#### **Unit Vectors and Vector Norms**

1. A unit vector is a vector that has a magnitude of \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** 1
2. The process of creating a unit vector from a nonzero vector v is called \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** normalizing
3. A unit vector u in the same direction as v is obtained by dividing v by its \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** length (or magnitude)

#### **Matrix Operations and Dimension Compatibility**

1. Matrix multiplication is defined if the number of \_\_\_\_\_\_\_\_\_\_\_ in the first matrix equals the number of \_\_\_\_\_\_\_\_\_\_\_ in the second matrix.  
   **Answer:** columns; rows
2. A square matrix is invertible if and only if its determinant is \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** nonzero
3. The transpose of the product of two matrices A and B is given by \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:**

#### **Least Squares**

1. The least-squares solution of the matrix equation Ax=b is given by x= \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:**
2. The least-squares problem minimizes the squared \_\_\_\_\_\_\_\_\_\_\_ between the observed and predicted values.  
   **Answer:** distance
3. The normal equations used to solve least-squares problems are given by \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:**

#### **Linear Combinations and Subspaces**

1. The span of a set of vectors is the \_\_\_\_\_\_\_\_\_\_\_ formed by all possible linear combinations of those vectors.  
   **Answer:** subspace
2. A vector b lies in the null space of A if and only if \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** Ab=0

#### **Eigenvalues and Eigenvectors**

1. The eigenvalues of a triangular matrix are the entries on its \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** main diagonal
2. The eigenvectors of a matrix A corresponding to an eigenvalue λ lie in the \_\_\_\_\_\_\_\_\_\_\_ of A−λI **Answer:** null space
3. The determinant of a matrix A equals the product of its \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** eigenvalues

#### **Principal Component Analysis (PCA)**

1. In PCA, the principal directions are given by the \_\_\_\_\_\_\_\_\_\_\_ of the covariance matrix.  
   **Answer:** eigenvectors
2. In PCA, the first principal component corresponds to the direction with the \_\_\_\_\_\_\_\_\_\_\_ variance in the data.  
   **Answer:** greatest
3. The covariance matrix in PCA is \_\_\_\_\_\_\_\_\_\_\_, which means its eigenvectors are orthogonal.  
   **Answer:** symmetric
4. The total variance in the data is preserved when performing PCA because the sum of the \_\_\_\_\_\_\_\_\_\_\_ equals the trace of the covariance matrix.  
   **Answer:** eigenvalues

#### **Singular Value Decomposition (SVD)**

1. The diagonal entries of the matrix Σ in SVD represent the \_\_\_\_\_\_\_\_\_\_\_ of the matrix.  
   **Answer:** singular values
2. The singular values in Σ are always arranged in \_\_\_\_\_\_\_\_\_\_\_ order.  
   **Answer:** descending
3. The eigenvalues of the covariance matrix are the squares of the \_\_\_\_\_\_\_\_\_\_\_ in the SVD of the data matrix.  
   **Answer:** singular values

#### **Null and Alternative Hypotheses**

1. The alternative hypothesis is a statement that we \_\_\_\_\_\_\_\_\_\_\_ if the null hypothesis is rejected.  
   **Answer:** accept
2. A two-tailed test is used when the alternative hypothesis states that the parameter is \_\_\_\_\_\_\_\_\_\_\_ the null hypothesis value.  
   **Answer:** not equal to

#### **Statistical Tests**

1. A Chi-Square test is used to assess the relationship between two \_\_\_\_\_\_\_\_\_\_\_ variables.  
   **Answer:** categorical
2. A paired t-test compares the means of two \_\_\_\_\_\_\_\_\_\_\_ samples.  
   **Answer:** dependent (or related)
3. The Chi-Square test of independence is used to determine if two \_\_\_\_\_\_\_\_\_\_\_ variables are related.  
   **Answer:** categorical
4. In hypothesis testing, the \_\_\_\_\_\_\_\_\_\_\_ level (α) represents the probability of making a Type I error.  
   **Answer:** significance

#### **Eigenvalues and Eigenvectors (Continued)**

1. An eigenvector of a matrix A satisfies the equation Ax=λx , where λ is the \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** eigenvalue
2. If λ is an eigenvalue of A, then A−λI is \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** non-invertible

#### **Singular Value Decomposition (SVD) (Continued)**

1. The matrix U in SVD represents the \_\_\_\_\_\_\_\_\_\_\_ singular vectors of the matrix.  
   **Answer:** left
2. The right singular vectors in SVD correspond to the \_\_\_\_\_\_\_\_\_\_\_ directions in PCA.  
   **Answer:** principal
3. The relationship between SVD and PCA is that the principal components in PCA are found using the \_\_\_\_\_\_\_\_\_\_\_ of V in SVD.  
   **Answer:** columns

#### **Principal Component Analysis (PCA) (Continued)**

1. The purpose of centering data in PCA is to ensure that the \_\_\_\_\_\_\_\_\_\_\_ of each attribute is 0.  
   **Answer:** mean

#### **Null and Alternative Hypotheses (Continued)**

1. The null hypothesis typically states that there is \_\_\_\_\_\_\_\_\_\_\_ effect or no difference.  
   **Answer:** no
2. A Type I error occurs when the null hypothesis is \_\_\_\_\_\_\_\_\_\_\_ and we reject it.  
   **Answer:** true
3. A Type II error occurs when the null hypothesis is \_\_\_\_\_\_\_\_\_\_\_ and we fail to reject it.  
   **Answer:** false

#### **Linear Combinations and Subspaces (Continued)**

1. A vector b lies in the column space of a matrix A if and only if Ax=b has at least one \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** solution
2. The column space of a matrix A consists of all linear combinations of the \_\_\_\_\_\_\_\_\_\_\_ of A.  
   **Answer:** columns

#### **Matrix Operations and Dimension Compatibility (Continued)**

1. The product of two matrices A (3×2) and B (2×3) results in a matrix of size \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** 3×3

#### **Statistical Tests (Continued)**

1. The Wilcoxon rank-sum test is used when data is \_\_\_\_\_\_\_\_\_\_\_ and not normally distributed.  
   **Answer:** unpaired
2. The Chi-Square test is used to test for independence between \_\_\_\_\_\_\_\_\_\_\_ variables.  
   **Answer:** categorical

#### **Applied Examples**

1. If a matrix A is of size 5×3, it will have \_\_\_\_\_\_\_\_\_\_\_ singular values in its decomposition.  
   **Answer:** 3
2. The determinant of a matrix A is the product of its \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:** eigenvalues

#### **PCA and Variance**

1. In PCA, the principal components are ordered by the amount of \_\_\_\_\_\_\_\_\_\_\_ they explain.  
   **Answer:** variance
2. The first principal component captures the \_\_\_\_\_\_\_\_\_\_\_ variance in the data.  
   **Answer:** greatest

#### **Eigenvalues and Eigenvectors (Continued)**

1. The eigenvalues of a covariance matrix represent the amount of \_\_\_\_\_\_\_\_\_\_\_ explained by each principal component.  
   **Answer:** variance

#### **SVD and Variance**

1. The total variance in a dataset is the sum of the squared \_\_\_\_\_\_\_\_\_\_\_ in the SVD of the data matrix.  
   **Answer:** singular values
2. SVD provides a decomposition of a matrix into three components: U, Σ, and \_\_\_\_\_\_\_\_\_\_\_.  
   **Answer:**

