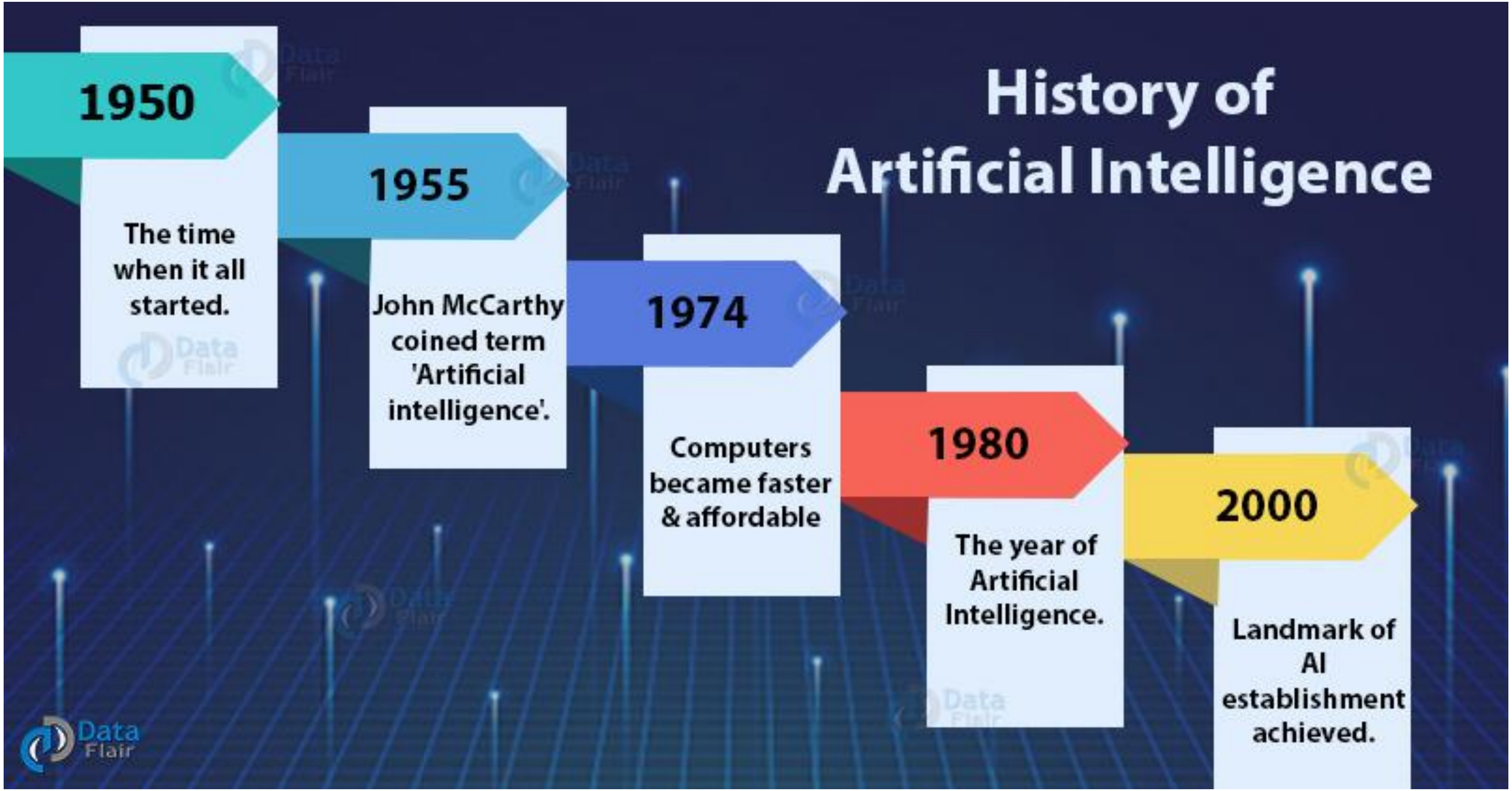
AI61003: LINEAR ALGEBRA FOR AI & ML: Motivation

Mahesh Mohan M R, Prabhat Kumar Mishra
Centre of Excellence in AI
Indian Institute of Technology Kharagpur
24-07-2024

Brief History of Artificial Intelligence



Brief History of Linear Algebra

4000 Years ago

4000 years ago the Babylonians knew how to solve a system of two linear equations in two unknowns

1693-1811

Modern study of systems of linear equations: Leibinitz (1693); Cramer (1750), Gaussian Elimination (1811)

1850-1858

Cayley formally introduced M × N matrices in 1850 and 1858.

