Mathematical Foundations of Data Science

Linear Algebra

Satishkumar L. Varma

Professor, Department of Information Technology
PCE, New Panvel
Scopus | Web of Science | Google Scholar | Google Site | Website

PILLAI COLLEGE OF ENGINEERING
Autonomous

Linear Algebra: Outline

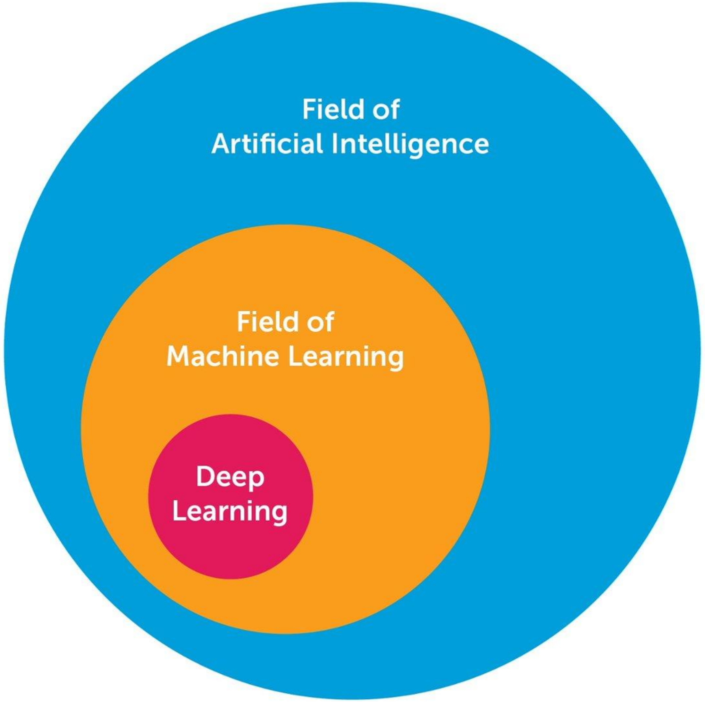
- Linear Algebra
- Components of linear algebra
- Significance of linear algebra in Data Science
- Scalar and Vector: Basis Vectors, Orthogonal Vectors, Orthonormal Vectors, Vector Spaces,
- Tensor: Types of Tensors, Matrix, Matrix Types.

