Team 66

June 2023

Linear Algebra Project

Team 66

Team Members

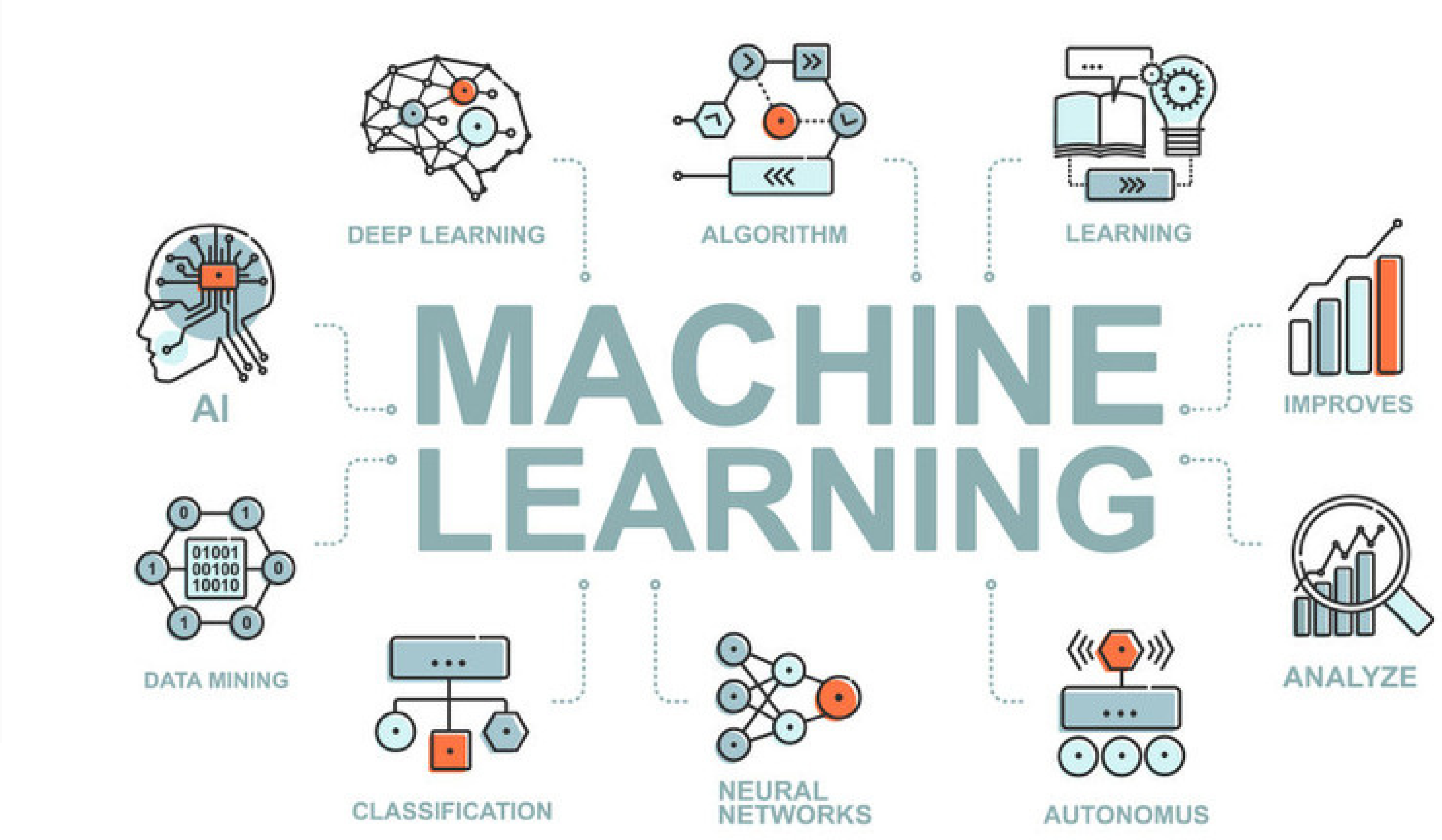
1 Chinmay Sharma

2 Chanukya Charyulu SVSK

3 Beerelly Vishwanth

Introduction

Project Topic: Linear Algebra and Its Applications in Machine Learning



Mathematical Background

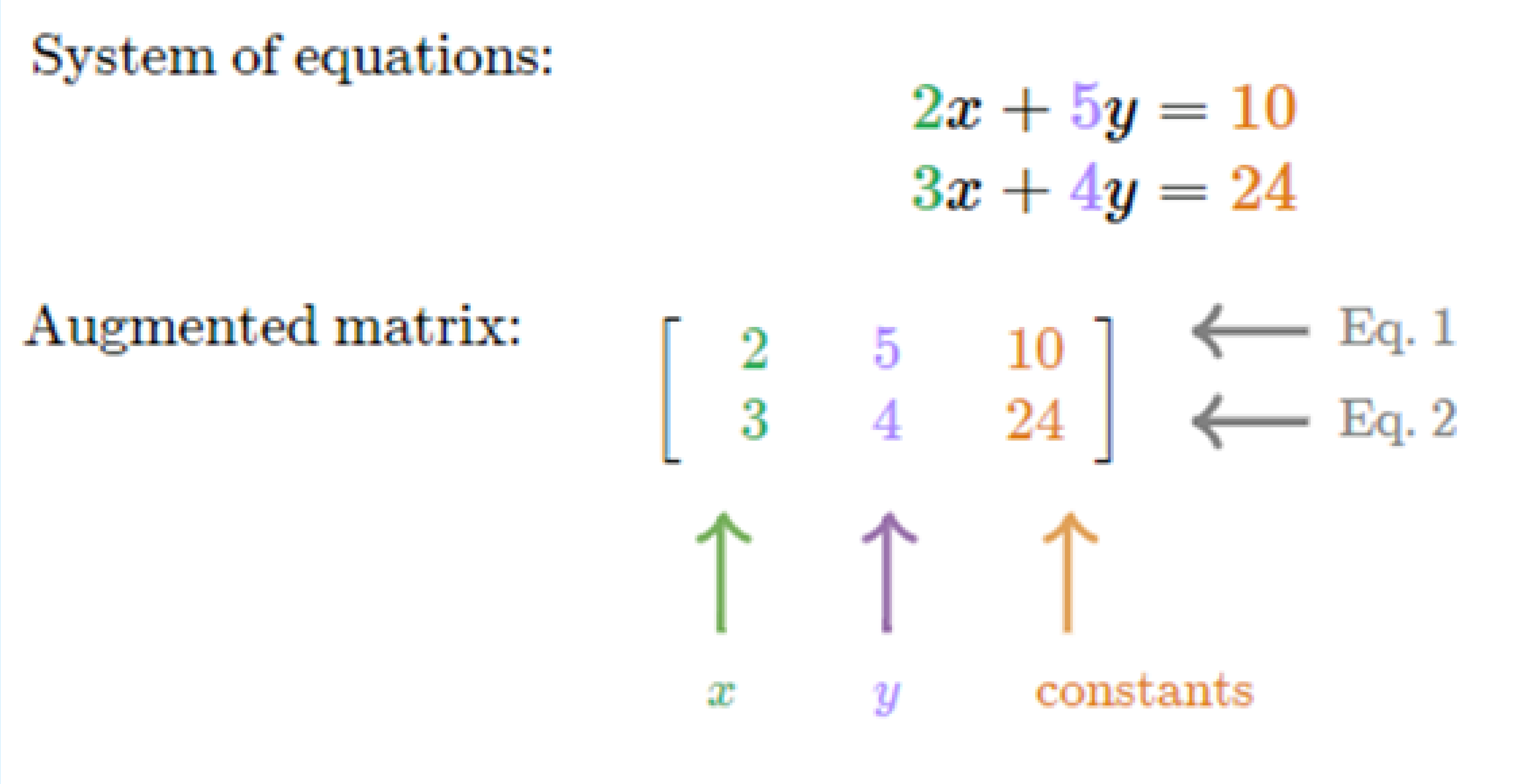
1. Using Matrices to Solve Linear Equations
2. Matrices as Linear Transformations
3. Orthogonalization and Norms
4. Eigenvalues, Eigenvectors and Diagonalization
5. Matrix Decompositions
6. Statistics and the Covariance Matrix
7. Vector Analysis
8. Optimization Theory

Machine Learning Applications

1. Overview of Machine Learning Terminology and Methods
2. Regression Analysis
3. Latent Semantic Analysis
4. Principal Components Analysis
5. Support Vector Machines
6. Neural Networks

Let's begin!

Matrices to Solve Linear Equations



In General, a system of m linear equations in n variables can be represented using the following matrices:

1. coefficient matrix of the order mxn
2. variable matrix of the order nx1
3. constant matrix of the order mx1