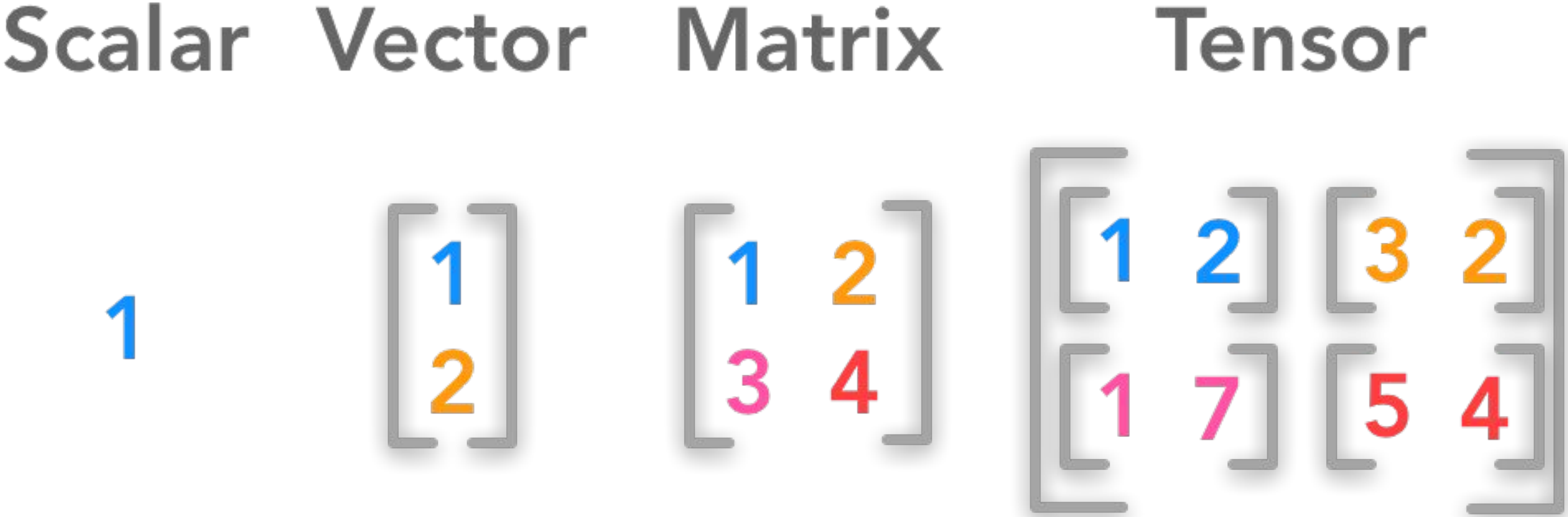
Till 1870s

Spectral theory of matrices, Eigen values and Eigen vectors: Cauchy, Jacobi, Jordan, Weierstrass, and other

1888

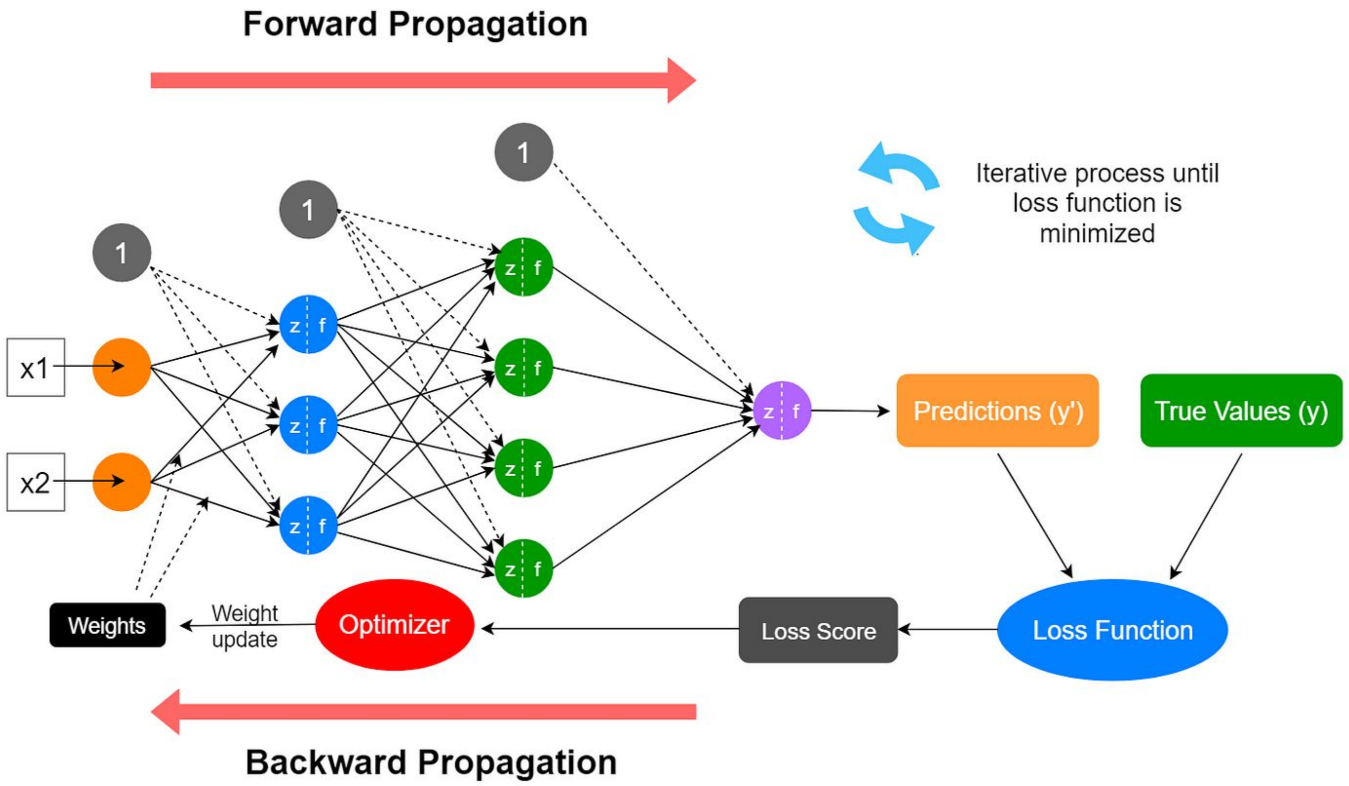
Vector space introduced by Peano in 1888