Linear Algebra: Learning Objectives & Outcomes

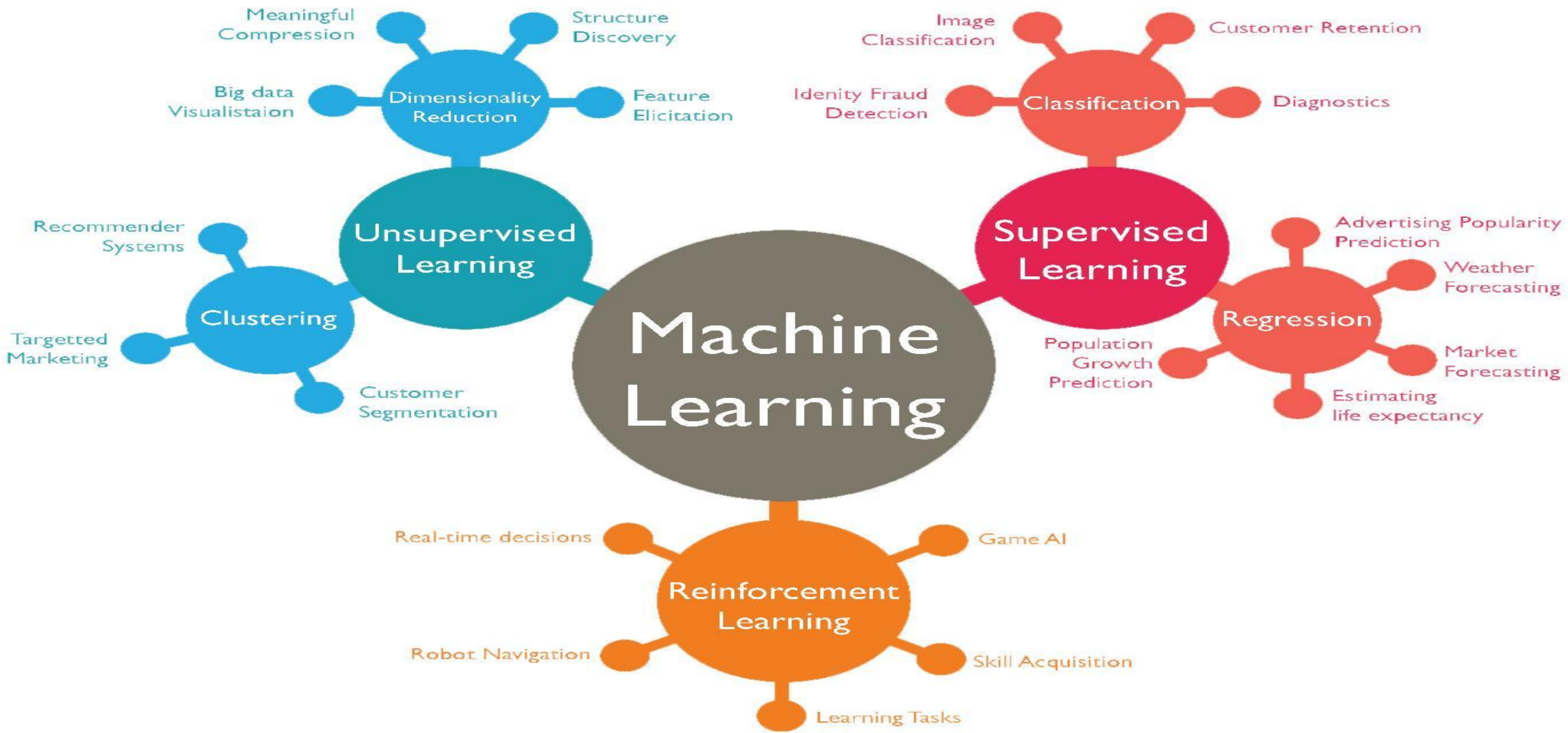
- **Learning Objectives:** Course Instructor or Faculty aims
  - To describe the mathematics used in ML and DL by data scientists.

Linear Algebra: Introduction

- Data Science: Interdisciplinary field of scientific methods, processes, algorithms and systems to extract knowledge or insights from data
    - Artificial intelligence
    - Machine learning
    - Deep learning



Linear Algebra: Introduction



https://sites.google.com/view/vsat2k/courses/AI-and-Data-Analytics

Linear Algebra: Introduction

- ML and DL algorithm require a mathematical tool - Linear Algebra
- What is Linear Algebra?
- Linear Algebra is a branch of mathematics
- It deals with vectors, vector spaces, linear transformations, and matrices.
- These entities can be used to depict and solve systems of linear equations, among other tasks.
- Following are basic terms required to understand linear algebra:
- **Vector**
- **Matrix**
    - Vector Spaces
    - Tensor
    - Linear Transformations
    - Dot Product
    - Matrix Multiplication
    - Matrix Addition and Subtraction
    - Determinant and Inverse

Linear Algebra: Introduction

- **Scalar**: A What is a Scalar?
- A scalar is the simplest form of a tensor.
- It’s a single number, without direction.
- Scalars contrast with higher order tensors like vectors (1st order), matrices (2nd order), and so on.
- In other words, a scalar has zero dimensions.
- Why are Scalars Important?
- **Foundation of Math Operations:**
  - When we work with high-dimensional data, the operations often boil down to scalar computations.
  - For instance, when you multiply two matrices, the individual operations involve multiplying scalars.
- **Understanding Basic Properties:**
  - Concepts like magnitude, units, and identity elements are best understood using scalars before they’re applied to vectors and matrices.
- **Performance Metrics:** In ML, metrics such as loss, accuracy, or precision are often represented as scalars.