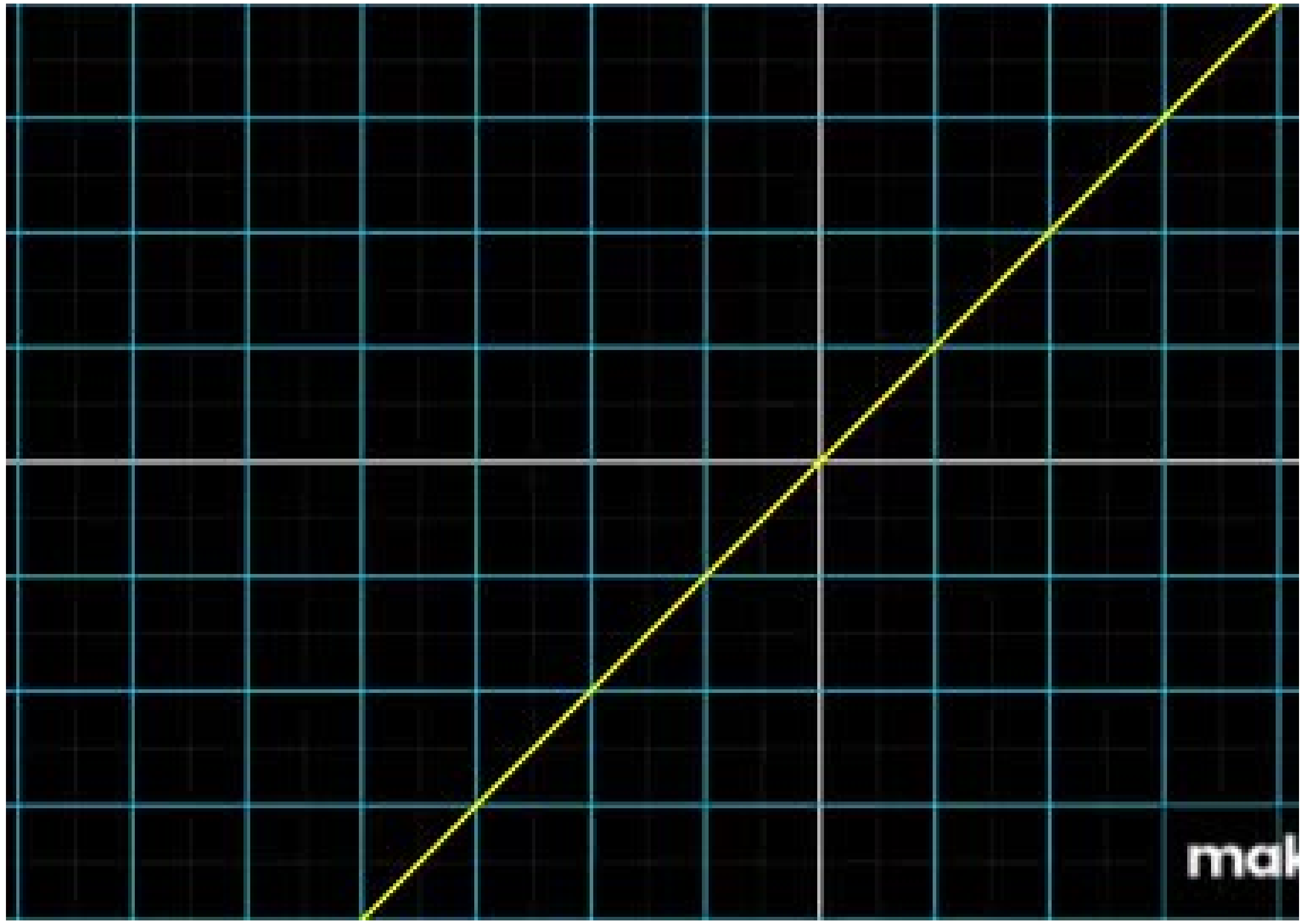
OR

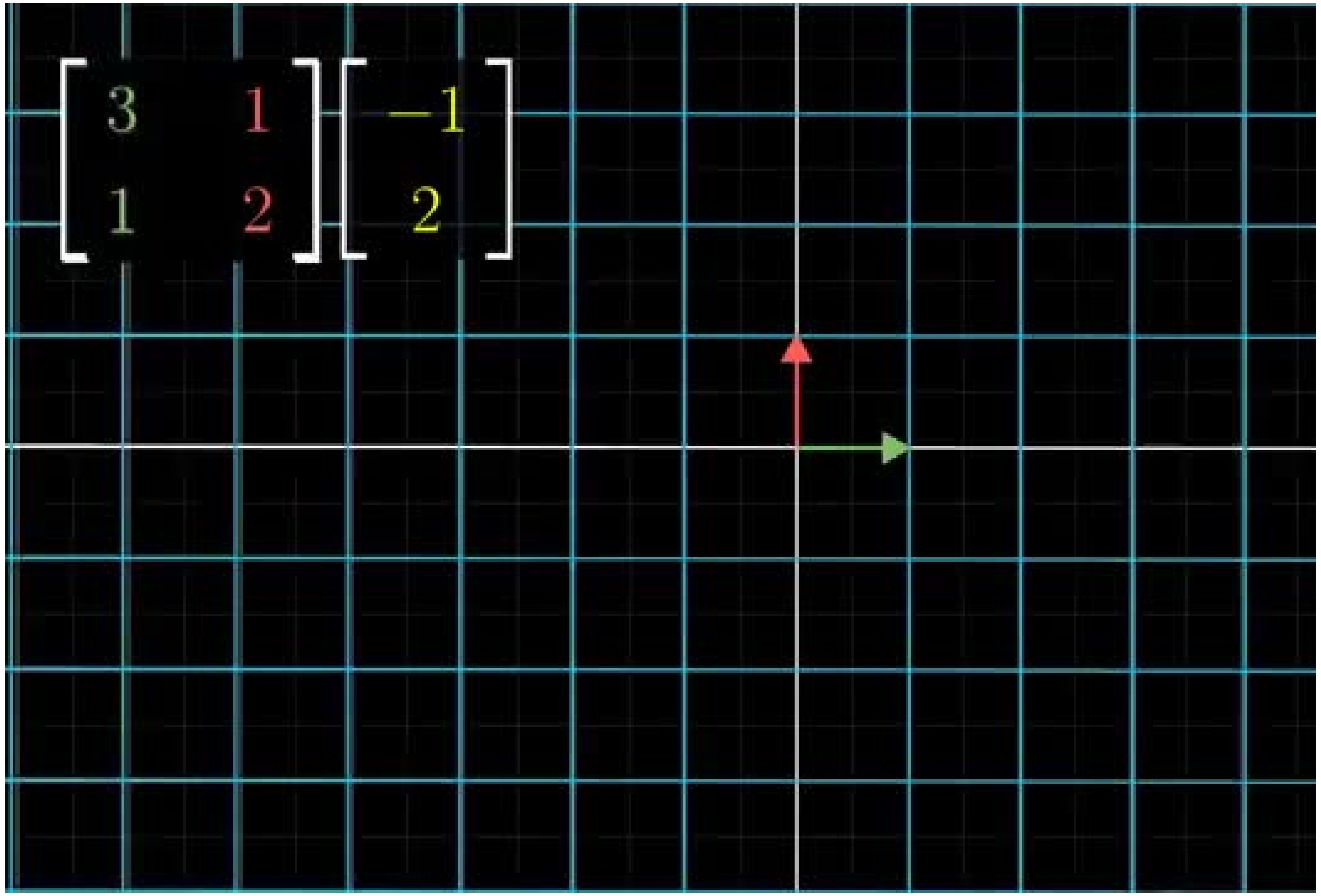
An augmented of the order mx(n+1)

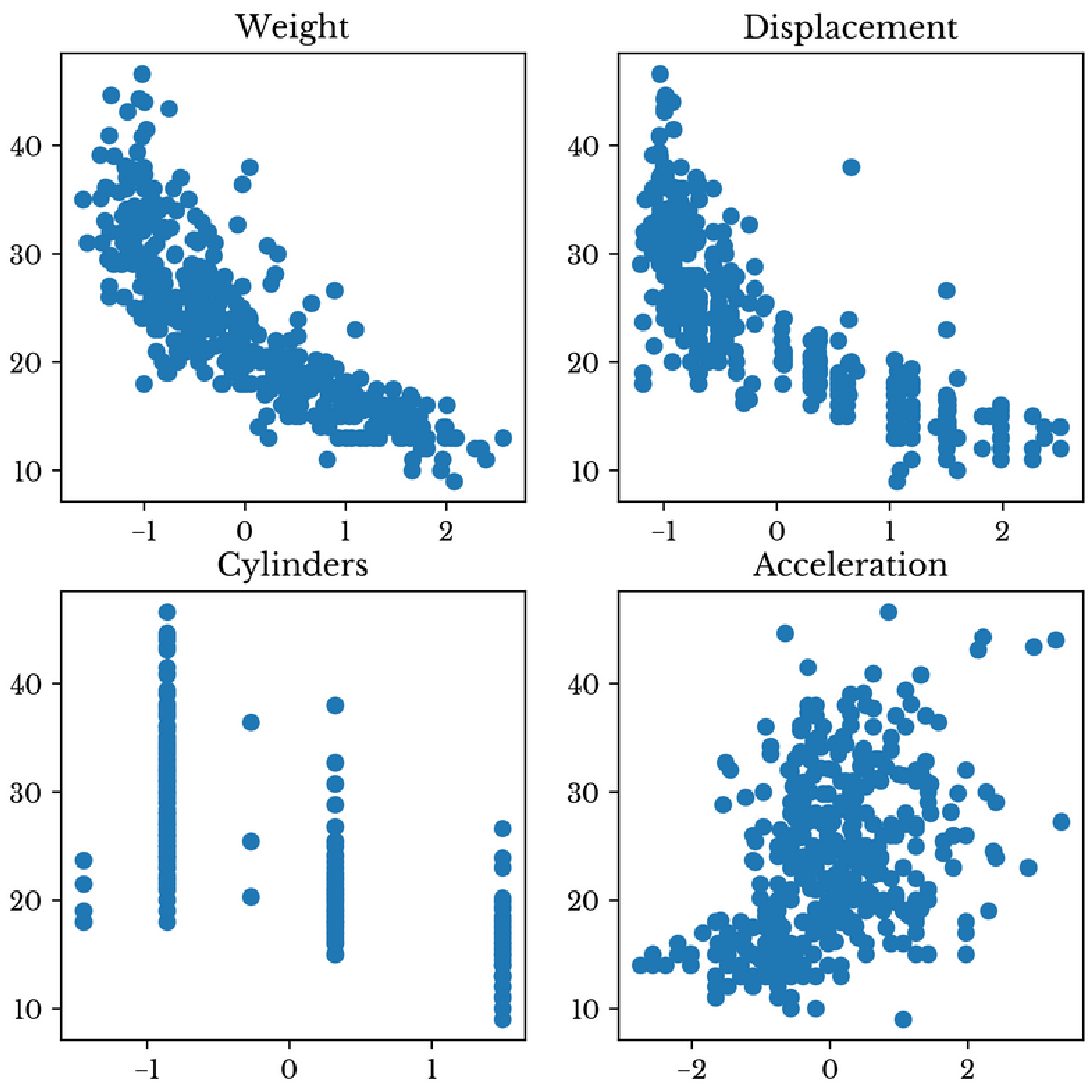
Also the solutions can be found using Gaussian Elimination technique which is:

1. Transform the augmented matrix into its row reduced echelon form
2. Using the Rouche-Capelli theorem to find the number of solutions givensystemof equations have
3. after that, solving the row reduced echelon matrix can be solved with much more ease.

Matrices as Linear Transformations







Orthogonalization and Norms

Orthogonality refers to vectors being perpendicular or independent, normalized vectors have a magnitude of 1, and projection represents the component of a vector along another vector.

FUNDAMENTALS USED

Inner product | **Inner Product Space** | **Orthogonality** | **Norm** | **Normalized Vectors**

Inner product

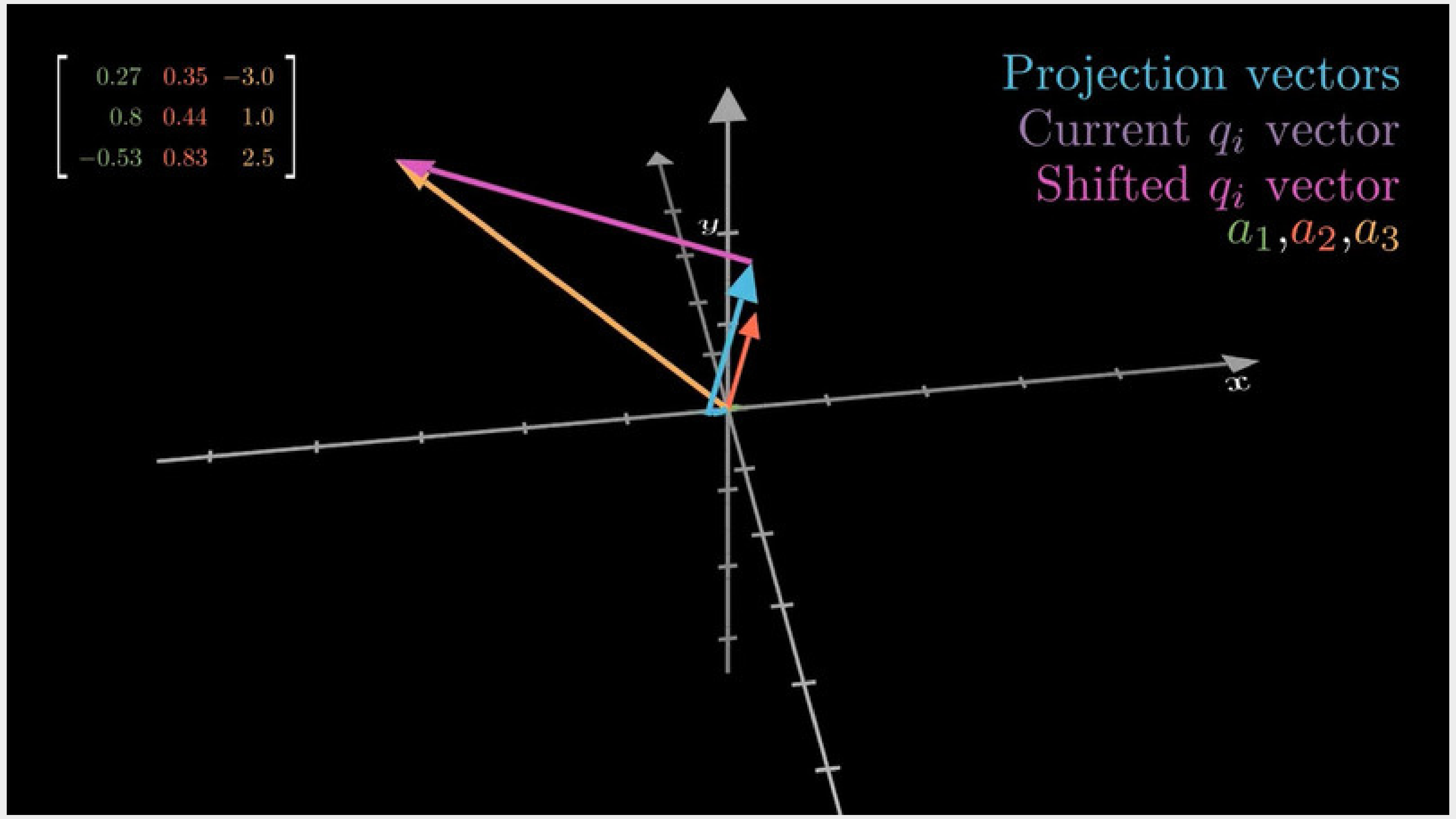
- Linearity in the First Argument
- Symmetry
- Positive Definiteness

Norm

- Non-Negativity
- Definiteness
- Homogeneity
- Triangle Inequality

Gram-Schmidt Process

The Gram-Schmidt orthogonalization procedure converts a list of linearly independent vectors into an orthonormal list or a basis into an orthonormal basis.