Units of Linear Algebra in AI

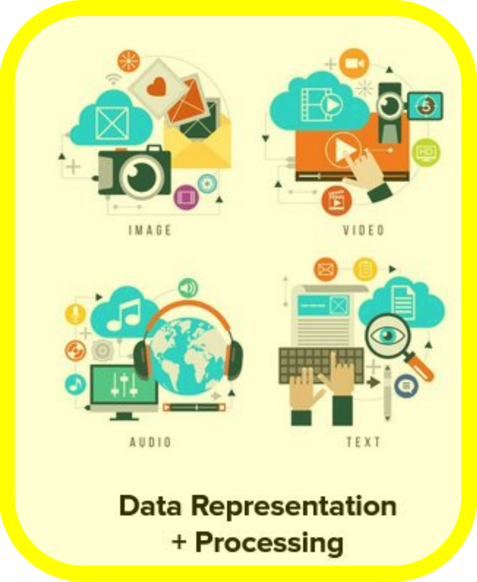


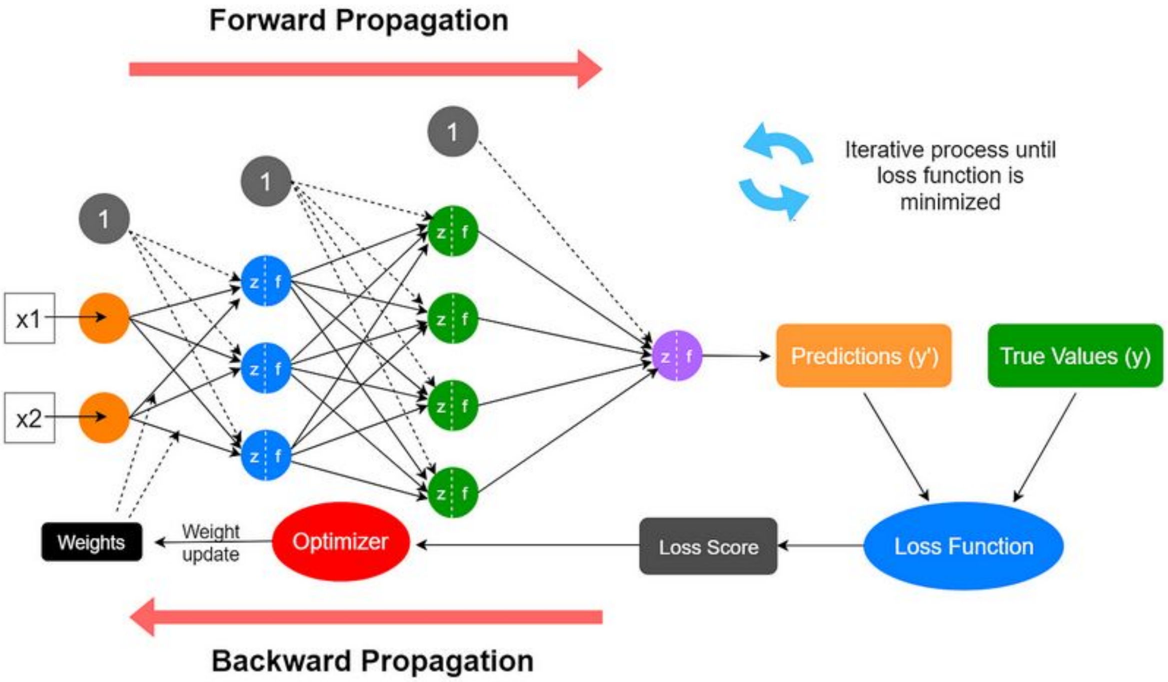
Stages of Neural Network Training

Data Representation + Processing

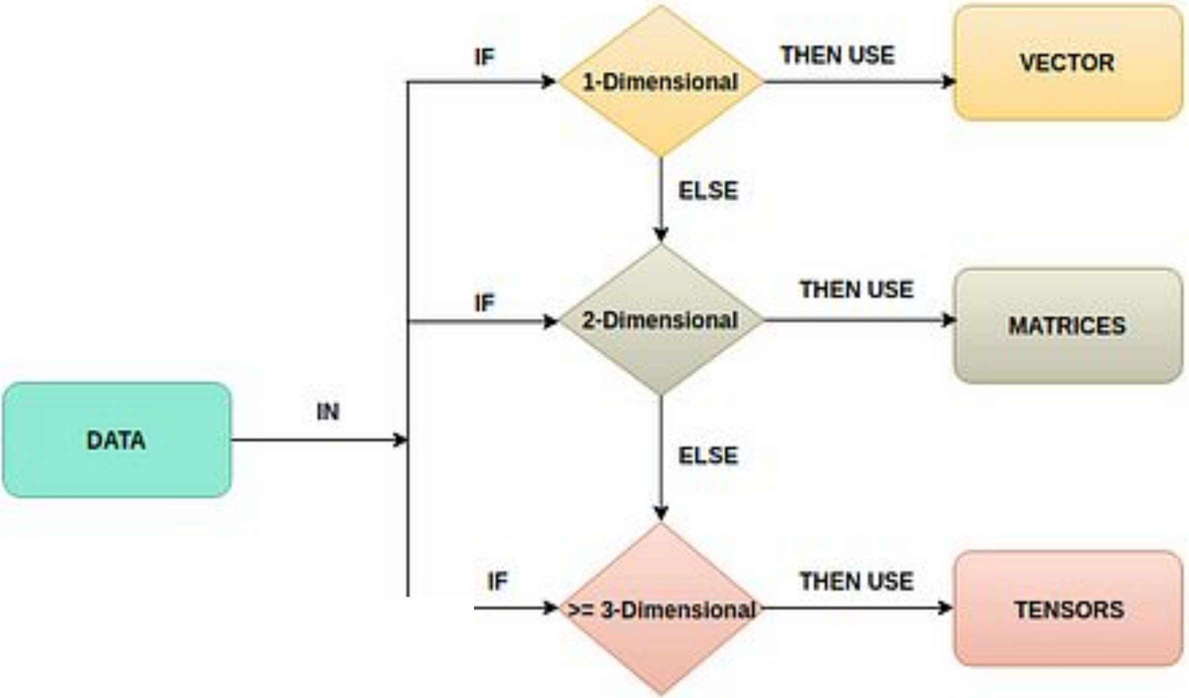


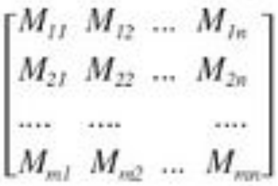
Linear Algebra for Data Representation and Pre-processing