#### **Null and Alternative Hypotheses (Continued)**

1. A one-tailed test is used when the alternative hypothesis specifies a \_\_\_\_\_\_\_\_\_\_\_ direction of the effect.  
   **Answer:** specific
2. The p-value in hypothesis testing represents the probability of observing the test statistic under the assumption that the \_\_\_\_\_\_\_\_\_\_\_ hypothesis is true.  
   **Answer:** null

### **Vectors and Their Representations**

1. A vector in R^2 is represented as a column with how many entries?
   * **Answer**: 2
2. The set R^n represents all vectors in n-dimensional \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: space
3. The geometric interpretation of a vector in R^2 is a directed line segment with both \_\_\_\_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: magnitude; direction
4. The ordered pair [x, y] represents a vector in which space?
   * **Answer**: R^2

### **Scalar Multiplication**

1. Multiplying a vector by a scalar changes the vector’s \_\_\_\_\_\_\_\_\_\_\_ without changing its \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: magnitude; direction
2. If v = [2, -3], then 3v is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: [6, -9]
3. Scaling a vector by a negative scalar reverses its \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: direction

### **Vector Addition and Subtraction**

1. Adding two vectors involves adding their corresponding \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: components
2. Given u = [1, -2] and v = [3, 4], the sum u + v is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: [4, 2]
3. Subtracting one vector from another involves subtracting their corresponding \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: components

### **Vector Operations in Python**

1. In Python, which library is commonly used for vector operations such as addition and scalar multiplication?
   * **Answer**: NumPy
2. Write a Python code snippet to compute the sum of u = [1, -2] and v = [3, 4].

**Answer**:  
python  
Copy code  
import numpy as np

u = np.array([1, -2])

v = np.array([3, 4])

result = u + v

print(result)

### **Parallelogram Rule**

1. The geometric representation of vector addition is known as the \_\_\_\_\_\_\_\_\_\_\_ rule.
   * **Answer**: parallelogram
2. The diagonal of the parallelogram formed by two vectors represents the \_\_\_\_\_\_\_\_\_\_\_ of the two vectors.
   * **Answer**: sum

### **Vectors in Higher Dimensions**

1. A vector in R^n is represented as a column with \_\_\_\_\_\_\_\_\_\_\_ entries.
   * **Answer**: n
2. Vector operations such as addition and scalar multiplication generalize to higher dimensions because they involve \_\_\_\_\_\_\_\_\_\_\_ operations on corresponding entries.
   * **Answer**: component-wise

### **Miscellaneous**

1. Vectors with the same magnitude but opposite directions are called \_\_\_\_\_\_\_\_\_\_\_ vectors.
   * **Answer**: opposite
2. A vector with all components equal to zero is called the \_\_\_\_\_\_\_\_\_\_\_ vector.
   * **Answer**: zero
3. If two vectors u and v are equal, their corresponding components must be \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: identical

### **Vectors and Their Representations**

1. A vector in R^2 is represented as a column with how many entries?
   * **Answer**: 2
2. The set R^n represents all vectors in n-dimensional \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: space
3. The geometric interpretation of a vector in R^2 is a directed line segment with both \_\_\_\_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: magnitude; direction
4. The ordered pair [x, y] represents a vector in which space?
   * **Answer**: R^2

### **Scalar Multiplication**

1. Multiplying a vector by a scalar changes the vector’s \_\_\_\_\_\_\_\_\_\_\_ without changing its \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: magnitude; direction
2. If v = [2, -3], then 3v is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: [6, -9]
3. Scaling a vector by a negative scalar reverses its \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: direction

### **Vector Addition and Subtraction**

1. Adding two vectors involves adding their corresponding \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: components
2. Given u = [1, -2] and v = [3, 4], the sum u + v is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: [4, 2]
3. Subtracting one vector from another involves subtracting their corresponding \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: components

### **Vector Operations in Python**

1. In Python, which library is commonly used for vector operations such as addition and scalar multiplication?
   * **Answer**: NumPy
2. Write a Python code snippet to compute the sum of u = [1, -2] and v = [3, 4].

**Answer**:  
python  
Copy code  
import numpy as np

u = np.array([1, -2])

v = np.array([3, 4])

result = u + v

print(result)

### **Parallelogram Rule**

1. The geometric representation of vector addition is known as the \_\_\_\_\_\_\_\_\_\_\_ rule.
   * **Answer**: parallelogram
2. The diagonal of the parallelogram formed by two vectors represents the \_\_\_\_\_\_\_\_\_\_\_ of the two vectors.
   * **Answer**: sum

### **Vectors in Higher Dimensions**

1. A vector in R^n is represented as a column with \_\_\_\_\_\_\_\_\_\_\_ entries.
   * **Answer**: n
2. Vector operations such as addition and scalar multiplication generalize to higher dimensions because they involve \_\_\_\_\_\_\_\_\_\_\_ operations on corresponding entries.
   * **Answer**: component-wise

### **Miscellaneous**

1. Vectors with the same magnitude but opposite directions are called \_\_\_\_\_\_\_\_\_\_\_ vectors.
   * **Answer**: opposite
2. A vector with all components equal to zero is called the \_\_\_\_\_\_\_\_\_\_\_ vector.
   * **Answer**: zero
3. If two vectors u and v are equal, their corresponding components must be \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: identical

### **Related example questions from File 2: matrixMultiplication.pdf**

### **Matrix Basics**

1. A matrix is a rectangular array of \_\_\_\_\_\_\_\_\_\_\_ arranged in rows and columns.
   * **Answer**: numbers
2. A matrix with m rows and n columns is called a \_\_\_\_\_\_\_\_\_\_\_ matrix.
   * **Answer**: m x n
3. Matrix multiplication is defined only if the number of \_\_\_\_\_\_\_\_\_\_\_ in the first matrix equals the number of \_\_\_\_\_\_\_\_\_\_\_ in the second matrix.
   * **Answer**: columns; rows

### **Matrix Multiplication**

1. The product of an m x n matrix and an n x p matrix results in a matrix of size \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: m x p
2. To compute the (i, j)-th entry of the product of two matrices, take the \_\_\_\_\_\_\_\_\_\_\_ of the i-th row of the first matrix and the j-th column of the second matrix.
   * **Answer**: dot product
3. If A = [[1, 2], [3, 4]] and B = [[5, 6], [7, 8]], what is the product AB?
   * **Answer**: [[19, 22], [43, 50]]

### **Properties of Matrix Multiplication**

1. Matrix multiplication is \_\_\_\_\_\_\_\_\_\_\_ but not \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: associative; commutative
2. The transpose of the product of two matrices A and B is equal to the product of their transposes in \_\_\_\_\_\_\_\_\_\_\_ order.
   * **Answer**: reverse
3. If A is an identity matrix, the product A \* B equals \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: B
4. True or False: The diagonal entries of a diagonal matrix are scaled when multiplied by another matrix.
   * **Answer**: True

### **Python Operations**

1. Write a Python snippet to compute the product of A = [[1, 2], [3, 4]] and B = [[5, 6], [7, 8]].

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[1, 2], [3, 4]])

B = np.array([[5, 6], [7, 8]])

result = np.dot(A, B)

print(result)

1. Which Python library provides efficient tools for matrix operations?
   * **Answer**: NumPy

### **Applications of Matrix Multiplication**

1. In a system of linear equations A \* x = b, the product A \* x represents a \_\_\_\_\_\_\_\_\_\_\_ of the columns of A weighted by the components of x.
   * **Answer**: linear combination
2. Matrix multiplication is used to represent \_\_\_\_\_\_\_\_\_\_\_ transformations in geometry.
   * **Answer**: linear
3. True or False: The determinant of a product of two square matrices is the product of their determinants.
   * **Answer**: True