QR-Decomposition

QR-Decomposition was developed by John G. F. Francis

Existence and Uniqueness: Given an $m \times n$ matrix A with linearly independent columns, the QR decomposition exists and is unique

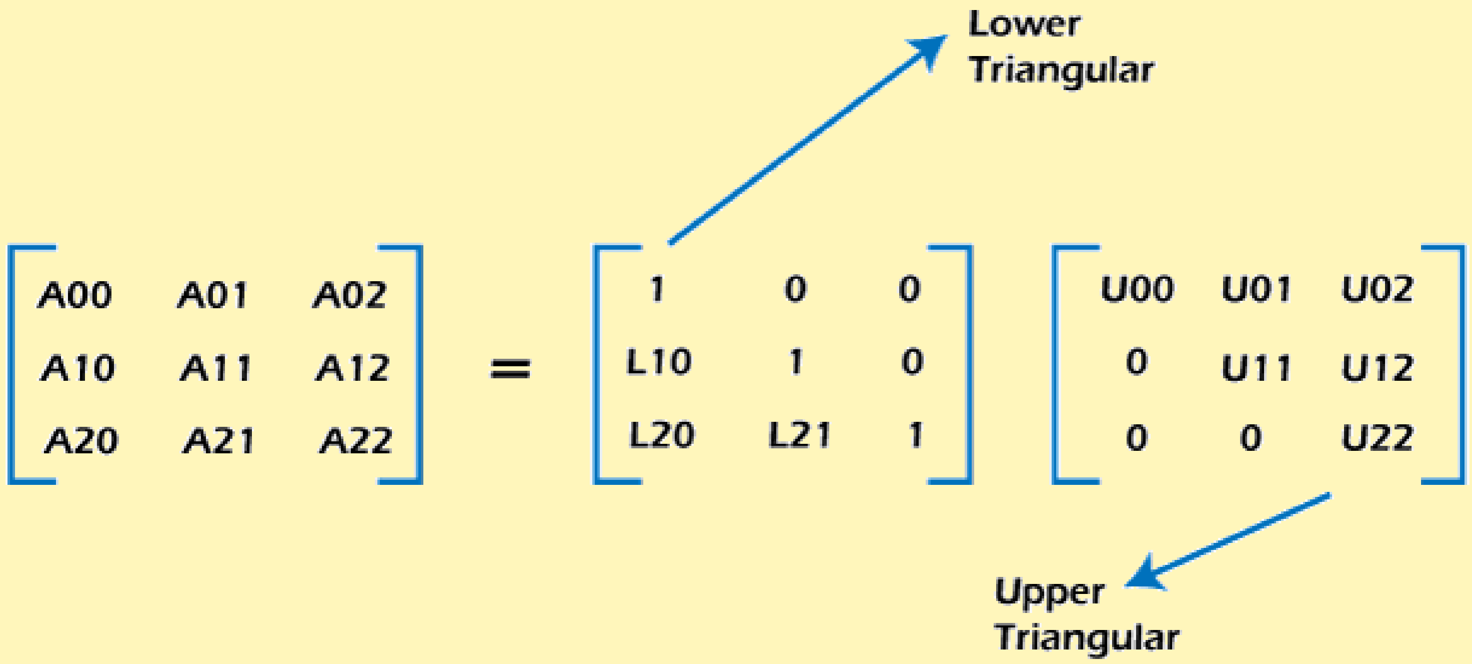
The QR decomposition in linear algebra factorizes a matrix A into the product of an orthogonal matrix Q and an upper triangular matrix R, i.e., $A = QR$. This decomposition is useful in various applications, such as solving linear systems of equations and least squares problems.

LU Decomposition:

- Developed by Alan Turing for Turing machine
- Unit lower and upper Triangular matrices
- LU Factorization with partial pivoting, complete pivoting, diagonal decomposition
- Types of LU Factorization:

1. partial pivoting (LUP) -- PA=LU
2. Full pivoting -- PAQ=LU
3. Lower-diagonal-upper(LDU) -- A=LDU

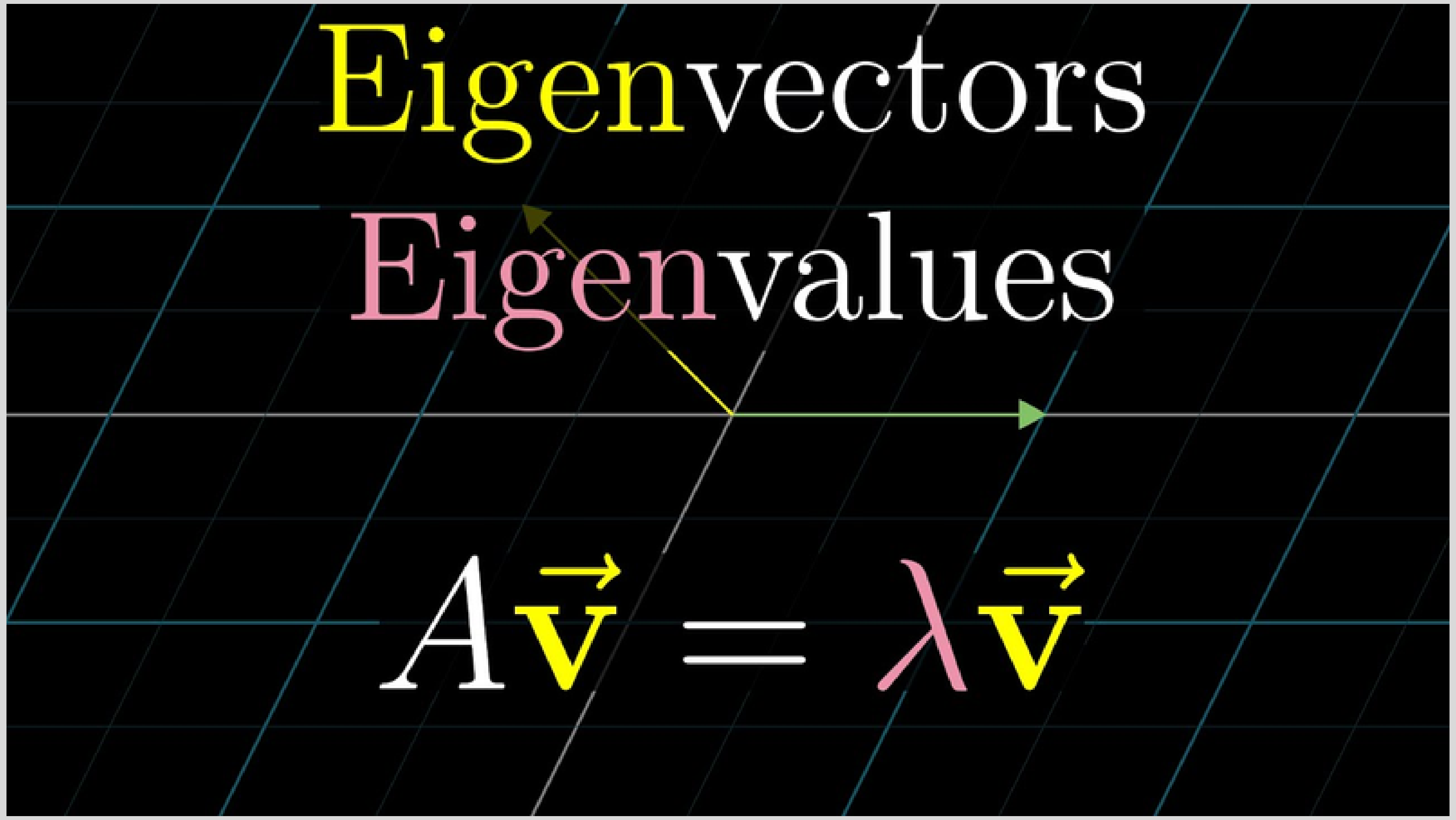
- symmetric positive definite matrix or cholesky factorisation (U is L transpose)



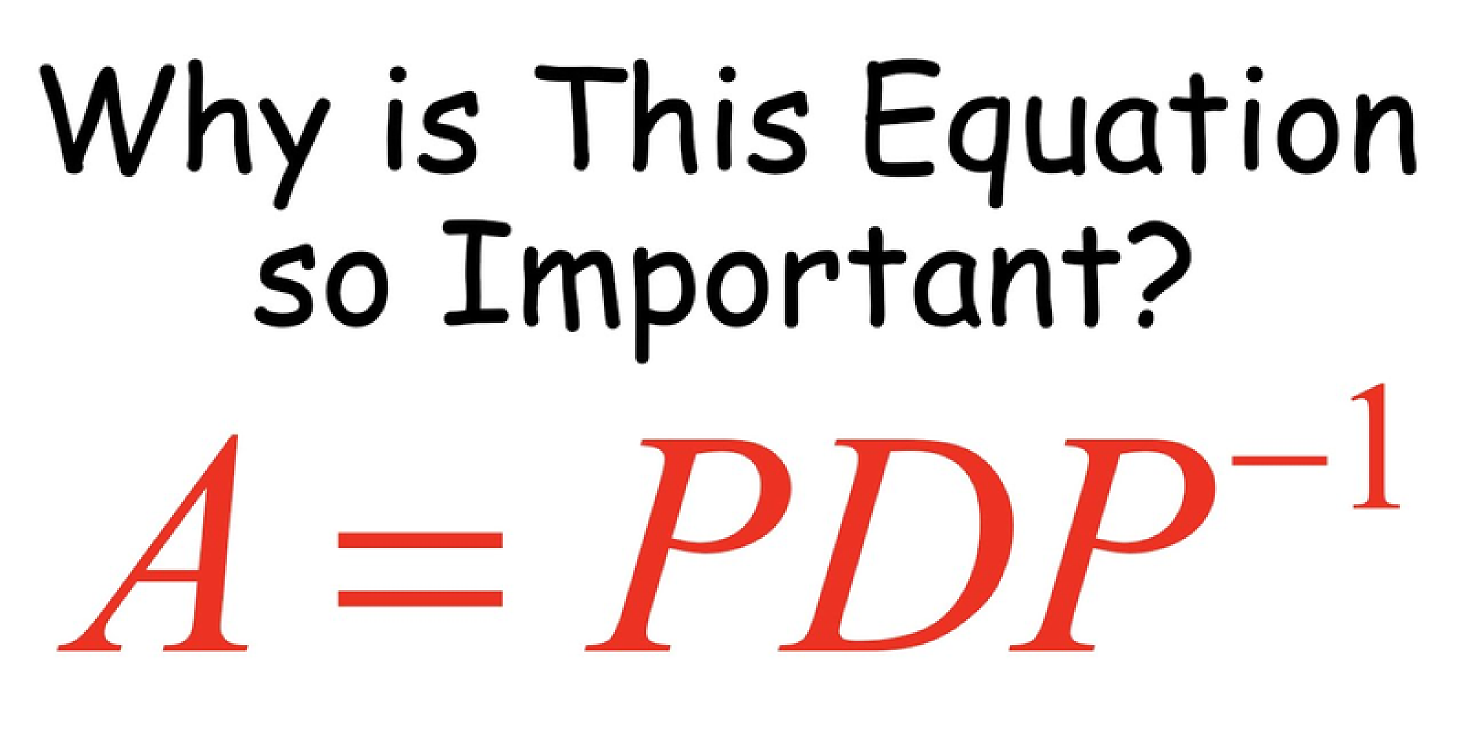
- forward substitutes and backward substitutes in solving AX=B

1. A=LU
2. LY=B
3. UX=Y

Eigenvalues, Eigenvectors and Diagonalization



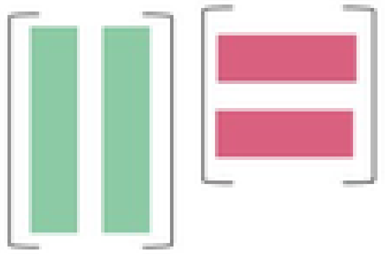
- eigenvectors,eigenvalues,eigenspace
- calculation of eigenvalues using CAYLEY-HAMILITON theorem
- Geometric application of eigenvectors
- Geometric and algebraic multiplicity of eigenvalues
- Fundamental theorem of Invertible matrices.