Linear Algebra: Introduction

- **Scalar**: Interactions with Higher-Order Tensors
- Scalars frequently interact with vectors, matrices, and higher-order tensors. For instance:
- Scalar Multiplication: Multiplying a matrix by a scalar involves multiplying each element of matrix by the scalar.
- **NumPy:**
- matrix = np.array([[1, 2], [3, 4]])
- result = matrix * 5
- print(result)
- Code Output
  - [[5 10]
  - [15 20]]

Linear Algebra: Introduction

- **Scalar**: Advanced Properties of Scalars
- **Identity**: The number 1 is often called a multiplicative identity because multiplying any number by 1 doesn't change that number. Likewise, 0 is an additive identity because adding 0 to a number doesn't change it.
- **Inverse**: Every scalar has a multiplicative inverse, such that when it's multiplied by its inverse, the result is 1. For example, the inverse of 5 is 1/5
- **Absolute Value**: It represents the magnitude of a scalar. In many libraries, it's computed using the abs function.
- **Scalar**: Scalars in Different Frameworks
- Illustrating scalar operations in three prominent libraries:
  - Scalars in NumPy: import numpy as np
  - Scalars in PyTorch: import torch
  - Scalars in TensorFlow: import tensorflow as tf

Linear Algebra: Introduction

$$F = \begin{bmatrix} v1 \\ v2 \\ v3 \end{bmatrix} \quad \text{or} \quad F = \begin{bmatrix} v1 & v2 & v3 \end{bmatrix}$$

- **Vector**: A vector is a one-dimensional array of numbers.
- For instance, in a 2D space, a vector v can be represented as v = [2, 3],
  - pointing 2 units in the x-direction and 3 units in the y-direction.
- In the world of data science, vectors play a vital role.
- They are fundamental to machine learning, data analysis, and AI.
- DS needs an understanding the concepts of vectors, vector transposition, norms, and unit vectors
- Vectors, also referred to as first-order tensors, are mathematical objects used to represent quantities that have both magnitude and direction.
- They are an ordered list of numbers that can describe anything from motion to the weights in a NN.
- Properties of Vectors
- Commutativity: A + B = B + A
- Associativity: (A + B) + C = A + (B + C)
- Distributive Property: a(A + B) = aA + aB, where ‘a’ is a scalar.

Linear Algebra: Introduction

- **Vector**: A vector is a one-dimensional array of numbers.
- Vectors are a foundational concept in linear algebra and have broad applications across the sciences.
- Among the most fundamental ideas related to vectors are the concepts of basis, orthogonality, and orthonormality.
- Basis Vectors
- Orthogonal Vectors
- Orthonormal Vectors

Linear Algebra: Introduction

- **Vector**: 1. Basis Vectors
- In linear algebra, a set of vectors is considered a basis for a vector space if:
  - They are linearly independent:
    - No vector can be represented as a linear combination of the other vectors.
  - They span the space:
    - Any vector in the space can be expressed as a linear combination of these basis vectors.
  - Data Science Context:
    - When dealing with datasets, you can think of each data point as a vector in some space.
    - Basis vectors can serve as a "coordinate system" for this space.
    - PCA, a popular dimensionality reduction technique, essentially tries to find a new basis for representing data where each basis vector (or principal component) captures a decreasing amount of variance in the dataset.

Linear Algebra: Introduction

- **Vector**: 1. Basis Vectors
- Example: Suppose you have data points in 3D space, and you want to represent them in 2D without much loss of information. PCA will find two basis vectors (the first two principal components) that capture the most variance, allowing you to project your data onto these vectors and move from 3D to 2D.
- Consider the vectors in R^2
- e1 = [1, 0] and e2 = [0, 1]
- These vectors are a basis for R^2 because any vector in R^2 can be written as a combination of e1 and e2.