### **Special Cases**

1. If A is a diagonal matrix, the product A \* B scales each \_\_\_\_\_\_\_\_\_\_\_ of B by the corresponding diagonal entry of A.
   * **Answer**: row (or column, depending on context)
2. The identity matrix acts as the \_\_\_\_\_\_\_\_\_\_\_ element for matrix multiplication.
   * **Answer**: neutral
3. A matrix multiplied by a zero matrix results in a \_\_\_\_\_\_\_\_\_\_\_ matrix.
   * **Answer**: zero
4. If A is 3 x 2 and B is 2 x 4, the resulting matrix will have dimensions \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: 3 x 4

### **Related example questions from File 3: dotProd.pdf**

### **Dot Product Basics**

1. The dot product of two vectors u = [u1, u2, ..., un] and v = [v1, v2, ..., vn] is defined as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: u1*v1 + u2*v2 + ... + un\*vn
2. The result of a dot product between two vectors is a \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: scalar
3. The geometric interpretation of the dot product involves the \_\_\_\_\_\_\_\_\_\_\_ of one vector onto another.
   * **Answer**: projection

### **Properties of the Dot Product**

1. The dot product of two orthogonal vectors is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: 0
2. True or False: The dot product is commutative, meaning u . v = v . u.
   * **Answer**: True
3. The dot product of a vector v with itself is equal to the \_\_\_\_\_\_\_\_\_\_\_ of v squared.
   * **Answer**: magnitude

### **Example Calculations**

1. Compute the dot product of u = [1, -3, 4] and v = [2, 1, -2].
   * **Answer**: -9
2. If u = [0, 1] and v = [1, 0], what is u . v?
   * **Answer**: 0

### **Applications of the Dot Product**

1. The dot product is used to determine the \_\_\_\_\_\_\_\_\_\_\_ between two vectors.
   * **Answer**: angle
2. Write the formula to compute the angle theta between two vectors using the dot product.
   * **Answer**: cos(theta) = (u . v) / (|u| \* |v|)

### **Python Implementation**

1. Write a Python code snippet to compute the dot product of u = [1, -3, 4] and v = [2, 1, -2].

**Answer**:  
python  
Copy code  
import numpy as np

u = np.array([1, -3, 4])

v = np.array([2, 1, -2])

dot\_product = np.dot(u, v)

print(dot\_product)

1. Which Python function is used to compute the dot product of two vectors?
   * **Answer**: np.dot

### **Geometric Interpretation**

1. The projection of v onto u is computed as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: (u . v) / |u|^2 \* u
2. The dot product reveals whether two vectors are \_\_\_\_\_\_\_\_\_\_\_, \_\_\_\_\_\_\_\_\_\_\_, or \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: parallel, orthogonal, anti-parallel

### **Special Cases**

1. If u and v are parallel, their dot product is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: maximum (or |u| \* |v|)
2. If u and v are anti-parallel, their dot product is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: negative

### **Related example questions from File 4: dotProd2.pdf**

### **Dot Product Properties (Extended)**

1. The dot product of two vectors u and v is zero if and only if the vectors are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: orthogonal
2. The projection of vector v onto vector u is given by the formula \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: [(u . v) / (u . u)] \* u
3. True or False: The magnitude of the projection of v onto u is equal to |v| \* cos(theta), where theta is the angle between u and v.
   * **Answer**: True

### **Applications of the Dot Product**

1. The dot product is used in graphics to compute \_\_\_\_\_\_\_\_\_\_\_, which determine the brightness of a surface.
   * **Answer**: lighting angles
2. In physics, the dot product of force and displacement vectors is used to calculate \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: work
3. If u = [3, 4] and v = [1, -2], the projection of v onto u is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: [1.32, 1.76]

### **Python Implementation**

1. Write a Python code snippet to compute the projection of vector v = [1, -2] onto vector u = [3, 4].

**Answer**:  
python  
Copy code  
import numpy as np

u = np.array([3, 4])

v = np.array([1, -2])

projection = (np.dot(u, v) / np.dot(u, u)) \* u

print(projection)

### **Special Cases**

1. If the angle between two vectors is 90 degrees, their dot product is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: zero
2. True or False: The dot product can be negative if the angle between the two vectors is greater than 90 degrees.
   * **Answer**: True

### **Geometric Applications**

1. The cosine of the angle between two vectors u and v can be calculated as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: (u . v) / (|u| \* |v|)
2. If u = [2, 2] and v = [4, -4], the cosine of the angle between them is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: -0.707

### **Related example questions from File 5: l2NormUnitVector.pdf**

### **Vector Norms**

1. The L2 norm (Euclidean norm) of a vector u = [u1, u2, ..., un] is calculated as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: sqrt(u1^2 + u2^2 + ... + un^2)
2. A unit vector has a magnitude of \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: 1
3. The process of dividing a vector by its magnitude to create a unit vector is called \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: normalization

### **Applications of Unit Vectors**

1. Unit vectors are often used in physics to represent \_\_\_\_\_\_\_\_\_\_\_ without specifying magnitude.
   * **Answer**: direction
2. True or False: Any nonzero vector can be converted into a unit vector by dividing each component by the vector's magnitude.
   * **Answer**: True
3. If v = [3, 4], the unit vector in the same direction as v is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: [0.6, 0.8]

### **Python Implementation**

1. Write a Python code snippet to calculate the L2 norm of vector v = [3, 4].

**Answer**:  
python  
Copy code  
import numpy as np

v = np.array([3, 4])

norm = np.linalg.norm(v)

print(norm)

1. Write a Python code snippet to normalize vector v = [3, 4].

**Answer**:  
python  
Copy code  
v = np.array([3, 4])

unit\_vector = v / np.linalg.norm(v)

print(unit\_vector)

### **Miscellaneous**

1. If the magnitude of a vector is 0, it is called the \_\_\_\_\_\_\_\_\_\_\_ vector.
   * **Answer**: zero
2. The dot product of a unit vector with itself is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: 1

### **Related example questions from File 6: matrixEquation.pdf**

### **Matrix Equations**

1. A matrix equation is typically written in the form \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: A \* x = b
2. The matrix equation A \* x = b has a solution if and only if b lies in the \_\_\_\_\_\_\_\_\_\_\_ of A.
   * **Answer**: column space
3. The inverse of a square matrix A exists if and only if its determinant is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: nonzero
4. If A = [[2, 1], [1, 3]] and b = [5, 7], solve for x in the equation A \* x = b.
   * **Answer**: x = [1, 2]

### **Properties**

1. The product A \* x represents a \_\_\_\_\_\_\_\_\_\_\_ of the columns of A, weighted by the components of x.
   * **Answer**: linear combination
2. A system of linear equations has a unique solution if the coefficient matrix is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: invertible

### **Python Implementation**

1. Write a Python code snippet to solve the matrix equation A \* x = b for A = [[2, 1], [1, 3]] and b = [5, 7].

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[2, 1], [1, 3]])

b = np.array([5, 7])

x = np.linalg.solve(A, b)

print(x)

### **Miscellaneous**

1. The set of all solutions to the equation A \* x = 0 forms the \_\_\_\_\_\_\_\_\_\_\_ of A.
   * **Answer**: null space
2. If A is 3 x 2, then the equation A \* x = b can have at most \_\_\_\_\_\_\_\_\_\_\_ linearly independent solutions.
   * **Answer**: 2