- The Diagonalization Theorem
- Need for the diagonalization of matrix
- Eigen decomposition
- Application of eigenvalues, eigenvectors and eigen decomposition and diagonalisation in Machine Learning

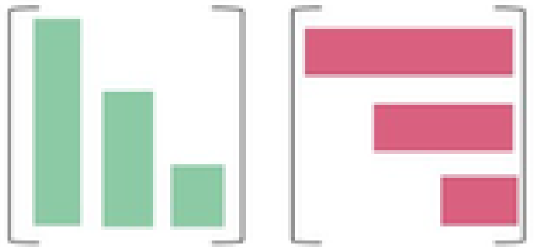
Matrix Decompositions

$$A = CR$$



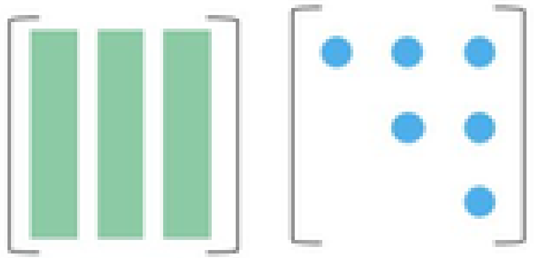
Independent column vectors times row echelon form to show row rank = column rank

$$A = LU$$



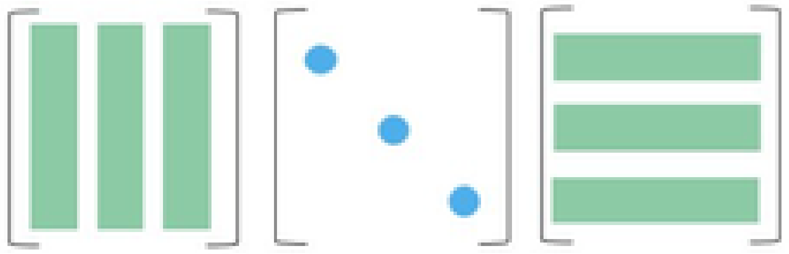
LU decomposition as Gaussian elimination

$$A = QR$$



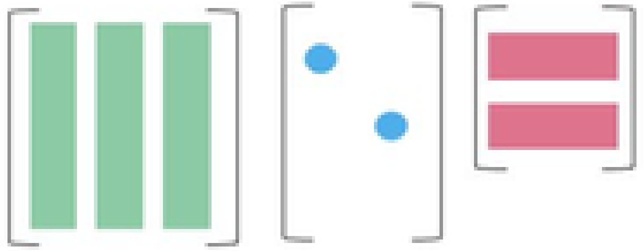
QR decomposition as Gram-Schmidt orthogonalization

$$S = Q\Lambda Q^{\mathrm{T}}$$



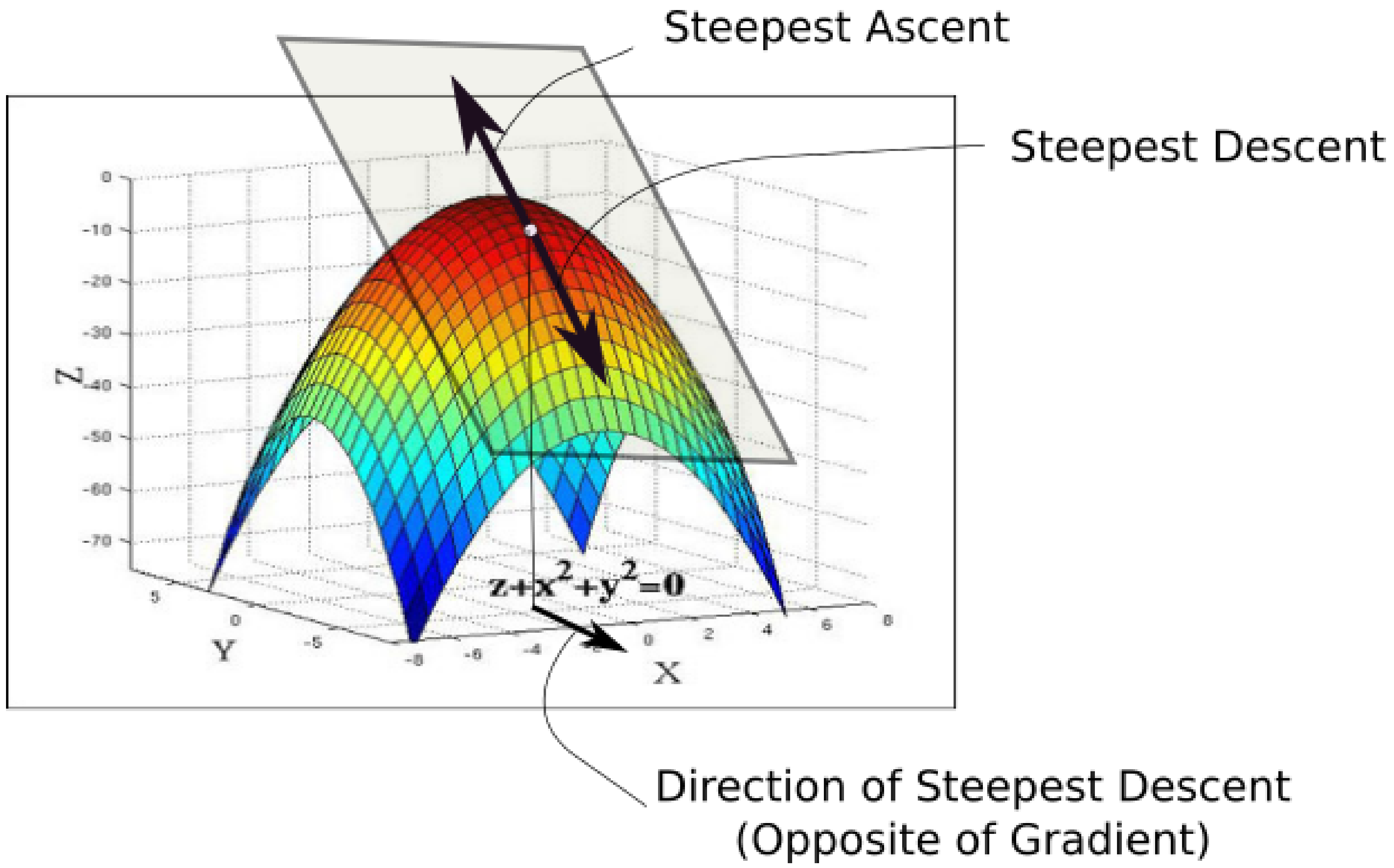
Eigenvalue decomposition of a symmetric matrix S

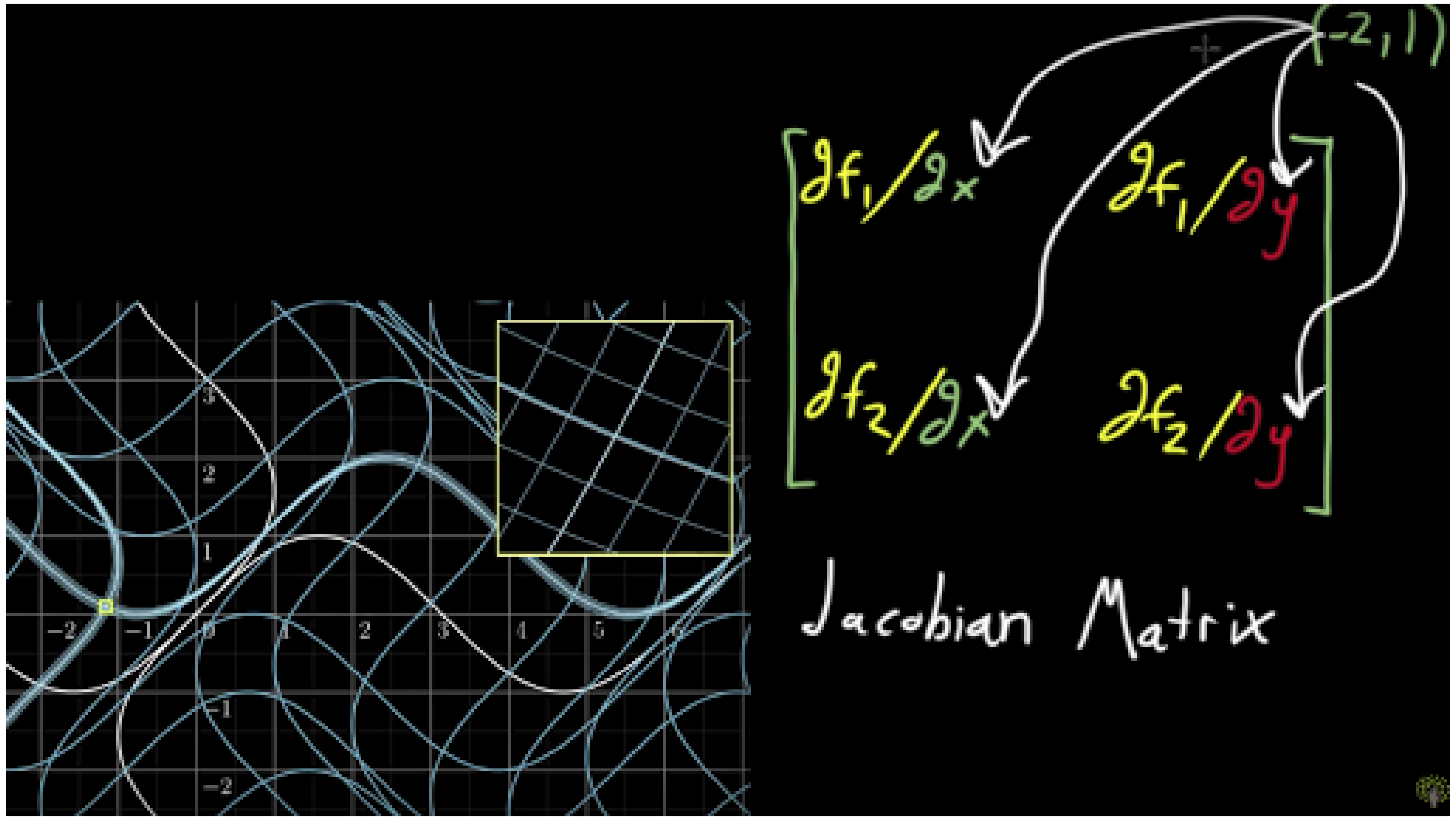
$$A = U\Sigma V^{\mathrm{T}}$$



Singular value decomposition of all matrices A

Vector Analysis





OPTIMIZATION THEORY

Optimization theory is a branch of mathematics that deals with finding the best possible solution from a set of alternatives, typically involving maximizing or minimizing an objective function while satisfying a set of constraints.

