual @ query), 10))

fig, axes = plt.subplots(1, 2, figsize=(10, 3.8))
axes[0].bar(names, scores, color=["#1665D8", "#0C8F8F", "#E5A823"])
axes[0].set_ylim(0, 1.05)
axes[0].set_title("Cosine similarity to query")
axes[0].set_ylabel("Similarity")

x = np.arange(len(terms))
width = 0.36
axes[1].bar(x - width/2, query, width, label="Query")
axes[1].bar(x + width/2, best_document, width, label="Document A")
axes[1].set_xticks(x, terms, rotation=20)
axes[1].set_title("Term-feature comparison")
axes[1].legend()
plt.tight_layout()
plt.show()
```

Selected output

```
Cosine scores: [0.9847 0.3273 0.6172]
Ranking: ['A' 'C' 'B']
Best document: A
Projection: [2.6667 1.3333 1.3333 0.    ]
Residual dot query: 0.0
```

Interpretation
Cosine similarity ranks by direction rather than raw size. The zero inner product between the residual and query verifies the defining geometry of an orthogonal projection.

8. Concept Check and Practice

Common misconceptions

Misconception	Why it fails	Correct practice
Zero dot product means one vector is zero	Nonzero perpendicular vectors also have dot product zero	Interpret zero as orthogonality
Projection equals the original vector	Only when the vector already lies in the subspace	Calculate the residual
Cosine similarity is a distance	It measures normalized angular similarity	State the chosen similarity convention
Orthogonal always means unit length	Orthogonality concerns direction, not size	Normalize to obtain orthonormal vectors

Three concept-check MCQs

1. If $\boldsymbol{u}^\top \boldsymbol{v} = 0$, then $\boldsymbol{u}$ and $\boldsymbol{v}$ are:

A. parallel

B. orthogonal

C. equal

D. unit vectors

2. A projection matrix satisfies:

A. $\mathbf{P}^2 = \mathbf{P}$

B. $\mathbf{P}^2 = \mathbf{I}$

C. $\det(\mathbf{P}) = 1$ always

D. $\mathbf{P}^{-1} = \mathbf{P}$ always

3. If $\mathbf{Q}$ is orthogonal, then:

A. $\mathbf{Q}^{-1} = \mathbf{Q}^\top$

B. $\mathbf{Q}^{-1} = \mathbf{0}$

C. $\mathbf{Q}^2 = \mathbf{Q}$

D. $\operatorname{rank}(\mathbf{Q}) = 0$

MCQ answers: 1-B, 2-A, 3-A

Practice exercises

1. Find the inner product, norms, and cosine of the angle between $(1,2)^\top$ and $(2,-1)^\top$.
2. Determine whether $(1,1,0)^\top$ and $(1,-1,2)^\top$ are orthogonal.
3. Normalize the vector $(3,4)^\top$.
4. Project $\boldsymbol{b} = (4,2)^\top$ onto $\boldsymbol{a} = (1,1)^\top$ and find the residual.
5. Find the projection matrix onto $\operatorname{span}\{(1,0,1)^\top\}$.
6. Apply one Gram-Schmidt step to $(1,0,1)^\top$ and $(1,1,0)^\top$.
7. Add Document D $= (2,2,0,1)^\top$ to the application. Use Python to calculate its cosine similarity and update the ranking.

9. Scenario Challenge, Answers, and Takeaways

Scenario-based problem: signal separation

A measured signal is

$$\boldsymbol{y} = \begin{pmatrix} 3 \\ 4 \\ 5 \end{pmatrix},$$

and the known clean-signal direction is

$$\boldsymbol{s} = \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix}.$$

Project $\boldsymbol{y}$ onto span$\{\boldsymbol{s}\}$, interpret the projection as the explained signal, and verify that the residual noise is orthogonal to $\boldsymbol{s}$.

Brief answers to practice

Exercise	Brief answer
1	Inner product $= 0$; norms $= \sqrt{5}$ and $\sqrt{5}$; $\cos\theta = 0$
2	Yes: $1 - 1 + 0 = 0$
3	$(3/5, 4/5)^{\mathsf{T}}$
4	Projection $= (3,3)^{\mathsf{T}}$; residual $= (1,-1)^{\mathsf{T}}$
5	$\mathbf{P} = \frac{1}{2}\begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}$
6	$\boldsymbol{u}_1 = (1,0,1)^{\mathsf{T}}$ and $\boldsymbol{u}_2 = (1/2, 1, -1/2)^{\mathsf{T}}$
7	Similarity $= 6/\sqrt{54} \approx 0.8165$; ranking remains A, D, C, B
Scenario	Projection $= (7/3, 14/3, 14/3)^{\mathsf{T}}$; residual $= (2/3, -2/3, 1/3)^{\mathsf{T}}$ and $\boldsymbol{s}^{\mathsf{T}}\boldsymbol{r} = 0$

Lecture summary

- Inner products measure alignment and generate norms, distances, and angles.
- Orthogonal nonzero vectors are linearly independent.
- Orthonormal bases give simple, stable coordinate calculations.
- Gram-Schmidt converts independent directions into orthonormal directions.
- Projection gives the closest vector in a selected subspace.
- The residual is orthogonal to the subspace and represents unexplained information.

Research and career connection

Inner products and projections support information retrieval, signal processing, recommender systems, computer vision, communications, numerical analysis, optimization, quantum theory, data science, and machine learning.

Key terms: inner product • dot product • norm • angle • cosine similarity • orthogonal • orthonormal • orthogonal complement • Gram-Schmidt • projection • residual

Suggested references: Gilbert Strang, Introduction to Linear Algebra; David C. Lay, Steven R. Lay, and Judi J. McDonald, Linear Algebra and Its Applications; NumPy dot-product documentation; NumPy norm documentation.