### **Related example questions from File 7: linearCombinationSpan.pdf**

### **Linear Combinations**

1. A linear combination of vectors v1, v2, ..., vn involves forming expressions of the type \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: c1 \* v1 + c2 \* v2 + ... + cn \* vn
2. The coefficients in a linear combination are typically \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: scalars
3. If v1 = [1, 0], v2 = [0, 1], and v3 = [1, 1], is [2, 2] a linear combination of these vectors?
   * **Answer**: Yes

### **Span**

1. The span of a set of vectors is the set of all \_\_\_\_\_\_\_\_\_\_\_ of those vectors.
   * **Answer**: linear combinations
2. If a set of vectors spans R^n, then any vector in R^n can be written as a \_\_\_\_\_\_\_\_\_\_\_ of those vectors.
   * **Answer**: linear combination
3. If v1 = [1, 2] and v2 = [3, 4], the span of v1 and v2 is a \_\_\_\_\_\_\_\_\_\_\_ in R^2.
   * **Answer**: plane

### **Linear Independence**

1. A set of vectors is linearly independent if no vector in the set can be written as a \_\_\_\_\_\_\_\_\_\_\_ of the others.
   * **Answer**: linear combination
2. If a set of vectors is linearly dependent, at least one vector in the set can be expressed as a \_\_\_\_\_\_\_\_\_\_\_ of the others.
   * **Answer**: linear combination

### **Applications**

1. The column space of a matrix A is spanned by the \_\_\_\_\_\_\_\_\_\_\_ of A.
   * **Answer**: columns
2. True or False: If a set of vectors is linearly dependent, their span is the entire vector space.
   * **Answer**: False

### **Python Implementation**

1. Write a Python code snippet to check if a vector v = [2, 4] lies in the span of vectors v1 = [1, 2] and v2 = [3, 4].

**Answer**:  
python  
Copy code  
import numpy as np

v = np.array([2, 4])

v1 = np.array([1, 2])

v2 = np.array([3, 4])

A = np.column\_stack([v1, v2])

solution = np.linalg.lstsq(A, v, rcond=None)[0]

print("Solution:", solution)

### **Miscellaneous**

1. The null space of a matrix A is the set of all vectors that are mapped to \_\_\_\_\_\_\_\_\_\_\_ by A.
   * **Answer**: zero
2. If A is a 3 x 3 matrix, its column space can have at most \_\_\_\_\_\_\_\_\_\_\_ dimensions.
   * **Answer**: 3

### **Related example questions from File 8: leastSquare.pdf**

### **Least Squares**

1. The least-squares method is used to find an approximate solution to an overdetermined system of equations of the form \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: A \* x = b
2. The least-squares solution minimizes the \_\_\_\_\_\_\_\_\_\_\_ of the residual vector.
   * **Answer**: norm
3. The residual vector is given by \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: r = b - A \* x

### **Normal Equation**

1. The least-squares solution satisfies the normal equation \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: A^T \* A \* x = A^T \* b
2. The solution to the normal equation is x = \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: (A^T \* A)^(-1) \* A^T \* b

### **Geometric Interpretation**

1. The least-squares solution projects the vector b onto the \_\_\_\_\_\_\_\_\_\_\_ of A.
   * **Answer**: column space
2. The residual vector is orthogonal to the \_\_\_\_\_\_\_\_\_\_\_ of A.
   * **Answer**: column space

### **Applications**

1. Least squares is used in linear regression to fit a \_\_\_\_\_\_\_\_\_\_\_ to a set of data points.
   * **Answer**: line
2. True or False: The total squared residual error increases as the least-squares solution becomes more accurate.
   * **Answer**: False

### **Python Implementation**

1. Write a Python code snippet to solve the least-squares problem for A = [[1, 1], [1, 2], [1, 3]] and b = [1, 2, 2].

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[1, 1], [1, 2], [1, 3]])

b = np.array([1, 2, 2])

x = np.linalg.lstsq(A, b, rcond=None)[0]

print(x)

### **Miscellaneous**

1. If A is a 3 x 2 matrix, the least-squares solution minimizes the error vector in \_\_\_\_\_\_\_\_\_\_\_ dimensions.
   * **Answer**: 3
2. The least-squares solution is unique if the columns of A are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: linearly independent

### **Related example questions from File 9: eigenvectorGeo.pdf**

### **Eigenvalues and Eigenvectors**

1. An eigenvector of a matrix A satisfies the equation \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: A \* v = lambda \* v
2. The scalar lambda in the eigenvector equation is called the \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: eigenvalue
3. The geometric interpretation of an eigenvector is a vector whose direction remains \_\_\_\_\_\_\_\_\_\_\_ after the matrix transformation.
   * **Answer**: unchanged

### **Properties**

1. Eigenvectors corresponding to different eigenvalues are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: linearly independent
2. True or False: The eigenvectors of a symmetric matrix are orthogonal.
   * **Answer**: True

### **Example**

1. If A = [[2, 1], [1, 2]], find the eigenvalues of A.
   * **Answer**: 3, 1
2. If lambda = 3 is an eigenvalue of A, solve for the eigenvector of A = [[2, 1], [1, 2]].
   * **Answer**: [1, 1]

### **Applications**

1. Eigenvectors are used in PCA to identify \_\_\_\_\_\_\_\_\_\_\_ in the data.
   * **Answer**: principal directions
2. True or False: The eigenvalues of a matrix determine the scaling factor in the direction of the eigenvector.
   * **Answer**: True

### **Miscellaneous**

1. A matrix is diagonalizable if it has enough \_\_\_\_\_\_\_\_\_\_\_ eigenvectors to form a basis.
   * **Answer**: linearly independent
2. The eigenvalues of a triangular matrix are the entries on its \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: main diagonal

### **Related example questions from File 10: parametricEigenvec.pdf**

### **Parametric Representation of Eigenvectors**

1. To find the eigenvector corresponding to an eigenvalue, solve the equation \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: (A - lambda \* I) \* v = 0
2. The eigenvector equation often results in a system of \_\_\_\_\_\_\_\_\_\_\_ equations.
   * **Answer**: linear
3. True or False: Eigenvectors can be expressed in parametric form with free variables.
   * **Answer**: True
4. If A = [[3, -2], [1, 0]] and lambda = 2, write the eigenvector of A in parametric form.
   * **Answer**: v = t \* [2, 1], where t is a free parameter

### **Eigenspaces**

1. The set of all eigenvectors corresponding to a single eigenvalue forms a \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: eigenspace
2. The eigenspace of a matrix is a \_\_\_\_\_\_\_\_\_\_\_ of the original space.
   * **Answer**: subspace
3. True or False: The eigenspace of a symmetric matrix is orthogonal.
   * **Answer**: True

### **Applications**

1. Eigenvectors in parametric form are used to represent \_\_\_\_\_\_\_\_\_\_\_ solutions.
   * **Answer**: infinite
2. If lambda = 1 and the eigenvector is v = t \* [1, -1], the eigenspace is a line passing through \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: the origin

### **Python Implementation**

1. Write a Python code snippet to compute the eigenvectors of A = [[3, -2], [1, 0]].

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[3, -2], [1, 0]])

eigenvalues, eigenvectors = np.linalg.eig(A)

print("Eigenvectors:", eigenvectors)

### **Miscellaneous**

1. An eigenvector is unique up to \_\_\_\_\_\_\_\_\_\_\_ scaling.
   * **Answer**: scalar
2. The parametric form of eigenvectors is used to represent \_\_\_\_\_\_\_\_\_\_\_ solutions to eigenvalue problems.
   * **Answer**: infinite