Covariance Matrix

data value of X

mean value of X

data value of Y

$$Cov(x,y) = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{n}$$

mean value of Y

Number of data values

$$\mathrm{Cov}[X,Y] = \begin{bmatrix} \mathrm{E}[(X_1 - \mathrm{E}[X_1])(Y_1 - \mathrm{E}[Y_1])] & \mathrm{E}[(X_1 - \mathrm{E}[X_1])(Y_2 - \mathrm{E}[Y_2])] \\ \mathrm{E}[(X_2 - \mathrm{E}[X_2])(Y_1 - \mathrm{E}[Y_1])] & \mathrm{E}[(X_2 - \mathrm{E}[X_2])(Y_2 - \mathrm{E}[Y_2])] \\ \mathrm{E}[(X_3 - \mathrm{E}[X_3])(Y_1 - \mathrm{E}[Y_1])] & \mathrm{E}[(X_3 - \mathrm{E}[X_3])(Y_2 - \mathrm{E}[Y_2])] \end{bmatrix}$$

$$= \begin{bmatrix} \mathrm{Cov}[X_1, Y_1] & \mathrm{Cov}[X_1, Y_2] \\ \mathrm{Cov}[X_2, Y_1] & \mathrm{Cov}[X_2, Y_2] \\ \mathrm{Cov}[X_3, Y_1] & \mathrm{Cov}[X_3, Y_2] \end{bmatrix}$$

PAC Model

PAC Model (Probably Approximately Correct):
- The PAC model is a theoretical framework for analyzing the generalization ability of machine learning algorithms.
- It defines the conditions under which an algorithm can learn from a finite set of training examples and make accurate predictions on unseen data.
- The model encompasses three key concepts: sample complexity, computational complexity, and error bounds.
- By understanding the PAC model, we can assess the trade-offs between the size of the training set, the complexity of the algorithm, and the achievable level of accuracy.

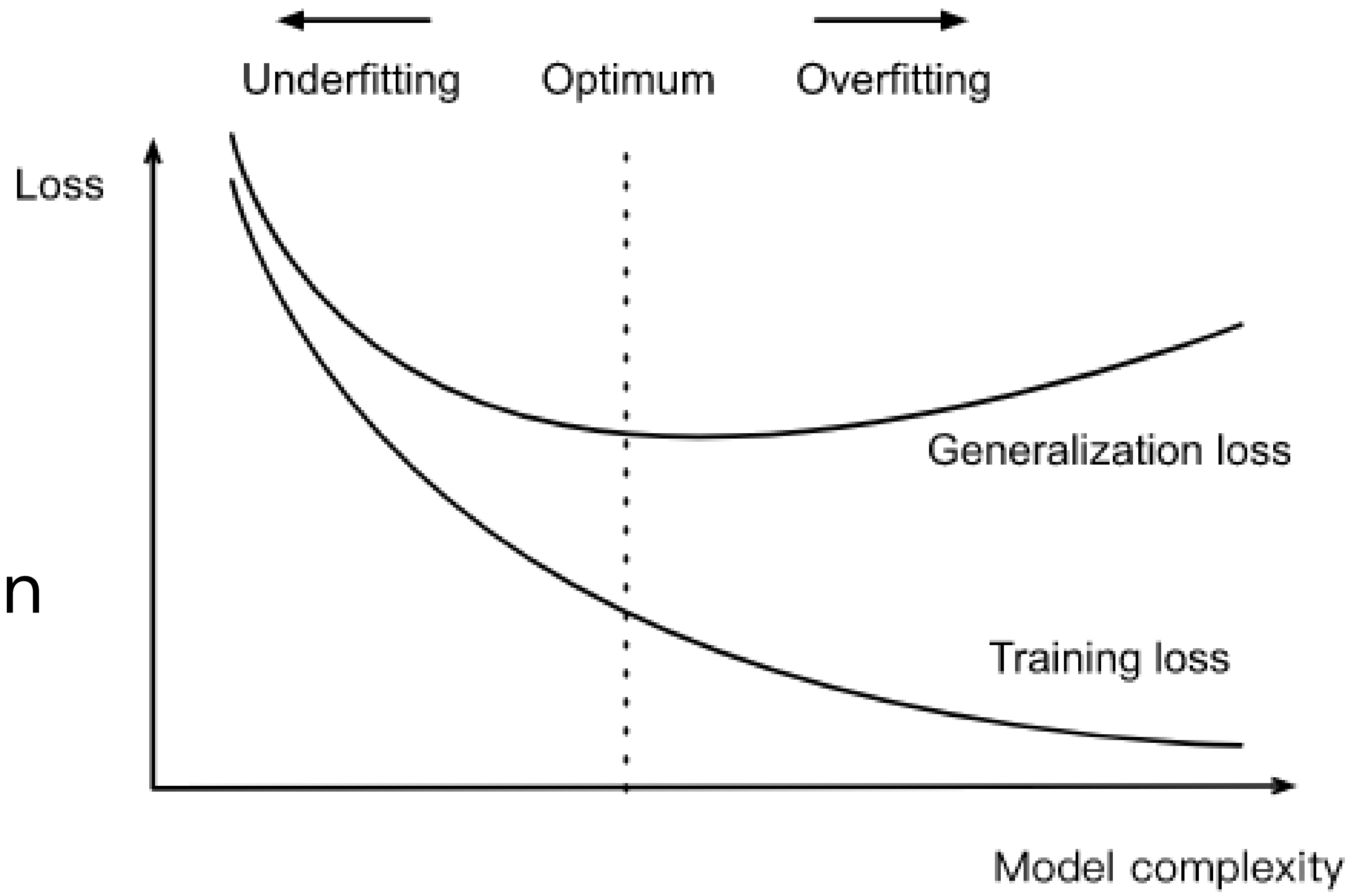
ERM vs SRM

ERM (Empirical Risk Minimization) vs. SRM (Structural Risk Minimization):

- ERM aims to minimize the empirical risk, which measures the average loss on the training data.
- SRM considers both empirical risk and model complexity to avoid overfitting.
- SRM seeks a trade-off between fitting the training data and generalizing to unseen data by incorporating regularization techniques such as weight decay or dropout.
- SRM helps prevent overfitting and improves the model's ability to generalize to new examples.

$$R_{\mathrm{srm}}(f) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i)) + \lambda J(f) \qquad R_{\mathrm{emp}}(h) = \frac{1}{m}\sum_{i=1}^{m} L(h(x_i), y_i).$$