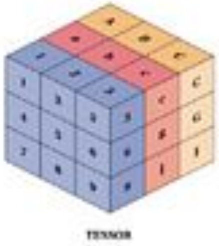


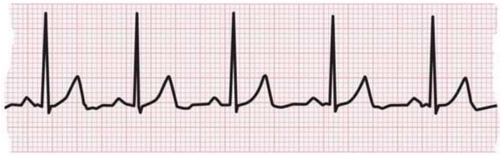


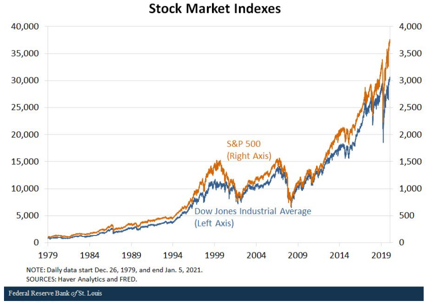
Data Representation

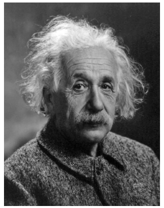


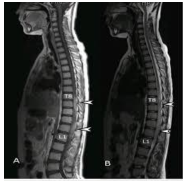


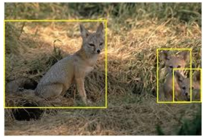


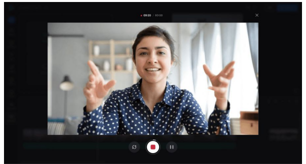




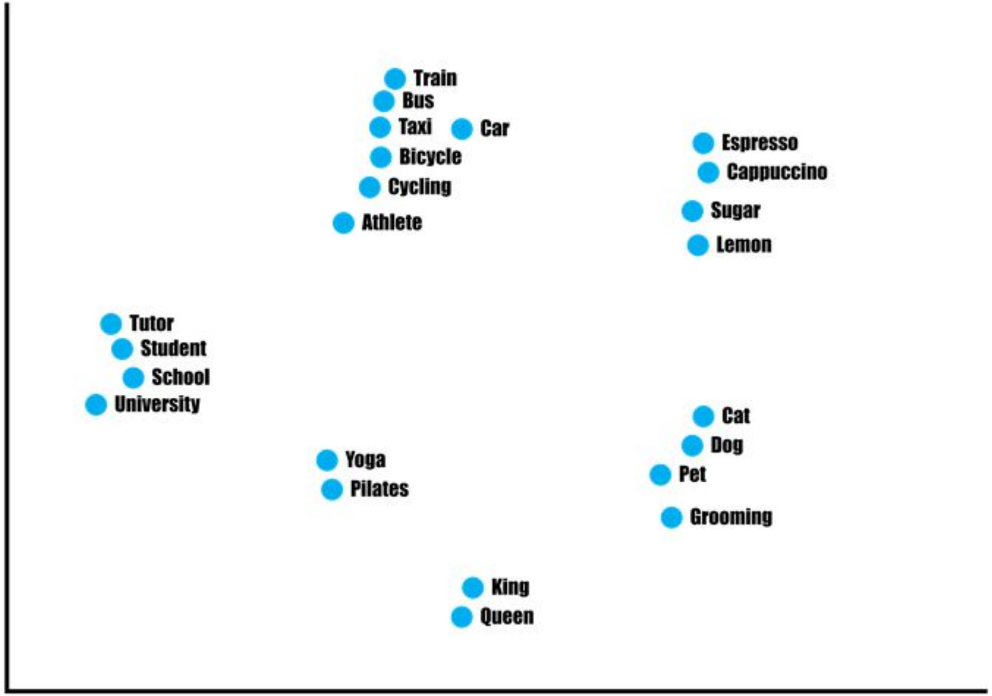








Word Embedding as Vectors

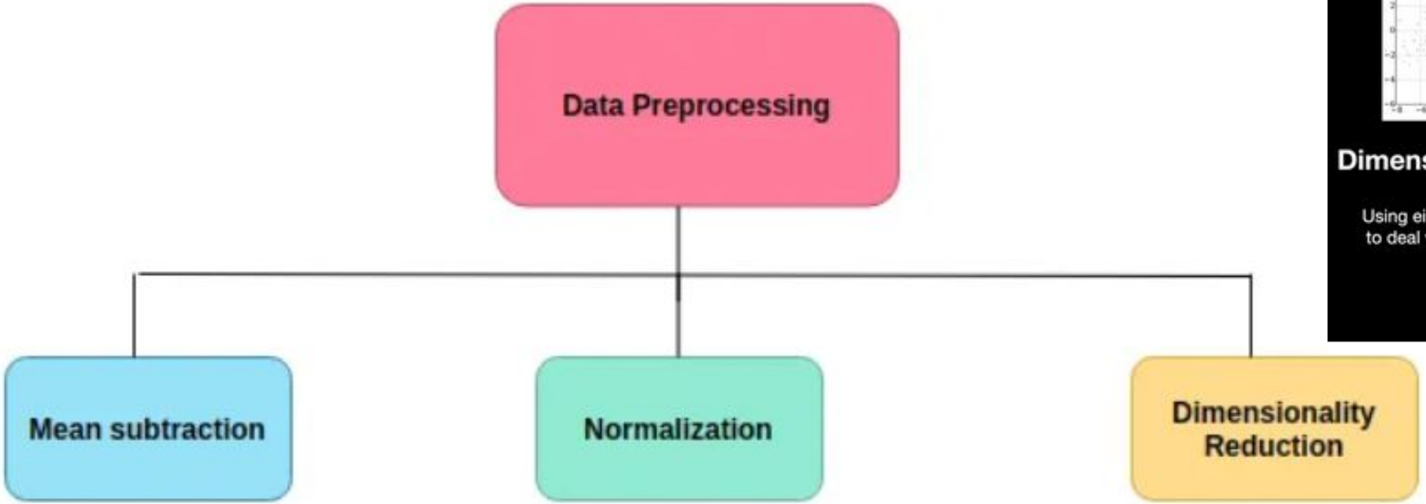


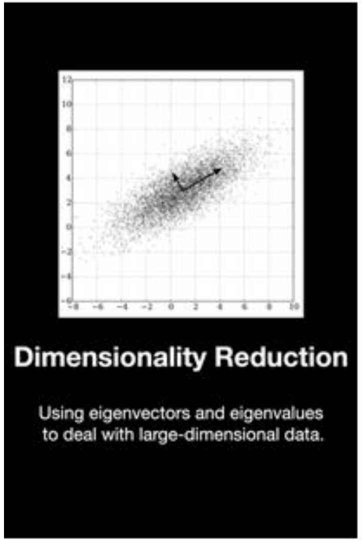