### **Related example questions from File 11: characteristicEq.pdf**

### **Characteristic Equation**

1. The characteristic equation of a matrix A is obtained from the determinant equation \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: det(A - lambda \* I) = 0
2. The roots of the characteristic equation are the \_\_\_\_\_\_\_\_\_\_\_ of the matrix.
   * **Answer**: eigenvalues
3. If A = [[3, -2], [1, 0]], write the characteristic equation.
   * **Answer**: lambda^2 - 3 \* lambda + 2 = 0

### **Steps to Compute Eigenvalues**

1. To find eigenvalues, subtract lambda from the \_\_\_\_\_\_\_\_\_\_\_ entries of the matrix.
   * **Answer**: diagonal
2. Solve the determinant equation det(A - lambda \* I) = 0 to find the \_\_\_\_\_\_\_\_\_\_\_ of the matrix.
   * **Answer**: eigenvalues
3. True or False: The characteristic equation of a triangular matrix is formed directly from its diagonal entries.
   * **Answer**: True

### **Applications**

1. The characteristic equation is used to determine the \_\_\_\_\_\_\_\_\_\_\_ of a matrix.
   * **Answer**: eigenvalues
2. True or False: The eigenvalues of a matrix are always real numbers.
   * **Answer**: False

### **Python Implementation**

1. Write a Python code snippet to compute the characteristic equation of A = [[3, -2], [1, 0]].

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[3, -2], [1, 0]])

eigenvalues = np.linalg.eigvals(A)

print("Eigenvalues:", eigenvalues)

### **Miscellaneous**

1. The characteristic polynomial of an n x n matrix is a polynomial of degree \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: n
2. A matrix is invertible if and only if none of its eigenvalues are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: zero

### **Related example questions from File 12: whyDiag.pdf**

### **Diagonalization Basics**

1. A matrix AAA is diagonalizable if it can be expressed as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: A = P \* D \* P^(-1)
2. In the diagonalization equation A=P∗D∗P(−1)A = P \* D \* P^(-1)A=P∗D∗P(−1), the columns of PPP are \_\_\_\_\_\_\_\_\_\_\_ of AAA.
   * **Answer**: eigenvectors
3. The diagonal entries of DDD in A=P∗D∗P(−1)A = P \* D \* P^(-1)A=P∗D∗P(−1) are the \_\_\_\_\_\_\_\_\_\_\_ of AAA.
   * **Answer**: eigenvalues

### **Properties of Diagonalizable Matrices**

1. A matrix is diagonalizable if it has enough \_\_\_\_\_\_\_\_\_\_\_ eigenvectors to form a basis.
   * **Answer**: linearly independent
2. True or False: Every square matrix is diagonalizable.
   * **Answer**: False
3. The geometric multiplicity of an eigenvalue must equal its \_\_\_\_\_\_\_\_\_\_\_ multiplicity for a matrix to be diagonalizable.
   * **Answer**: algebraic

### **Applications of Diagonalization**

1. Diagonalization simplifies the computation of \_\_\_\_\_\_\_\_\_\_\_ of a matrix.
   * **Answer**: powers
2. In dynamical systems, diagonalization is used to analyze \_\_\_\_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: stability; behavior
3. True or False: A symmetric matrix is always diagonalizable.
   * **Answer**: True

### **Example**

1. If A=[[2,1],[1,2]]A = [[2, 1], [1, 2]]A=[[2,1],[1,2]], find its eigenvalues and verify if it is diagonalizable.
   * **Answer**: Eigenvalues = 3, 1; Yes, it is diagonalizable.

### **Python Implementation**

1. Write a Python code snippet to check if a matrix A=[[2,1],[1,2]]A = [[2, 1], [1, 2]]A=[[2,1],[1,2]] is diagonalizable.

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[2, 1], [1, 2]])

eigenvalues, eigenvectors = np.linalg.eig(A)

print("Eigenvalues:", eigenvalues)

print("Eigenvectors:", eigenvectors)

### **Miscellaneous**

1. A matrix that is not diagonalizable is called \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: defective
2. The eigenvalues of a diagonal matrix are the \_\_\_\_\_\_\_\_\_\_\_ of the matrix.
   * **Answer**: diagonal entries

### **Related example questions from File 13: howToDiag.pdf**

### **Steps to Diagonalize a Matrix**

1. The first step to diagonalize a matrix is to compute its \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: eigenvalues
2. After finding the eigenvalues, solve for the \_\_\_\_\_\_\_\_\_\_\_ of each eigenvalue.
   * **Answer**: eigenvectors
3. Form a matrix PPP using the eigenvectors as its \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: columns
4. The diagonal matrix DDD contains the \_\_\_\_\_\_\_\_\_\_\_ of the matrix.
   * **Answer**: eigenvalues

### **Conditions for Diagonalization**

1. True or False: A matrix is diagonalizable only if it has distinct eigenvalues.
   * **Answer**: False
2. The eigenvectors of a symmetric matrix are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: orthogonal
3. If AAA is diagonalizable, then AkA^kAk can be computed as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: P∗Dk∗P(−1)P \* D^k \* P^(-1)P∗Dk∗P(−1)

### **Example**

1. Diagonalize A=[[1,3],[3,1]]A = [[1, 3], [3, 1]]A=[[1,3],[3,1]].
   * **Answer**: P=[[1,1],[1,−1]],D=[[4,0],[0,−2]]P = [[1, 1], [1, -1]], D = [[4, 0], [0, -2]]P=[[1,1],[1,−1]],D=[[4,0],[0,−2]]

### **Applications**

1. Diagonalization is used in Principal Component Analysis (PCA) to \_\_\_\_\_\_\_\_\_\_\_ covariance matrices.
   * **Answer**: simplify
2. Diagonalization helps in solving \_\_\_\_\_\_\_\_\_\_\_ differential equations.
   * **Answer**: systems of

### **Python Implementation**

1. Write a Python code snippet to diagonalize a matrix A=[[1,3],[3,1]]A = [[1, 3], [3, 1]]A=[[1,3],[3,1]].

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[1, 3], [3, 1]])

eigenvalues, eigenvectors = np.linalg.eig(A)

D = np.diag(eigenvalues)

P = eigenvectors

P\_inv = np.linalg.inv(P)

print("P:", P)

print("D:", D)

### **Miscellaneous**

1. If a matrix AAA is diagonalizable, the eigenvectors must form a \_\_\_\_\_\_\_\_\_\_\_ basis.
   * **Answer**: complete
2. The sum of the eigenvalues of a matrix equals the \_\_\_\_\_\_\_\_\_\_\_ of the matrix.
   * **Answer**: trace

### **Related example questions from File 14: whyPCA.pdf**

### **Principal Component Analysis (PCA)**

1. The main purpose of PCA is to reduce \_\_\_\_\_\_\_\_\_\_\_ while preserving as much \_\_\_\_\_\_\_\_\_\_\_ as possible.
   * **Answer**: dimensionality; variance
2. PCA identifies the directions in the data with the greatest \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: variance
3. The directions identified by PCA are called \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: principal components

### **Data Simplification**

1. PCA is used to simplify datasets by identifying and removing \_\_\_\_\_\_\_\_\_\_\_ features.
   * **Answer**: redundant
2. True or False: PCA reduces the number of attributes while retaining the structure of the data.
   * **Answer**: True

### **Applications**

1. PCA is commonly used in \_\_\_\_\_\_\_\_\_\_\_ to reduce the number of pixels while retaining visual information.
   * **Answer**: image compression
2. PCA is used in text processing to identify key \_\_\_\_\_\_\_\_\_\_\_ in documents.
   * **Answer**: topics