Linear Algebra: Introduction

- **Vector**: 2. Orthogonal Vectors
- Two vectors are orthogonal if their dot product is zero.
- This means they are perpendicular to each other in geometric terms.
- Data Science Context:
  - Orthogonality is valuable in data science because orthogonal features (or vectors) are uncorrelated.
  - This property is especially useful in regression analysis,
  - where multicollinearity (correlation among predictor variables) can distort results and interpretations.
- **Example**: Imagine you have two features: height and height_squared. These features are not orthogonal since they are correlated. But height and, say, weight, could be orthogonal if they are uncorrelated in your datase
- For vectors u and v, if $u \cdot v = 0$, then u and v are orthogonal.
- Consider the vectors: $u = [1, 2]$ and $v = [2, -1]$
- The dot product is: $u.v = 1(2) + 2(-1) = 0$
- Thus, u and v are orthogonal.

Linear Algebra: Introduction

- **Vector**: 3. Orthonormal Vectors
- Orthonormal vectors are not only orthogonal but also of unit length (magnitude of 1).
- An orthonormal set of vectors provides a “normalized” basis for a space.
- They are orthogonal to each other.
- Each vector has a magnitude (or length) of 1.
- Data Science Context: Orthonormal vectors simplify computations as their lengths are 1, and they are mutually perpendicular. This property is heavily leveraged in algorithms like the Gram-Schmidt process, which orthogonalizes a set of vectors and then normalizes them to generate an orthonormal set.
- Example: In the aforementioned PCA, the principal components are not only orthogonal but also orthonormal.
  - This means that when you project your data onto these components, the length of the projections remains unchanged, ensuring that the relative distances between data points are maintained.
  - These vectors u and v are orthonormal because:
  - Their dot product is zero, making them **orthogonal**.
- The magnitude of each vector is 1.

$$u = [1/sqrt(2), 1/sqrt(2)]$$

$$v = [-1/sqrt(2), 1/sqrt(2)]$$

Linear Algebra: Introduction

- **Vector**: Basic Operations with Vectors
- Addition: When you add vectors, you place them head to tail and draw the resultant vector from the tail of the first vector to the head of the last.
- Subtraction: This is equivalent to adding a negative vector.
- Scalar Multiplication: Multiplying a vector by a scalar changes its magnitude but not its direction.
- Dot Product (Scalar Product): The result is a scalar. For two vectors A and B, $A \cdot B = |A||B|\cos(\theta)$, where θ is the angle between the vectors.
- Cross Product (Vector Product): The result is a vector. For vectors A and B in 3D, the magnitude is $|A||B|\sin(\theta)$, and the direction is perpendicular to both A and B, following the right-hand rule.

Linear Algebra: Introduction

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

- Matrix: A matrix is a m x n, two-dimensional array of numbers.
- It's essentially a collection of vectors.
- Matrix theory is a fundamental aspect of linear algebra and plays a crucial role in a variety of scientific and engineering domains.
- While we often start with general matrices, certain types possess unique properties that make them essential in both theoretical and practical applications.
- You can think of numbers arranged in rows and columns.
- A matrix is a rectangular arrangement of numbers or symbols, structured in rows and columns.
- For example, the matrix A has 2 rows and 2 columns, classifying it as a 2×2 matrix.
- For example: Here, the matrix B has three rows and two columns.
- B = [2 3]
- [4 5]
- [6 7]