[Experiment]

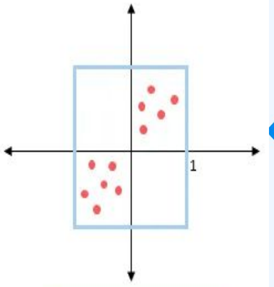
```
"Hello, do you like
tea? <|endoftext|>
In the sunlit
terraces of
someunknownPlace."
```

```
[15496, 11, 466, 345,
588, 8887, 30, 220,
50256, 554, 262,
4252, 18250, 8812,
2114, 1659, 617,
34680, 27271, 13]
```

Data Pre-processing





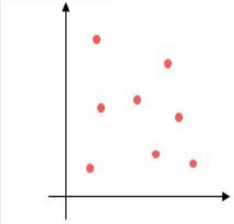


Structured data

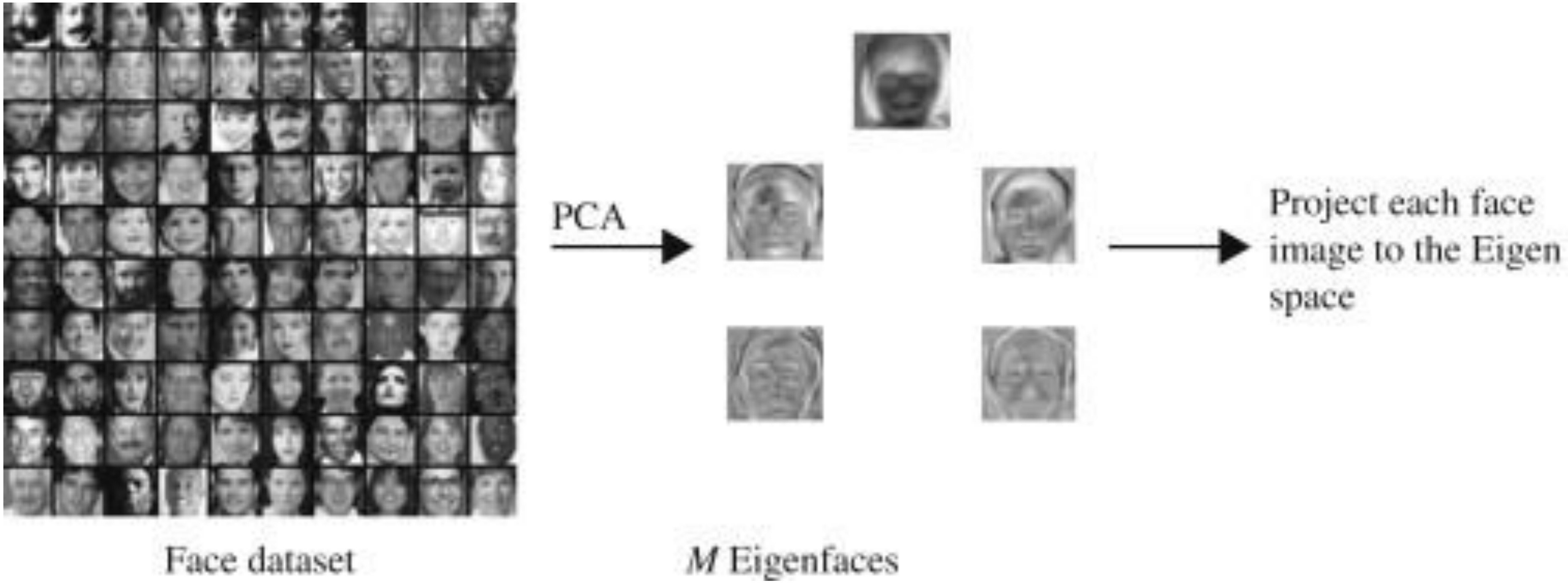
Data that is collected from a known method and can be neatly arranged