### **Miscellaneous**

1. The principal components are orthogonal because the covariance matrix is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: symmetric
2. PCA is particularly useful when working with \_\_\_\_\_\_\_\_\_\_\_ datasets.
   * **Answer**: high-dimensional

### **Related example questions from File 15: whyGreatestVariability.pdf**

### **Why PCA Focuses on Greatest Variability**

1. PCA identifies directions in the data with the \_\_\_\_\_\_\_\_\_\_\_ variance.
   * **Answer**: greatest
2. High-variance dimensions in the data capture the most \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: information
3. True or False: PCA assumes that data with low variance is less significant.
   * **Answer**: True

### **Dimensionality Reduction**

1. The principal components are ordered by the amount of \_\_\_\_\_\_\_\_\_\_\_ they explain.
   * **Answer**: variance
2. When performing PCA, the first principal component corresponds to the \_\_\_\_\_\_\_\_\_\_\_ variance in the data.
   * **Answer**: largest
3. Removing low-variance dimensions minimizes the loss of \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: information

### **Geometric Interpretation**

1. PCA projects data onto a new coordinate system where the axes correspond to the \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: principal components
2. True or False: The principal components are orthogonal to one another.
   * **Answer**: True

### **Applications**

1. In a dataset with temperature in Celsius and Fahrenheit, PCA would identify the \_\_\_\_\_\_\_\_\_\_\_ dimension as redundant.
   * **Answer**: second
2. PCA is useful in datasets like MNIST to reduce the number of \_\_\_\_\_\_\_\_\_\_\_ while retaining structural information.
   * **Answer**: pixels

### **Miscellaneous**

1. The variance of each principal component is equal to the corresponding \_\_\_\_\_\_\_\_\_\_\_ of the covariance matrix.
   * **Answer**: eigenvalue
2. True or False: PCA can be used to visualize high-dimensional data in 2D or 3D.
   * **Answer**: True

### **Related example questions from File 16: howToPCA.pdf**

### **Steps for PCA**

1. The first step in PCA is to center the data by subtracting the \_\_\_\_\_\_\_\_\_\_\_ of each attribute.
   * **Answer**: mean
2. The covariance matrix is computed as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: (1 / (n - 1)) \* X^T \* X
3. The principal components are the \_\_\_\_\_\_\_\_\_\_\_ of the covariance matrix.
   * **Answer**: eigenvectors
4. The new data coordinates after PCA are computed as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: Y = X \* P

### **Example**

1. Given a dataset with X = [[2, 0], [0, 2], [3, 1], [1, 3]], compute the covariance matrix.
   * **Answer**: [[1.6667, 0.6667], [0.6667, 1.6667]]
2. The eigenvalues of the covariance matrix represent the \_\_\_\_\_\_\_\_\_\_\_ of the data along each principal component.
   * **Answer**: variance

### **Geometric Interpretation**

1. The first principal component aligns with the direction of \_\_\_\_\_\_\_\_\_\_\_ variance.
   * **Answer**: maximum
2. The transformed data in PCA is a projection onto the \_\_\_\_\_\_\_\_\_\_\_ directions.
   * **Answer**: principal component

### **Python Implementation**

1. Write a Python code snippet to perform PCA on X = [[2, 0], [0, 2], [3, 1], [1, 3]].

**Answer**:  
python  
Copy code  
import numpy as np

X = np.array([[2, 0], [0, 2], [3, 1], [1, 3]])

X\_centered = X - np.mean(X, axis=0)

covariance\_matrix = np.cov(X\_centered.T)

eigenvalues, eigenvectors = np.linalg.eig(covariance\_matrix)

print("Eigenvectors:", eigenvectors)

print("Eigenvalues:", eigenvalues)

### **Miscellaneous**

1. The sum of the eigenvalues of the covariance matrix equals the \_\_\_\_\_\_\_\_\_\_\_ variance in the data.
   * **Answer**: total
2. True or False: PCA always reduces the dimensionality of a dataset.
   * **Answer**: False

### **Related example questions from File 17: howToScorePCA.pdf**

### **PCA Scores**

1. The PCA scores represent the \_\_\_\_\_\_\_\_\_\_\_ of the data in the new principal component space.
   * **Answer**: coordinates
2. The scores in PCA are calculated as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: Y = P^T \* X
3. The variance explained by each principal component is proportional to its corresponding \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: eigenvalue

### **Example**

1. If X = [[1, 2], [2, 4], [3, 6]] and the principal components are P = [[0.8, 0.6], [-0.6, 0.8]], compute the scores.
   * **Answer**: [[2.68, 0.0], [5.36, 0.0], [8.04, 0.0]]
2. True or False: The PCA scores are uncorrelated in the new space.
   * **Answer**: True

### **Applications**

1. PCA scores are used to reduce high-dimensional data to \_\_\_\_\_\_\_\_\_\_\_ dimensions for visualization.
   * **Answer**: 2 or 3
2. In feature extraction, PCA scores are treated as \_\_\_\_\_\_\_\_\_\_\_ features.
   * **Answer**: new

### **Python Implementation**

1. Write a Python code snippet to compute PCA scores for X = [[1, 2], [2, 4], [3, 6]].

**Answer**:  
python  
Copy code  
import numpy as np

X = np.array([[1, 2], [2, 4], [3, 6]])

mean = np.mean(X, axis=0)

X\_centered = X - mean

covariance\_matrix = np.cov(X\_centered.T)

eigenvalues, eigenvectors = np.linalg.eig(covariance\_matrix)

scores = np.dot(X\_centered, eigenvectors)

print("PCA Scores:", scores)

### **Miscellaneous**

1. PCA scores are orthogonal because the principal components are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: orthogonal
2. The total variance in the original data equals the total variance in the \_\_\_\_\_\_\_\_\_\_\_ data.
   * **Answer**: transformed

### **Related example questions from File 18: LandsetPCA (2).pdf**

### **PCA on Multichannel Data**

1. Landsat data often includes multiple channels, each representing a specific \_\_\_\_\_\_\_\_\_\_\_ band.
   * **Answer**: wavelength
2. PCA is used to reduce redundancy in multichannel data by identifying \_\_\_\_\_\_\_\_\_\_\_ components.
   * **Answer**: principal
3. The covariance matrix of the transformed data after PCA is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: diagonal
4. The variance along each principal component is equal to the corresponding \_\_\_\_\_\_\_\_\_\_\_ of the covariance matrix.
   * **Answer**: eigenvalue

### **Dimensionality Reduction**

1. The principal components with the highest eigenvalues capture the majority of the \_\_\_\_\_\_\_\_\_\_\_ in the data.
   * **Answer**: variance
2. In Landsat data, PCA reduces the number of channels while preserving most of the \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: information

### **Example**

1. If a Landsat dataset has 7 spectral bands, how many principal components will the PCA compute?
   * **Answer**: 7
2. True or False: The first principal component in Landsat PCA often captures more than 90% of the variance.
   * **Answer**: True

### **Applications**

1. PCA is used in remote sensing to detect \_\_\_\_\_\_\_\_\_\_\_ in multichannel images.
   * **Answer**: patterns
2. Landsat PCA is applied to monitor \_\_\_\_\_\_\_\_\_\_\_, such as urban growth and deforestation.
   * **Answer**: environmental changes