Linear Algebra: Introduction

- Matrix: Matrix Types
- **Diagonal Matrix:** A square matrix in which all off-diagonal entries are zero.
  - In other words, only the main diagonal might have non-zero elements. Example: Matrix D
- **Scalar Matrix:** A diagonal matrix where all the diagonal entries are the same.
  - It's a special case of the diagonal matrix. Example: Matrix E
- **Identity Matrix:** A scalar matrix where all the diagonal entries are 1.
  - It acts as the multiplicative identity in matrix multiplication. Example: Matrix I
- **Zero Matrix:** A matrix in which all entries are zero. Example: Matrix O
- **Symmetric Matrix:** A matrix that is equal to its transpose, i.e., (A = A^T).
  - Its entries are symmetric about the main diagonal. Example: Matrix S
- **Anti-Symmetric Matrix:** A matrix is anti-symmetric if (A = -A^T).
  - This means that if you take the transpose of the matrix and then change the sign of every element, you get back the original matrix. Example: Matrix B
- **Hermitian Matrix:** In the context of complex matrices, a matrix is Hermitian if (A = A^H), where (A^H) is the conjugate transpose (or adjoint) of matrix (A).
  - It means taking the transpose of matrix and then taking the complex conjugate of each entry.
  - This matrix has real diagonal entries, and its eigenvalues are also real. Example: Matrix C

$$D = \begin{bmatrix} 5 & 0 \\ 0 & 7 \end{bmatrix}$$

$$E = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}$$

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$O = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$S = \begin{bmatrix} 1 & 4 \\ 4 & 3 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 4 \\ -4 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 3 & 2+i \\ 2-i & 1 \end{bmatrix}$$

Linear Algebra: Introduction

- Matrix: Importance and Applications:
- Diagonal and Scalar Matrices: These matrices are crucial for simplifying matrix operations. When multiplying, for instance, diagonal matrices scale rows/columns without mixing them, which can be computationally efficient in certain algorithms.
- Identity Matrix: Acts as the “neutral element” in matrix multiplication. When a matrix is multiplied by the identity matrix, it remains unchanged. This property is fundamental in various algorithms and mathematical proofs.
- Zero Matrix: Represents the additive identity in matrix addition. Adding any matrix to the zero matrix yields the original matrix.
- Symmetric and Anti-Symmetric Matrices: These are significant in various branches of mathematics and physics. In physics, symmetric matrices often arise when defining certain types of operators. Anti-Symmetric matrices have applications in fields such as differential geometry.
- Hermitian Matrices: Fundamental in quantum mechanics where they represent observable quantities, given their property of having real eigenvalues.

Linear Algebra: Introduction

- Vector Spaces:
- A vector space is a set of vectors that adhere to specific rules when undergoing addition and scalar multiplication.
- For a set to qualify as a vector space, it must satisfy properties like commutativity, associativity, and distributivity.
- Tensor: What are tensors?
- An n-dimensional array where n can be more than 2.
- Tensors is a mathematical entity that form the base for ML and AI.
- It is generalization of scalars, vectors, and matrices that can accommodate higher dimensions.
- We can think of tensors as a higher-dimension generalization.
- Vectors are first-order tensors, and matrices are second-order tensors.
- Higher order tensors simply extend this structure into more dimensions.
- Linear Transformations:
- Transformations between vector spaces while preserving the operations of vector addition and scalar multiplication.
- Matrices can represent these transformations.

Linear Algebra: Introduction

- Dot Product:
  - This is the sum of the products of corresponding elements of two vectors.
  - For vectors a = [a1, a2] and b = [b1, b2], the dot product is a1*b1 + a2*b2.
  - Example: For vectors v1 = [2,3] and v2 = [4,5], the dot product is 2*4 + 3*5 = 8 + 15 = 23.
- Matrix Multiplication:
  - Involves taking the dot product of rows of the first matrix with columns of the second matrix.
  - Example: To multiply matrices A and B, the entry in the 1st row, 1st column of the resulting matrix is the dot product of the 1st row of A and the 1st column of B.
- Matrix Addition and Subtraction:
  - Addition: The element at row i, column j in the resulting matrix is the sum of the elements at row i, column j in the two matrices being added.
  - Subtraction: Same as addition but with subtraction of elements.
- Determinant and Inverse:
  - Determinant: A scalar value that indicates the “volume scaling factor” of a linear transformation.
  - Inverse: If matrix A’s inverse is B, then the multiplication of A and B yields the identity matrix.