Unstructured data

Data that has no pre-defined format or organization, making analysis more difficult

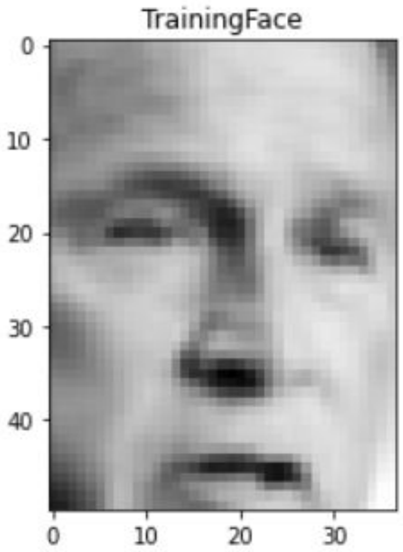


Dimensionality Reduction



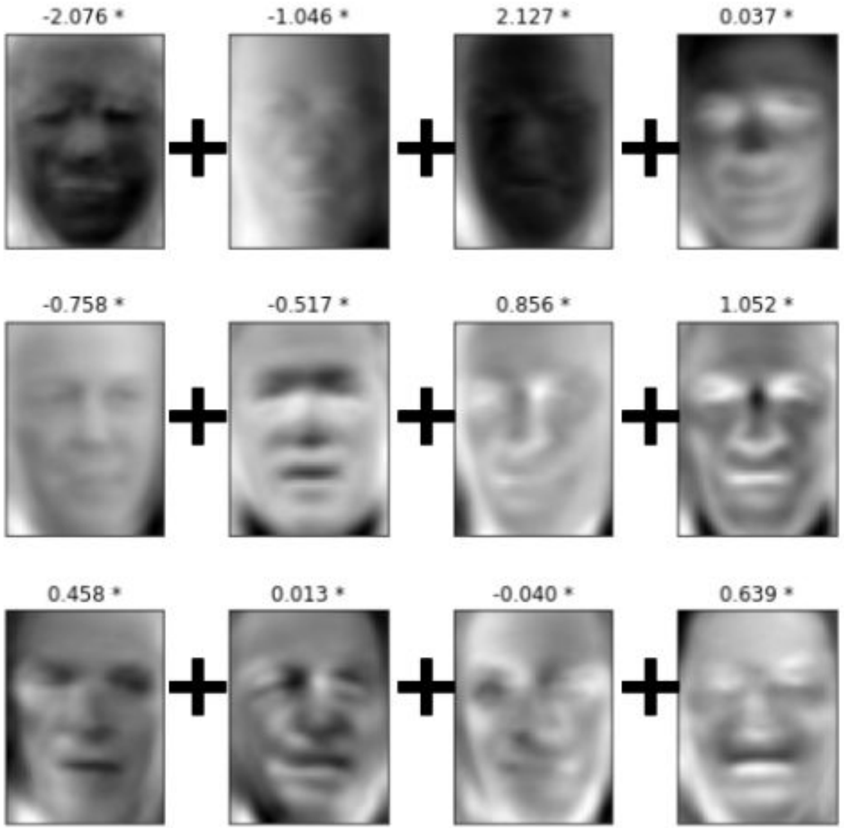
Dimensionality Reduction