### **Python Implementation**

1. Write a Python snippet to perform PCA on a multichannel Landsat dataset with 7 bands.

**Answer**:  
python  
Copy code  
import numpy as np

X = np.random.rand(7, 1000) # Simulated data: 7 bands, 1000 pixels

X\_centered = X - np.mean(X, axis=1, keepdims=True)

covariance\_matrix = np.cov(X\_centered)

eigenvalues, eigenvectors = np.linalg.eig(covariance\_matrix)

principal\_components = np.dot(eigenvectors.T, X\_centered)

print("Principal Components Shape:", principal\_components.shape)

### **Miscellaneous**

1. The total variance in the original data is equal to the sum of the \_\_\_\_\_\_\_\_\_\_\_ of the covariance matrix.
   * **Answer**: eigenvalues
2. PCA projects high-dimensional data onto a lower-dimensional \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: subspace

### **Related example questions from File 19: svdRecommenderSystems.pdf**

### **SVD Basics**

1. Singular Value Decomposition (SVD) decomposes a matrix AAA into three matrices: \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: U, Σ, and V^T
2. In SVD, the matrix Σ contains the \_\_\_\_\_\_\_\_\_\_\_ of the matrix AAA.
   * **Answer**: singular values
3. The columns of U in SVD are the \_\_\_\_\_\_\_\_\_\_\_ singular vectors of AAA.
   * **Answer**: left
4. The columns of V in SVD are the \_\_\_\_\_\_\_\_\_\_\_ singular vectors of AAA.
   * **Answer**: right

### **Applications in Recommender Systems**

1. SVD is used in recommender systems to identify \_\_\_\_\_\_\_\_\_\_\_ features.
   * **Answer**: latent
2. True or False: SVD can approximate a user-item rating matrix for personalized recommendations.
   * **Answer**: True

### **Low-Rank Approximation**

1. A low-rank approximation of a matrix is obtained by retaining only the top kkk \_\_\_\_\_\_\_\_\_\_\_ and corresponding singular vectors.
   * **Answer**: singular values
2. The approximation error for a low-rank representation is proportional to the \_\_\_\_\_\_\_\_\_\_\_ singular values excluded.
   * **Answer**: smallest

### **Example**

1. Given a user-item matrix A=[[0,1,0],[1,0,1],[0,1,1]]A = [[0, 1, 0], [1, 0, 1], [0, 1, 1]]A=[[0,1,0],[1,0,1],[0,1,1]], use SVD to find a rank-2 approximation.
   * **Answer**: Approximate AAA using the top 2 singular values and corresponding singular vectors.

### **Python Implementation**

1. Write a Python snippet to compute the low-rank approximation of a user-item matrix A=[[0,1,0],[1,0,1],[0,1,1]]A = [[0, 1, 0], [1, 0, 1], [0, 1, 1]]A=[[0,1,0],[1,0,1],[0,1,1]].

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[0, 1, 0], [1, 0, 1], [0, 1, 1]])

U, Sigma, VT = np.linalg.svd(A, full\_matrices=False)

k = 2

U\_k = U[:, :k]

Sigma\_k = np.diag(Sigma[:k])

VT\_k = VT[:k, :]

A\_k = np.dot(U\_k, np.dot(Sigma\_k, VT\_k))

print("Low-rank Approximation:\n", A\_k)

### **Miscellaneous**

1. True or False: The singular values of a matrix are always non-negative.
   * **Answer**: True
2. The rank of a matrix equals the number of non-zero \_\_\_\_\_\_\_\_\_\_\_ in Σ.
   * **Answer**: singular values

### **Related example questions from File 20: PCA\_SVD.pdf**

### **Relationship Between PCA and SVD**

1. PCA and SVD are related because the principal components in PCA are the \_\_\_\_\_\_\_\_\_\_\_ of the matrix ATAA^T AATA.
   * **Answer**: eigenvectors
2. In PCA, the eigenvalues of the covariance matrix correspond to the squares of the \_\_\_\_\_\_\_\_\_\_\_ in SVD.
   * **Answer**: singular values
3. The SVD of a centered data matrix AAA gives the principal components as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: rows of UΣU \SigmaUΣ

### **Applications**

1. SVD is used to compute PCA efficiently, especially for \_\_\_\_\_\_\_\_\_\_\_ datasets.
   * **Answer**: large
2. True or False: PCA can be performed by using only the right singular vectors in SVD.
   * **Answer**: False

### **Python Implementation**

1. Write a Python snippet to compute PCA using SVD for a data matrix AAA.

**Answer**:  
python  
Copy code  
import numpy as np

A = np.random.rand(5, 3) # Example data

U, Sigma, VT = np.linalg.svd(A, full\_matrices=False)

principal\_components = np.dot(U, np.diag(Sigma))

print("Principal Components:\n", principal\_components)

### **Miscellaneous**

1. The covariance matrix of a centered dataset is diagonalized in PCA because the principal components are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: orthogonal
2. True or False: PCA preserves the total variance in the data.
   * **Answer**: True

### **Related example questions from File 21: svdDimReduction.pdf**

### **Dimensionality Reduction with SVD**

1. Dimensionality reduction using SVD involves retaining only the top kkk \_\_\_\_\_\_\_\_\_\_\_ and corresponding singular vectors.
   * **Answer**: singular values
2. A low-rank approximation of a matrix AAA is expressed as Ak=Uk∗Σk∗VkTA\_k = U\_k \* Σ\_k \* V\_k^TAk​=Uk​∗Σk​∗VkT​, where kkk is the \_\_\_\_\_\_\_\_\_\_\_ of the approximation.
   * **Answer**: rank
3. The reconstruction error in SVD-based dimensionality reduction is proportional to the \_\_\_\_\_\_\_\_\_\_\_ singular values excluded.
   * **Answer**: smallest

### **Noise Removal**

1. SVD can remove noise by truncating the \_\_\_\_\_\_\_\_\_\_\_ singular values, which typically represent noise.
   * **Answer**: smallest
2. True or False: In SVD-based noise removal, the high-magnitude singular values represent significant patterns.
   * **Answer**: True

### **Applications**

1. SVD is used to reduce the dimensionality of \_\_\_\_\_\_\_\_\_\_\_ matrices, such as in text or image data.
   * **Answer**: high-dimensional
2. In signal processing, SVD helps denoise data by removing \_\_\_\_\_\_\_\_\_\_\_ components.
   * **Answer**: irrelevant

### **Example**

1. If AAA is a 4 x 3 matrix, how many singular values are there in its decomposition?
   * **Answer**: 3
2. True or False: The low-rank approximation AkA\_kAk​ minimizes the Frobenius norm of A−AkA - A\_kA−Ak​.
   * **Answer**: True

### **Python Implementation**

1. Write a Python snippet to perform a rank-2 approximation of a matrix AAA.

**Answer**:  
python  
Copy code  
import numpy as np

A = np.array([[4, 2, 0], [2, 4, 2], [0, 2, 4]])

U, Sigma, VT = np.linalg.svd(A, full\_matrices=False)

k = 2

U\_k = U[:, :k]

Sigma\_k = np.diag(Sigma[:k])

VT\_k = VT[:k, :]

A\_k = np.dot(U\_k, np.dot(Sigma\_k, VT\_k))

print("Low-rank Approximation:\n", A\_k)

### **Miscellaneous**

1. The rank of a matrix AAA equals the number of non-zero \_\_\_\_\_\_\_\_\_\_\_ in ΣΣΣ.
   * **Answer**: singular values
2. True or False: SVD can reduce the storage requirements of a large dataset.
   * **Answer**: True