Linear Algebra: Introduction

- Types of Tensors
- 1. Scalar or Zero-Order Tensor
- 2. Vector or First-Order Tensor
- 3. Matrix or Second-Order Tensor
- 4. Third-Order Tensors and Beyond
- **1. Scalar or Zero-Order Tensor**
- A scalar is the simplest type of tensor that we encounter regularly in mathematics and physics.
- A scalar is a single number, like 3 or -2.7.
- In the context of tensors, we call it a zero-order tensor because it contains only a single value and has no direction.
- Example:
  - a = 5

Linear Algebra: Introduction

- Types of Tensors
- **2. Vector or First-Order Tensor**
- A vector is a first-order tensor and can be thought of as a list of numbers.
- In the physical world, vectors represent quantities that have both magnitude and direction, such as velocity or force.
- In ML, vectors are often used to represent features of data or weights in neural networks.
- Example:
  - V = [3, 5, 4]

Linear Algebra: Introduction

- Types of Tensors
- 3. Matrix or Second-Order Tensor
- A matrix is a grid of numbers arranged in rows and columns.
- Matrices are second-order tensors.
- They are often used in ML to represent datasets and transformations, for instance, rotation or scaling.
- Example:

$$V = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

- 4. Third-Order Tensors and Beyond
- When we go beyond second-order tensors, we start dealing with entities that are challenging to visualize because they occupy more than 3D.
- But they are essential in many computational problems, such as DIP,
  - where each pixel's color is represented in a 3D vector (red, green, blue), and
  - the image itself is a 2D grid of these vectors, resulting in a 5D tensor.
- Example:

$$\left[\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 5 & 6 \\ 7 & 7 \end{bmatrix} \right]$$

Linear Algebra: Introduction

- Why is Linear Algebra Essential for Data Scientists? Why Linear Algebra matters for Data Scientists.
- **Foundational to Machine Learning:**
  - ML and DL rely heavily on linear algebra.
  - Data is stored and worked on such objects.
  - The idea of matrices and tensors is present everywhere in the world of AI
  - Concepts like transformations, eigenvalues, and eigenvectors are used in algorithms like PCA, t-SNE
  - These algorithms are used for **dimensionality reduction** of data and visualizing high dimensional data.
- Data Representation:
  - In data science, data is often represented as matrices or tensors (multi-dimensional arrays).
  - For example, an image in a computer can be represented as a matrix of pixel values.
  - Understanding how to manipulate these matrices is needed for many data tasks.
- Memory Efficient computations:

Linear Algebra: Introduction

- Why is Linear Algebra Essential for Data Scientists? Why Linear Algebra matters for Data Scientists.
- **Memory Efficient computations:**
    - Operations on matrices and vectors can be highly optimized in modern computational libraries.
    - Knowing how to use linear algebra allows one to tap into these optimizations, making computations faster and more memory efficient.
    - Libraries like NumPy (in Python) or MATLAB are grounded in linear algebra.
    - These tools are staples in the data science toolbox, and they are designed to handle matrix operations efficiently.
- Conceptual Understanding:
    - Beyond the computational benefits, a solid grasp of linear algebra provides a deeper conceptual understanding of many data science techniques.
    - For example, understanding the geometric interpretation of vectors and matrices can provide intuition about why certain algorithms work and how they can be improved.

Linear Algebra: Introduction

- Why is Linear Algebra Essential for Data Scientists? Why Linear Algebra matters for Data Scientists.
- **Optimization:**
  - Optimization problems in ML and statistics, like LR, can be formulated and solved using linear algebraic techniques.
  - Techniques such as gradient descent involve vector and matrix calculations.
  - Techniques such as ridge and lasso regression employ linear algebra for regularization to prevent overfitting.
- **Signal Processing:**
  - For those working with time series data or images, DFT and convolution operations, which are rooted in linear algebra, are crucial.
- **Network Analysis:**
  - If you're working with graph data or network data, the adjacency matrix and the Laplacian matrix are foundational, and understanding their properties requires knowledge of linear algebra.