50*40 = 2000 dimensions



=

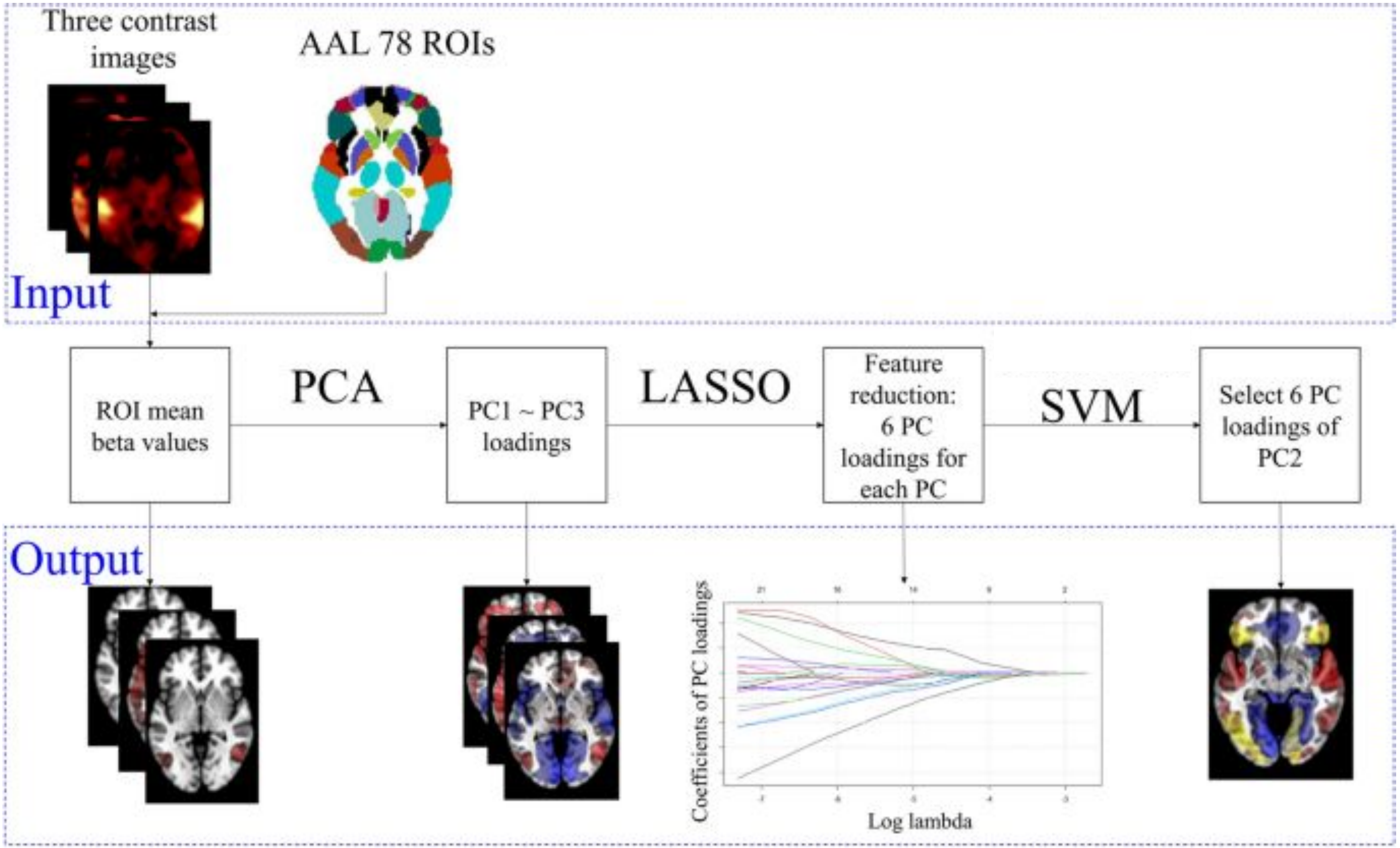
12 dimensions



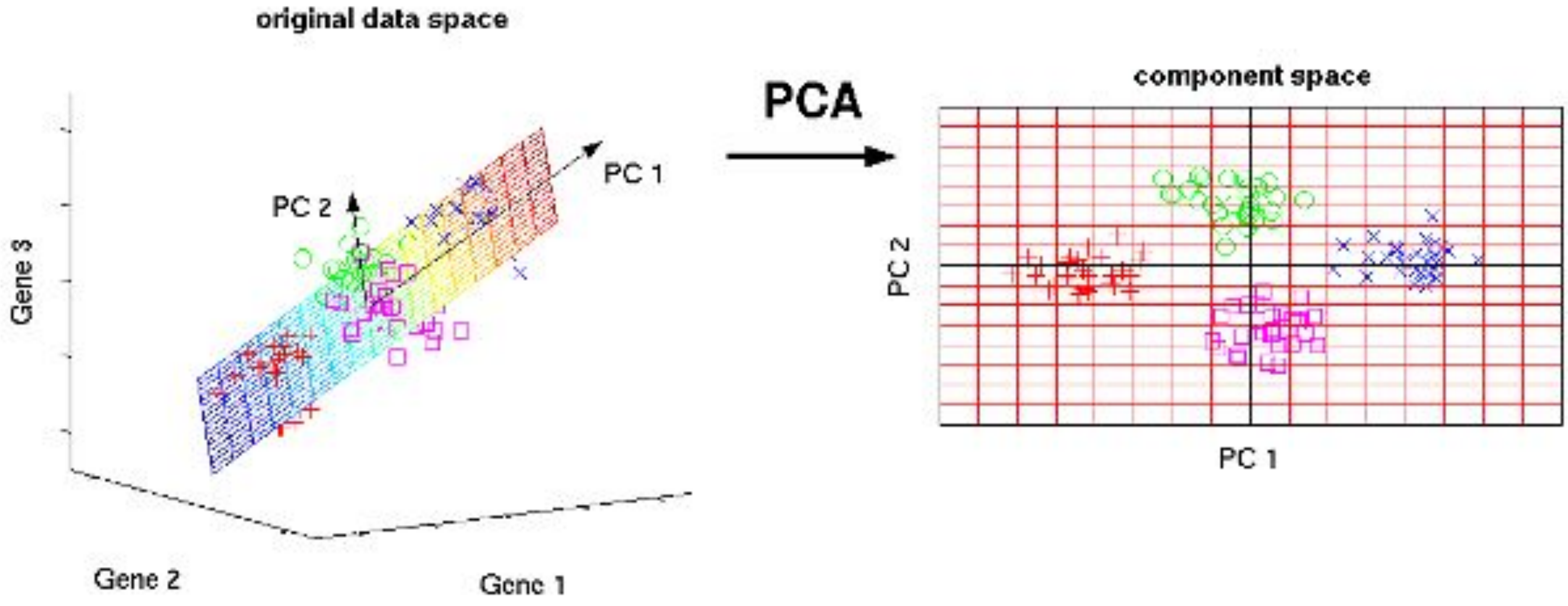
Dimensionality Reduction: Application 1

	George	Jeff	John	Tom
distanc-e	252.996	253.641	47.8731	158.434