### **Related example questions from File 22: nullAltHyp.pdf**

### **Hypothesis Testing**

1. The null hypothesis (H0) typically states that there is \_\_\_\_\_\_\_\_\_\_\_ effect or difference.
   * **Answer**: no
2. The alternative hypothesis (HaH\_aHa​) is a statement that is accepted if the \_\_\_\_\_\_\_\_\_\_\_ hypothesis is rejected.
   * **Answer**: null
3. A two-sided test is used when the alternative hypothesis states that the parameter is \_\_\_\_\_\_\_\_\_\_\_ the null hypothesis value.
   * **Answer**: not equal to

### **Significance Level**

1. The significance level (α\alphaα) is the probability of making a \_\_\_\_\_\_\_\_\_\_\_ error.
   * **Answer**: Type I
2. True or False: A smaller significance level decreases the probability of rejecting a true null hypothesis.
   * **Answer**: True

### **Examples**

1. In a hypothesis test where H0:μ=100 and Ha:μ>100, the test is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: one-sided
2. A Type II error occurs when the null hypothesis is \_\_\_\_\_\_\_\_\_\_\_, but it is not rejected.
   * **Answer**: false

### **Applications**

1. Hypothesis testing is widely used in \_\_\_\_\_\_\_\_\_\_\_ research to validate experimental results.
   * **Answer**: scientific
2. True or False: The p-value represents the probability of observing data as extreme as the test statistic, assuming H0 is true.
   * **Answer**: True

### **Miscellaneous**

1. A test with H0:μ=10 and Ha:μ<10 is an example of a \_\_\_\_\_\_\_\_\_\_\_ test.
   * **Answer**: one-tailed
2. The power of a test is defined as 1−β where β is the probability of a \_\_\_\_\_\_\_\_\_\_\_ error.
   * **Answer**: Type II

### **Related example questions from File 23: oneSampleTtest (2).pdf**

### **One-Sample t-Test**

1. A one-sample t-test compares the mean of a sample to a \_\_\_\_\_\_\_\_\_\_\_ mean.
   * **Answer**: population
2. The t-statistic for a one-sample t-test is calculated as:
   * **Answer**: t = (x\_bar - mu) / (s / sqrt(n))
3. Degrees of freedom for a one-sample t-test are equal to \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: n - 1

### **Significance Testing**

1. If the p-value is less than the significance level (α\alphaα), the null hypothesis is \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: rejected
2. True or False: A one-sample t-test can be one-tailed or two-tailed.
   * **Answer**: True

### **Examples**

1. If xbar=10, mu=12 =, s=3, and n=9, compute the t-statistic.
   * **Answer**: -2.0
2. A researcher wants to test whether the mean weight of a population is 70 kg. The null hypothesis is:
   * **Answer**: H0:μ=70

### **Applications**

1. The one-sample t-test is used in quality control to verify if a machine produces items with the \_\_\_\_\_\_\_\_\_\_\_ specification.
   * **Answer**: target
2. True or False: The one-sample t-test assumes the data is approximately normally distributed.
   * **Answer**: True

### **Python Implementation**

1. Write a Python snippet to perform a one-sample t-test with a sample = [11, 12, 13, 14, 15] and population mean = 13.

**Answer**:  
python  
Copy code  
from scipy.stats import ttest\_1samp

sample = [11, 12, 13, 14, 15]

t\_stat, p\_value = ttest\_1samp(sample, popmean=13)

print("t-statistic:", t\_stat)

print("p-value:", p\_value)

### **Related example questions from File 24: threeStatisticalTests.pdf**

### **Two-Sample t-Test**

1. A two-sample t-test compares the means of two \_\_\_\_\_\_\_\_\_\_\_ samples.
   * **Answer**: independent
2. The null hypothesis for a two-sample t-test typically states that the means of the two samples are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: equal
3. The formula for the t-statistic in a two-sample t-test is:
   * **Answer**: t = (x1\_bar - x2\_bar) / sqrt((s1^2 / n1) + (s2^2 / n2))
4. Degrees of freedom in a two-sample t-test can be computed using the \_\_\_\_\_\_\_\_\_\_\_ formula.
   * **Answer**: Welch-Satterthwaite

### **Paired t-Test**

1. A paired t-test is used to compare the means of two \_\_\_\_\_\_\_\_\_\_\_ samples.
   * **Answer**: related (or dependent)
2. In a paired t-test, the t-statistic is computed based on the \_\_\_\_\_\_\_\_\_\_\_ differences between paired observations.
   * **Answer**: mean
3. True or False: A paired t-test assumes that the differences between paired observations are normally distributed.
   * **Answer**: True

### **Wilcoxon Rank-Sum Test**

1. The Wilcoxon rank-sum test is a \_\_\_\_\_\_\_\_\_\_\_ alternative to the two-sample t-test.
   * **Answer**: nonparametric
2. The Wilcoxon test is used when data are \_\_\_\_\_\_\_\_\_\_\_ and do not meet the assumptions of normality.
   * **Answer**: ordinal (or non-normal)
3. True or False: The Wilcoxon rank-sum test compares medians instead of means.
   * **Answer**: True

### **Chi-Square Test**

1. The chi-square test is used to test the independence of two \_\_\_\_\_\_\_\_\_\_\_ variables.
   * **Answer**: categorical
2. The formula for the chi-square statistic is:
   * **Answer**: chi^2 = sum((O - E)^2 / E)
3. True or False: The chi-square test requires expected frequencies to be at least 5 in each category.
   * **Answer**: True

### **Examples**

1. In a two-sample t-test, if x1bar=20, x2bar=15, s1=5, s2=4, n1=10, and n2=12, compute the t-statistic.
   * **Answer**: 3.35
2. In a chi-square test, the observed frequencies are [[30, 20], [50, 40]]. The expected frequencies are [[40, 10], [40, 50]]. Compute the chi-square statistic.
   * **Answer**: 25

### **Applications**

1. The two-sample t-test is commonly used to compare test scores between two \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: groups
2. The Wilcoxon rank-sum test is often used in \_\_\_\_\_\_\_\_\_\_\_ studies where data does not meet parametric assumptions.
   * **Answer**: medical
3. The chi-square test is used to determine if education level and annual income are \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: related

### **Python Implementation**

1. Write a Python snippet to perform a two-sample t-test for samples [10, 15, 20] and [12, 17, 22].

**Answer**:  
python  
Copy code  
from scipy.stats import ttest\_ind

sample1 = [10, 15, 20]

sample2 = [12, 17, 22]

t\_stat, p\_value = ttest\_ind(sample1, sample2)

print("t-statistic:", t\_stat)

print("p-value:", p\_value)

1. Write a Python snippet to perform a chi-square test for observed frequencies [[30, 20], [50, 40]] and expected frequencies [[40, 10], [40, 50]].

**Answer**:  
python  
Copy code  
from scipy.stats import chi2\_contingency

observed = [[30, 20], [50, 40]]

chi2, p\_value, \_, \_ = chi2\_contingency(observed)

print("Chi-square:", chi2)

print("p-value:", p\_value)

### **Miscellaneous**

1. The p-value in a chi-square test indicates the likelihood of observing the data if the \_\_\_\_\_\_\_\_\_\_\_ hypothesis is true.
   * **Answer**: null
2. True or False: The Wilcoxon rank-sum test assumes that the data are independent.
   * **Answer**: True
3. The degrees of freedom for a chi-square test are computed as \_\_\_\_\_\_\_\_\_\_\_.
   * **Answer**: (number of rows - 1) \* (number of columns - 1)