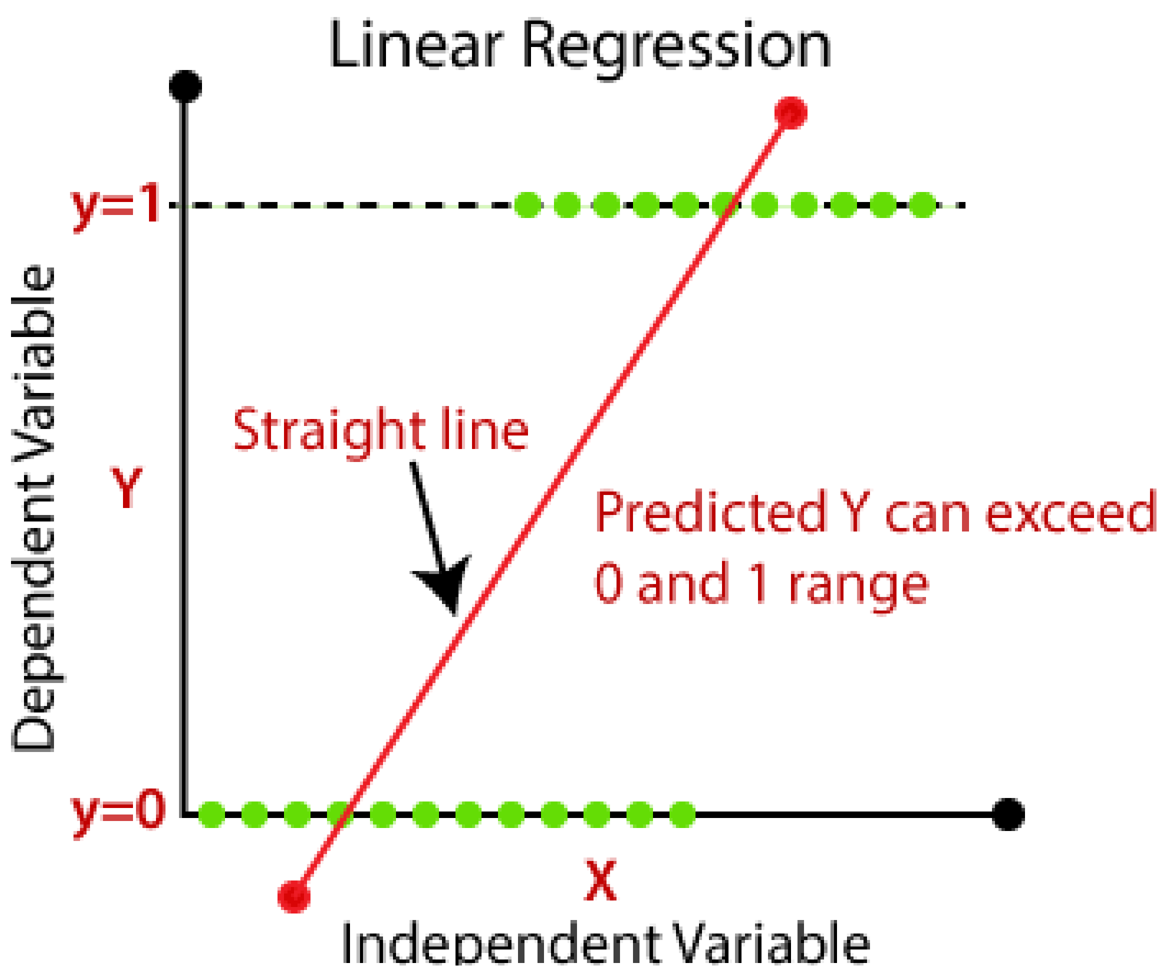
Generalization vs Prediction Loss

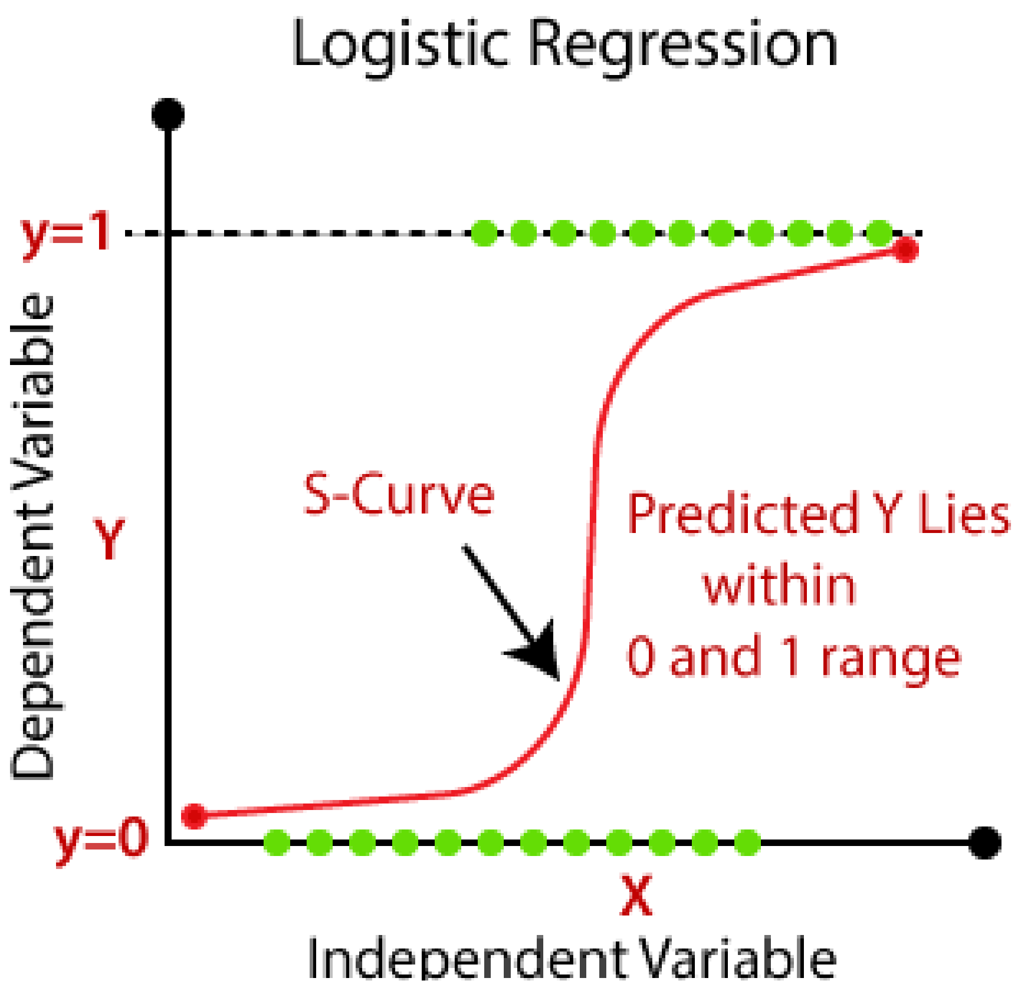


Generalization vs. Prediction Loss:
- Generalization refers to how well a model performs on unseen data.
- Prediction loss measures the discrepancy between predicted and actual values during training.
- Overfitting occurs when a model memorizes training data, leading to poor generalization.
- Regularization techniques and validation sets help prevent overfitting and improve generalization performance.

Regression Analysis

- Regression analysis is a statistical method used to model the relationship between a dependent variable and one or more independent variables.
- It aims to find the best-fitting line (or curve) that represents the relationship between variables.
- It helps us understand how changes in independent variables impact the dependent variable and make predictions based on the model.
- Common regression techniques include linear regression, polynomial regression, and multiple regression.



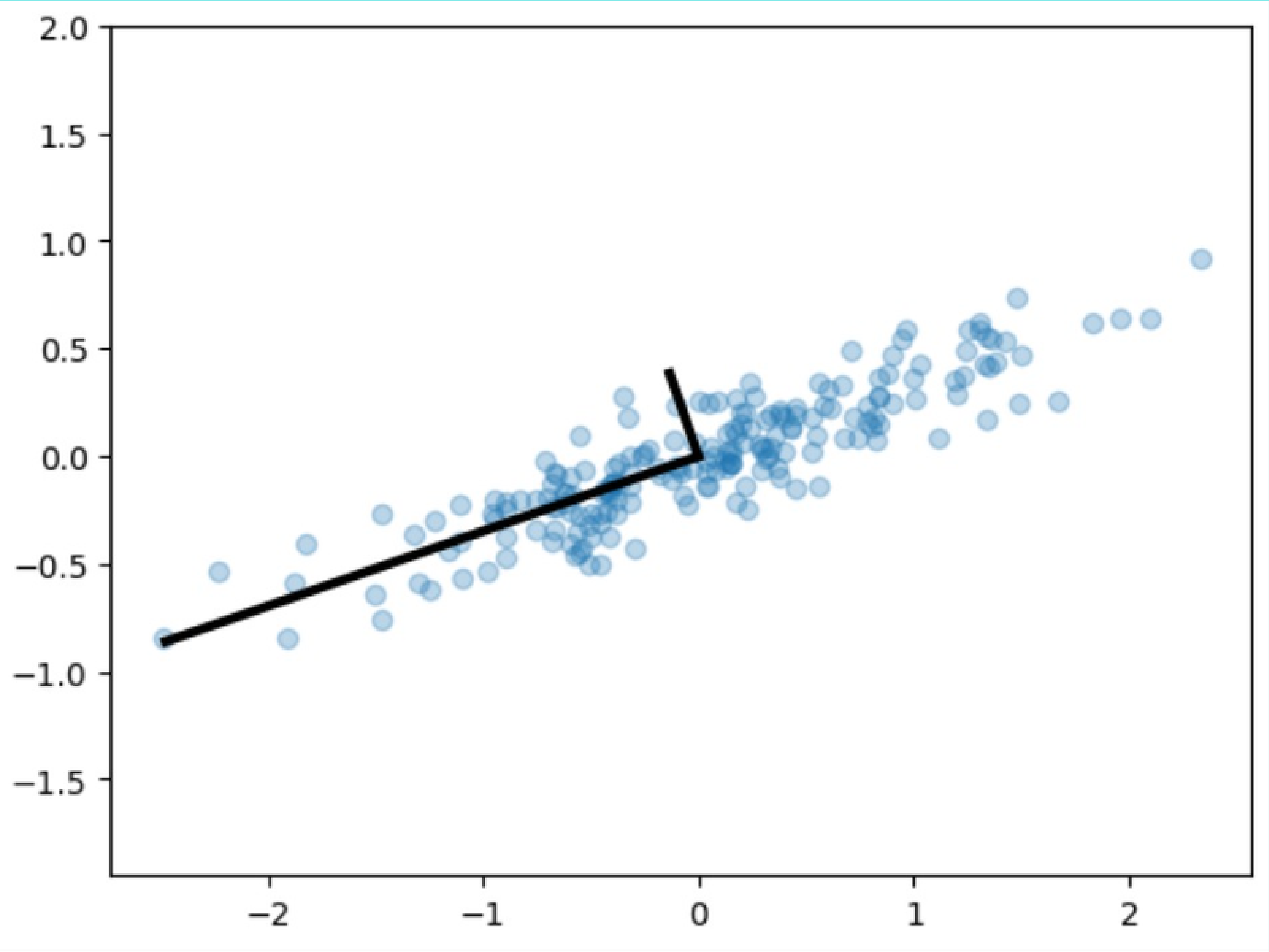


Principal Components Analysis

- Main purpose is Dimensionalty reduction of dataset while preserving most of the information
- Eigen-concepts play a key role
- Applications of LA in PCA

1. Covariance matrix
2. eigenvalues and eigenvectors

- PCA built-in Library in sklearn.decomposition

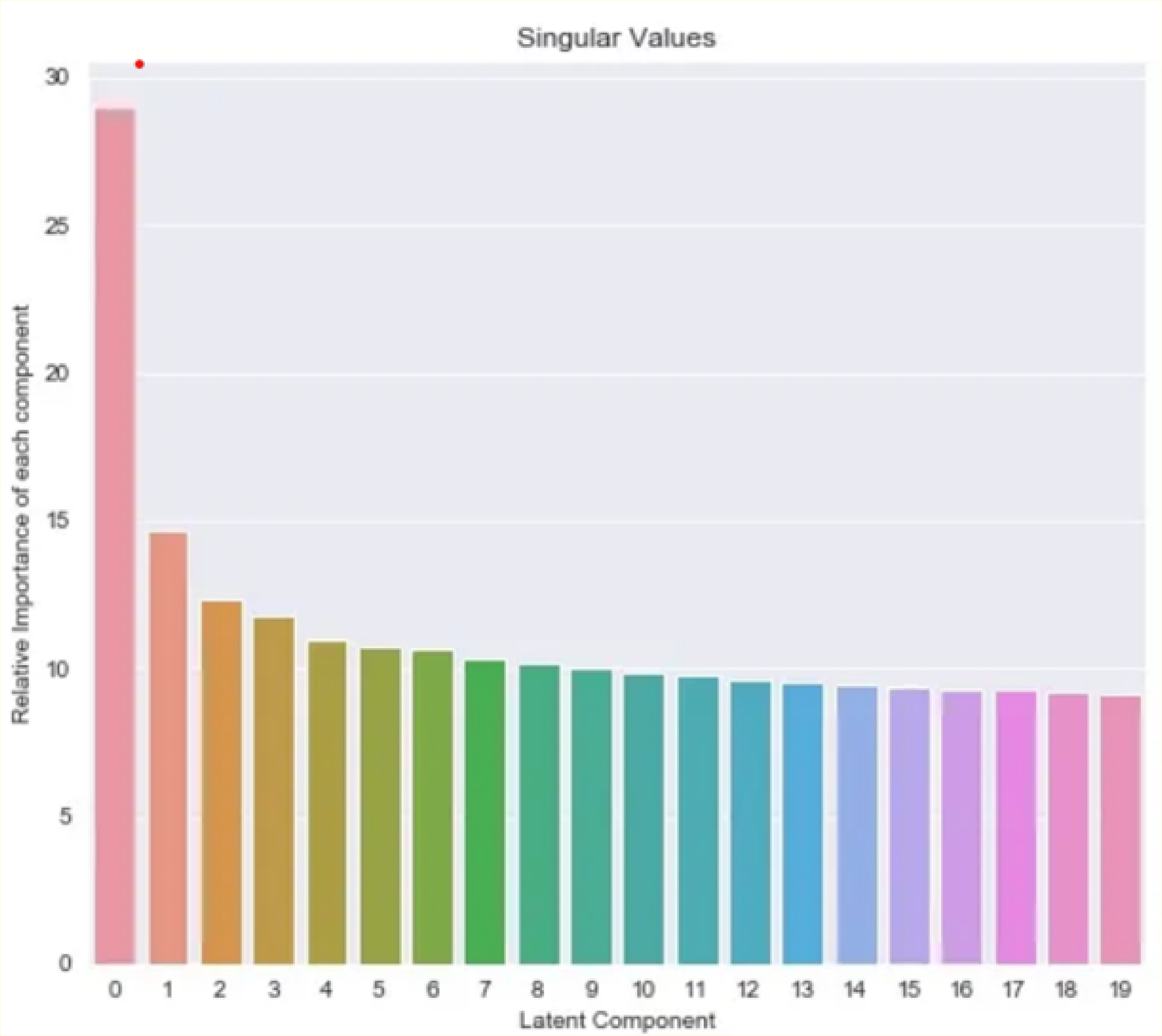


- Applications of PCA in ML:

1. Data visualization
2. Dimensionality reduction
3. Noise compression
4. Feature Extraction
5. Data Preprocessing

Latent Semantic Analysis

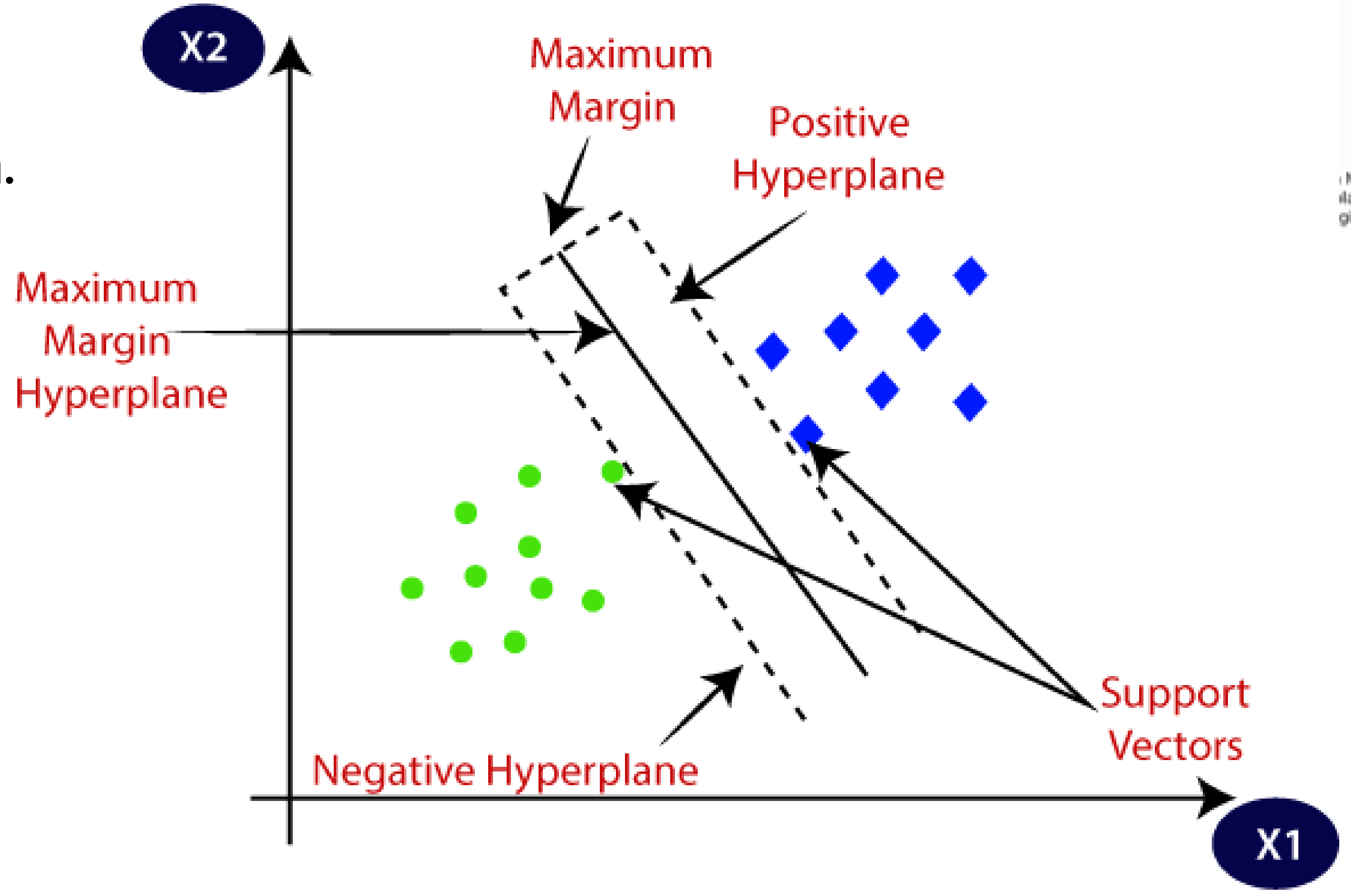
- main purpose is to analyse the relationships between a set off documents and terms
- Application of LA in LSA
1. Singular value decomposition (SVD)
2. Term-Document Matrix formation
- semantic representation and analysis
- LSA using scikit-Learn



- Applications of LSA in ML:
1. Information Retrieval
2. Document clustering
3. Question Answering system

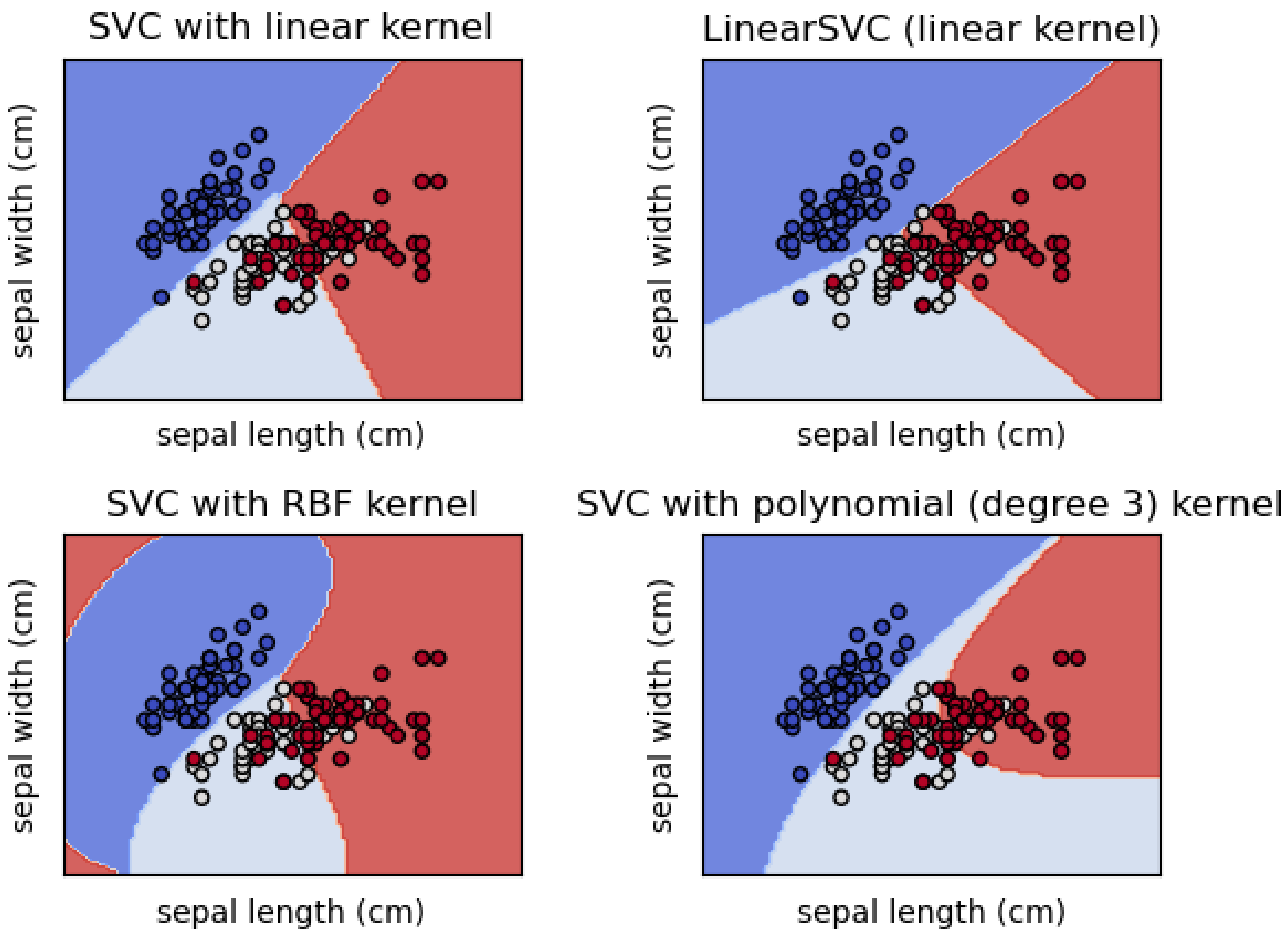
Support Vector Machines

- Support Vector Machines are powerful supervised learning models used for classification and regression.
- They find the optimal hyperplane that maximally separates data points of different classes in feature space.
- SVMs are effective in handling high-dimensional data and can also handle non-linear decision boundaries using kernel functions.
- They aim to maximize the margin between the classes, making them robust against overfitting and suitable for small to medium-sized datasets.



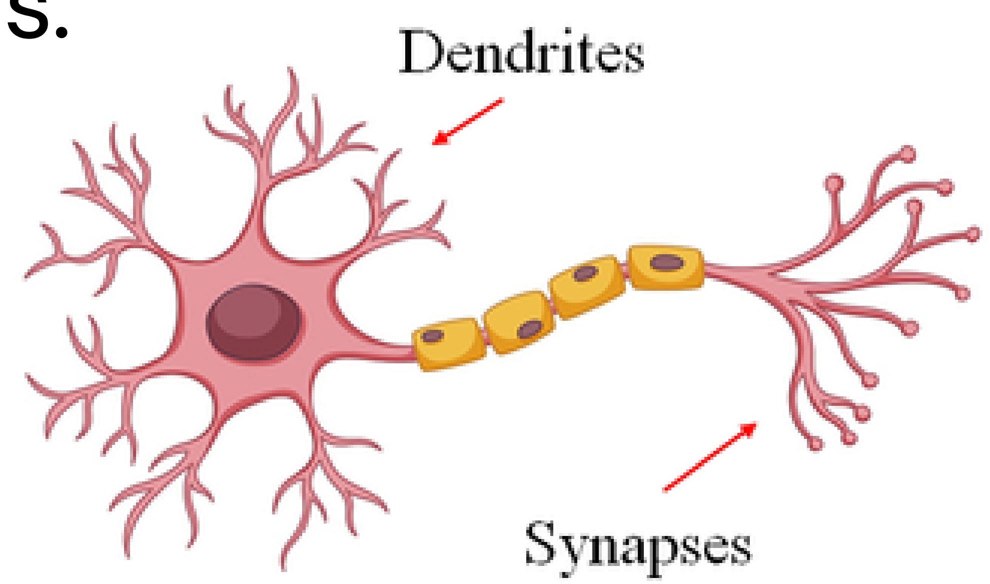
Support Vector Machines- Kernels

Support Vector Machine (SVM) Kernels:
- SVM kernels are mathematical functions used to transform input data into higher-dimensional feature spaces.
- They enable SVMs to handle nonlinearly separable data by mapping it to a space where linear separation becomes possible.
- Common kernel functions include linear, polynomial, Gaussian (RBF), and sigmoid.
- Kernels play a crucial role in SVM's ability to capture complex relationships and achieve higher classification accuracy.

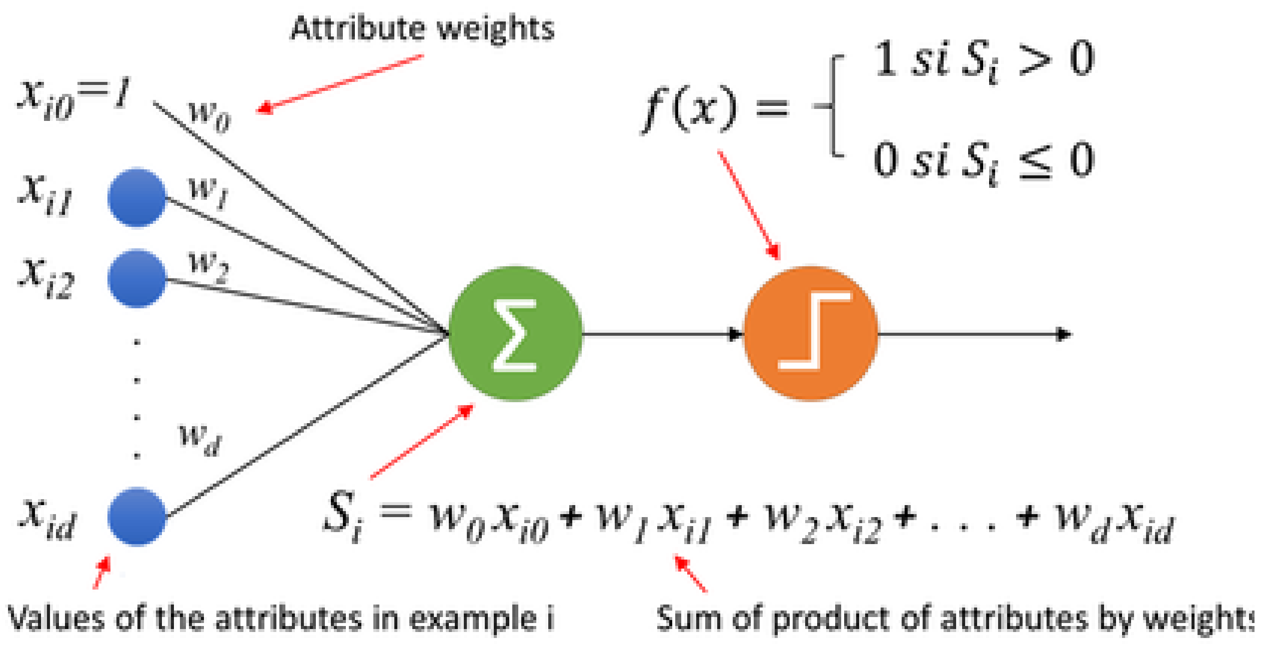


Neural Networks- Introduction

- ML models inspired by the human brain.
- Interconnected layers of artificial neurons (nodes) that process and transmit information.
- By adjusting the weights and biases of these connections, neural networks can learn from data and make predictions or classifications.
- Applications- image recognition, natural language processing, and recommendation systems.