Linear Algebra: Applications

- Use of Linear Algebra in Machine Learning Algorithms
- Neural Networks:
  - Neural networks are made of data connections called Neurons.
  - Each neuron’s output is nothing but a linear transformation (via weights, which are matrices) of the input, passed through an activation function.
  - This is mostly multiplication operations of linear algebra.
- Support Vector Machines:
  - SVM’s use the dot product to determine the margin between classes in classification problems.
  - So, Linear algebra is used.
- Image Processing:
  - Filters applied to images are matrices that transform the pixels.

Linear Algebra: Applications

- Use of Linear Algebra in Machine Learning Algorithms
- Principal Component Analysis (PCA):
  - PCA is a shining application of linear algebra.
  - At its core, PCA is about finding the “principal components” (or directions) in which data varies the most.
  - Step-by-Step Process (each involve linear algebra):
    - **Standardization**: Ensure all features have a mean of 0 and standard deviation of 1.
    - **Covariance Matrix Computation**: A matrix capturing the variance between features.
    - **Eigendecomposition**: Find the eigenvectors (the principal components) and eigenvalues of the covariance matrix.
    - **Projection**: Data is projected onto the top eigenvectors, reducing its dimensions while preserving as much variance as possible.

Linear Algebra: Applications

- **Use of Tensors**
- In ML, tensors are the fundamental data structure used in libraries like TensorFlow and PyTorch.
    - They are employed to encode the inputs and outputs of models, and even the models themselves.
- In DS and BDA, multi-dimensional tensors play a crucial role in organizing and processing large datasets for analysis.
- **Data Representation**
    - Machine Learning models, at their core, work with numeric data.
    - Whatever the input data, be it text, images, or sound, all of it is ultimately converted into numbers.
    - This is where tensors become important.For instance, a color image can be represented as a 3D tensor.
    - The height and width of the image are represented by the first two dimensions, while the color channels (typically Red, Green, Blue) are represented by the third dimension.
- **Efficient Computation**
    - Tensors, especially in software libraries like TensorFlow and PyTorch, are designed to be highly efficient for large-scale computations.
    - These libraries allow for operations on tensors to be executed on GPU (Graphics Processing Unit), leading to significantly faster computations.

Summary

- Linear algebra concepts is used in Data Science, day in and day out.
- It offers a framework to manipulate, transform, and interpret data, making it essential for various ML algorithms and processes.
- Tensors, in all their shapes and forms, serve as essential tools in various disciplines.
- Understanding the different types of tensors enables us to comprehend how they function in different contexts, from encoding information in ML models to explaining the fundamental principles of the universe in physics.
- As you advance in data science, knowing the foundation in linear algebra will help.
- Special matrix types simplify computations and provide insights into the underlying structure of mathematical problems.
- By understanding these matrices and their properties, we gain a valuable toolset for addressing a multitude of challenges across disciplines.
- Understanding the nuances between basis, orthogonal, and orthonormal vectors is key for various applications across mathematics and science.

References

A. Text Books:

1. P. Bruce and A. Bruce, Practical Statistics for Data Scientists: 50 Essential Concepts .
2. C. O'Neil and R. Schutt, Doing Data Science, O'Reilly.

B. References:

3. G. Strang, Introduction to Linear Algebra, 5th edition, Wellesley-Cambridge Press, USA.
4. W. Hines, D. Montgomery, D. Goldman, C. Borror, Probability and Statistics in Engineering, Wiley India Pvt. Ltd.
5. A. Agresti, C. Franklin, B. Klingenberg, Statistics: The Art and Science of Learning from Data, Global Edition, Pearson.

Thank You.