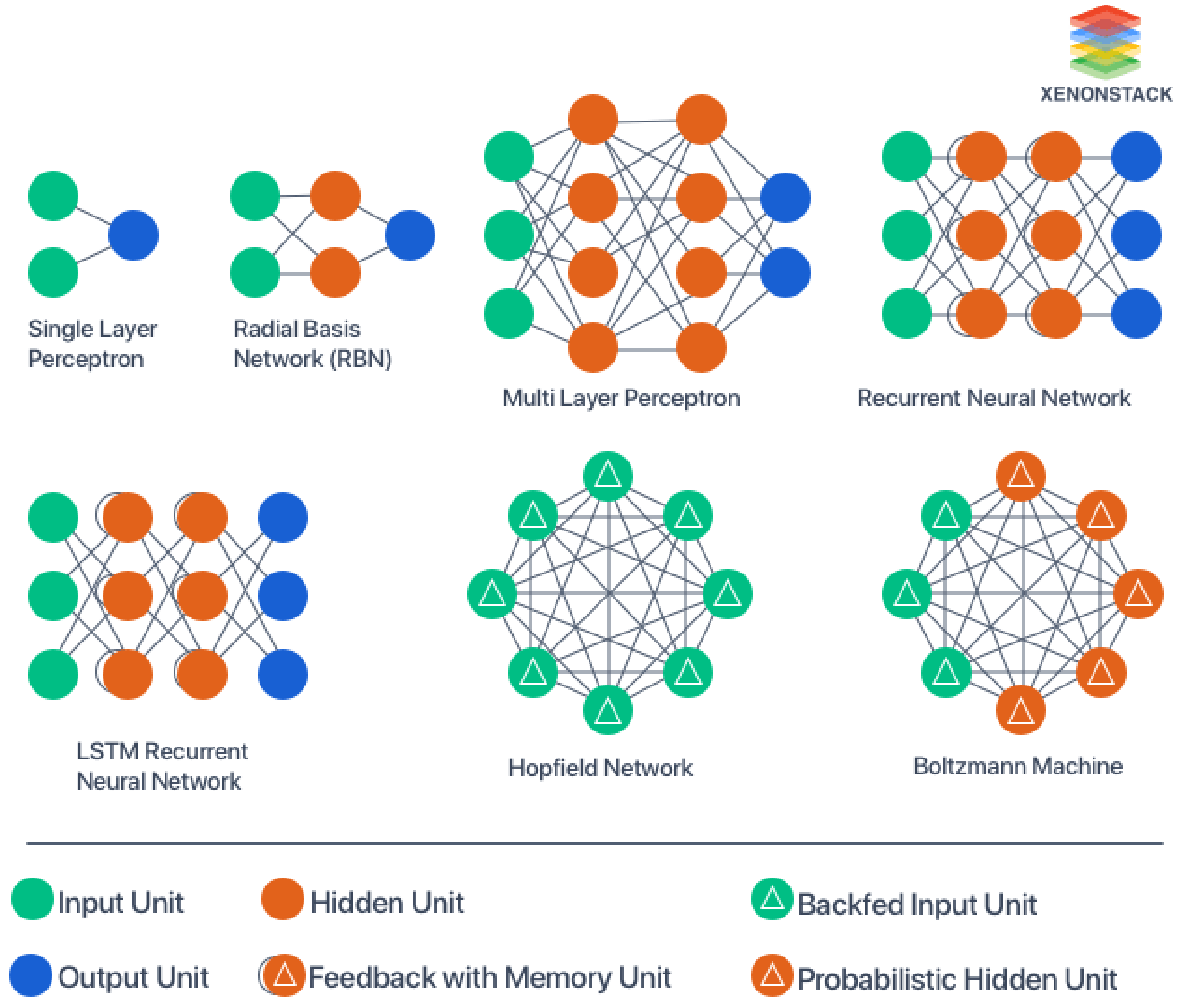
NEURON



PERCEPTRON

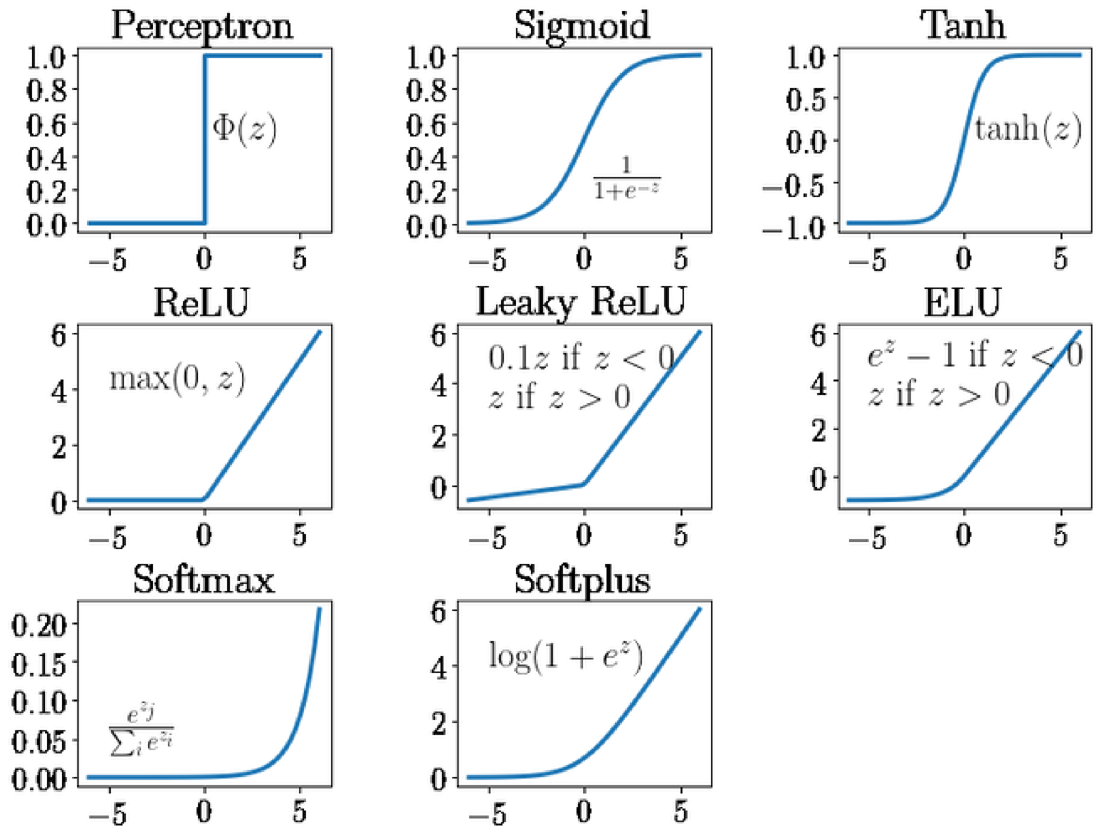
Neural Networks–Working

- NNs are composed of interconnected layers of artificial neurons called nodes or units.
- Each layer performs mathematical operations on the input data, transforming it through non-linear activation functions.
- The architecture consists of an input layer, one or more hidden layers, and an output layer.
- Hidden layers enable the network to learn complex representations, while the output layer produces the final predictions or classifications.

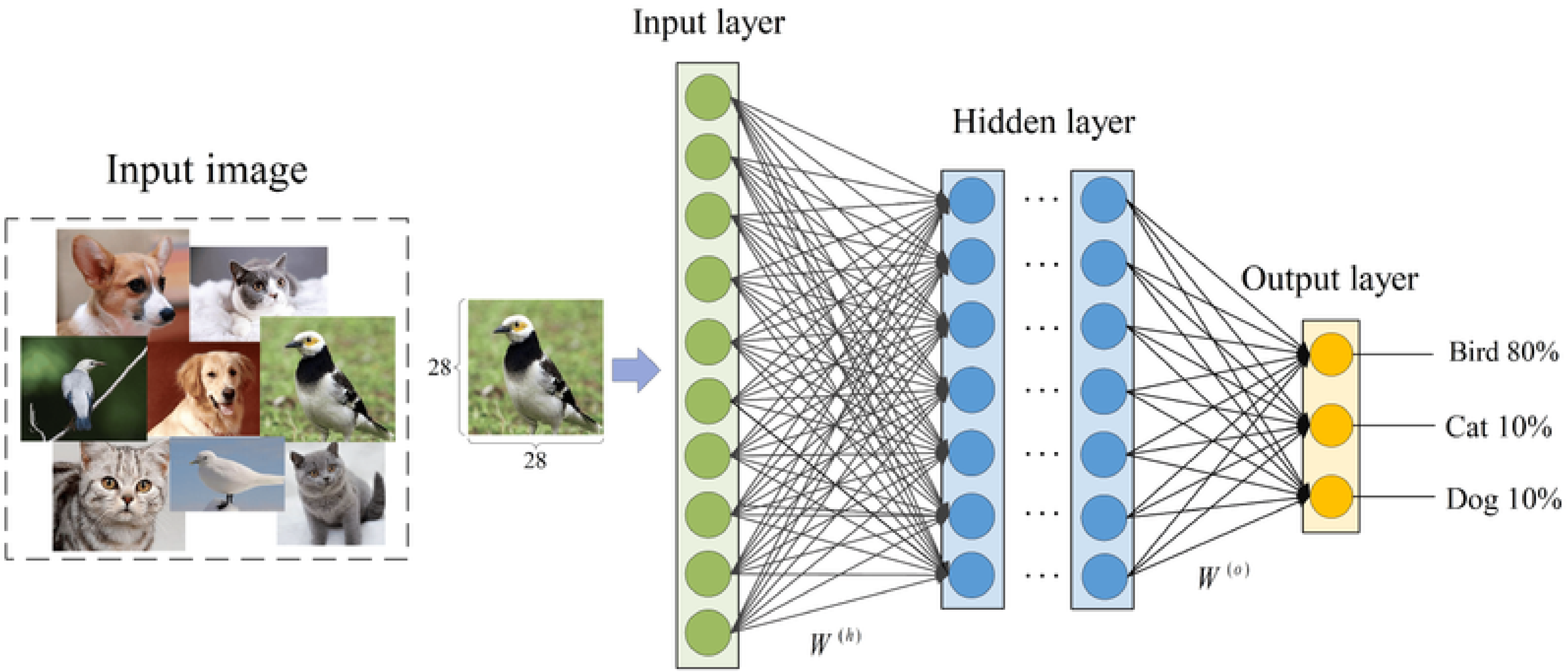


Neural Networks-Working

- Activation functions introduce non-linearity to neural networks, enabling them to model complex relationships.
- Common activation functions include sigmoid, tanh, and ReLU.
- Sigmoid and tanh functions squash input values to a specific range, while ReLU activates only for positive inputs.
- The choice of activation function impacts the network's ability to learn and handle different types of data.



Neural Networks – Example



Team 66

Thank you!

Have a great day ahead.