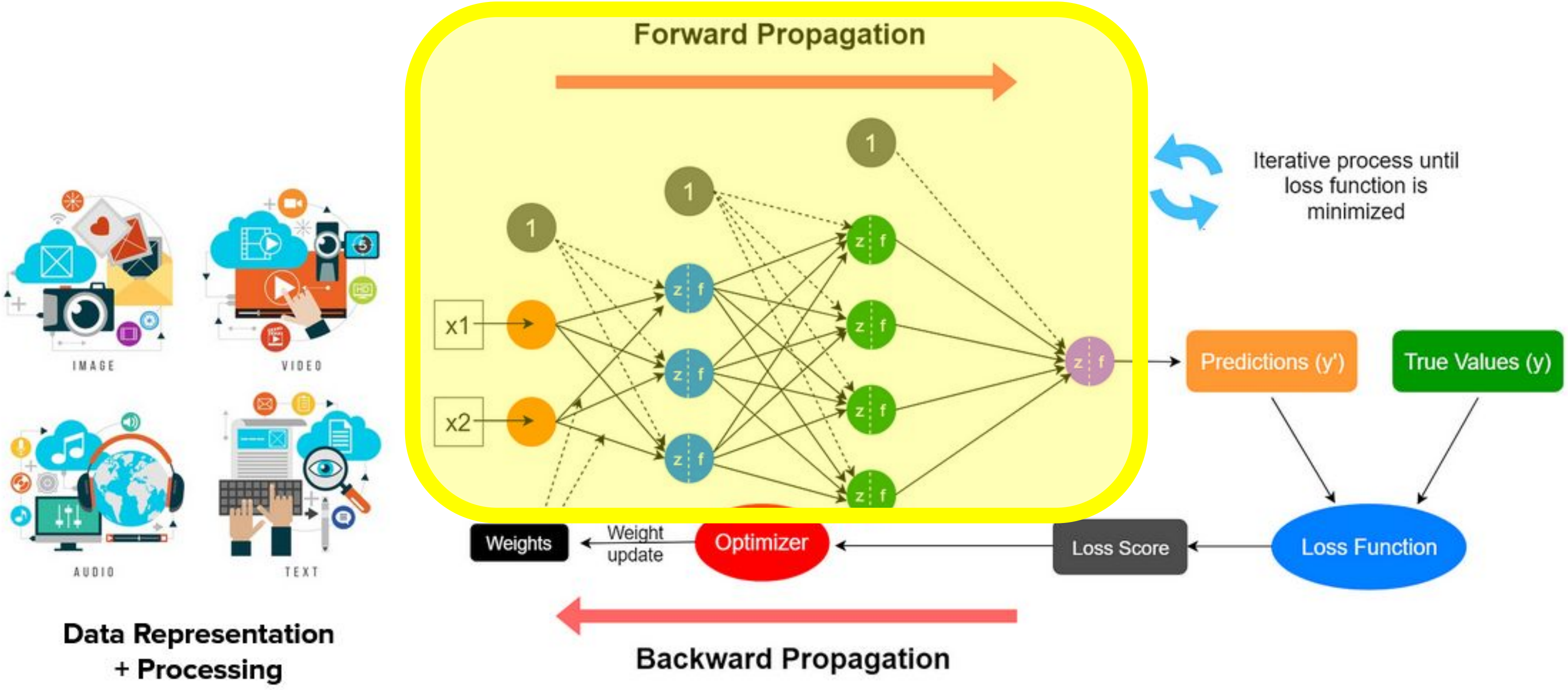
Dimensionality Reduction: AI Application



Dimensionality Reduction: Exploratory Data Analysis

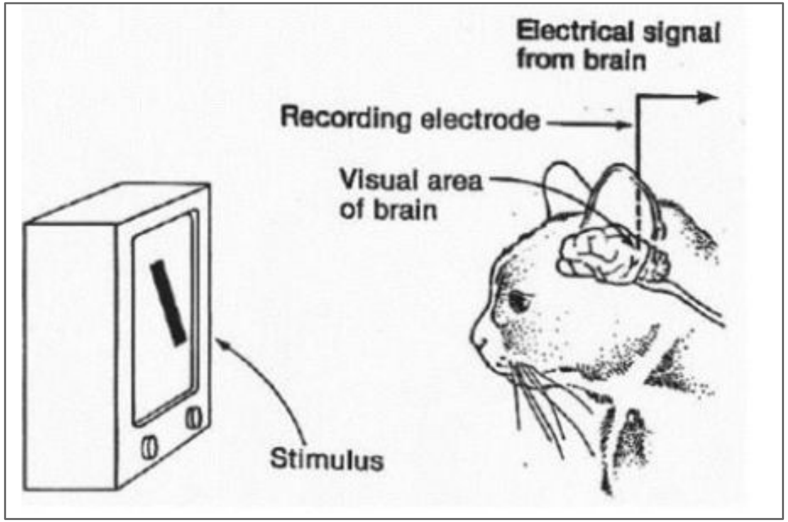


Linear Algebra for ANN Modelling



Biological Motivation: Mammalian Vision System

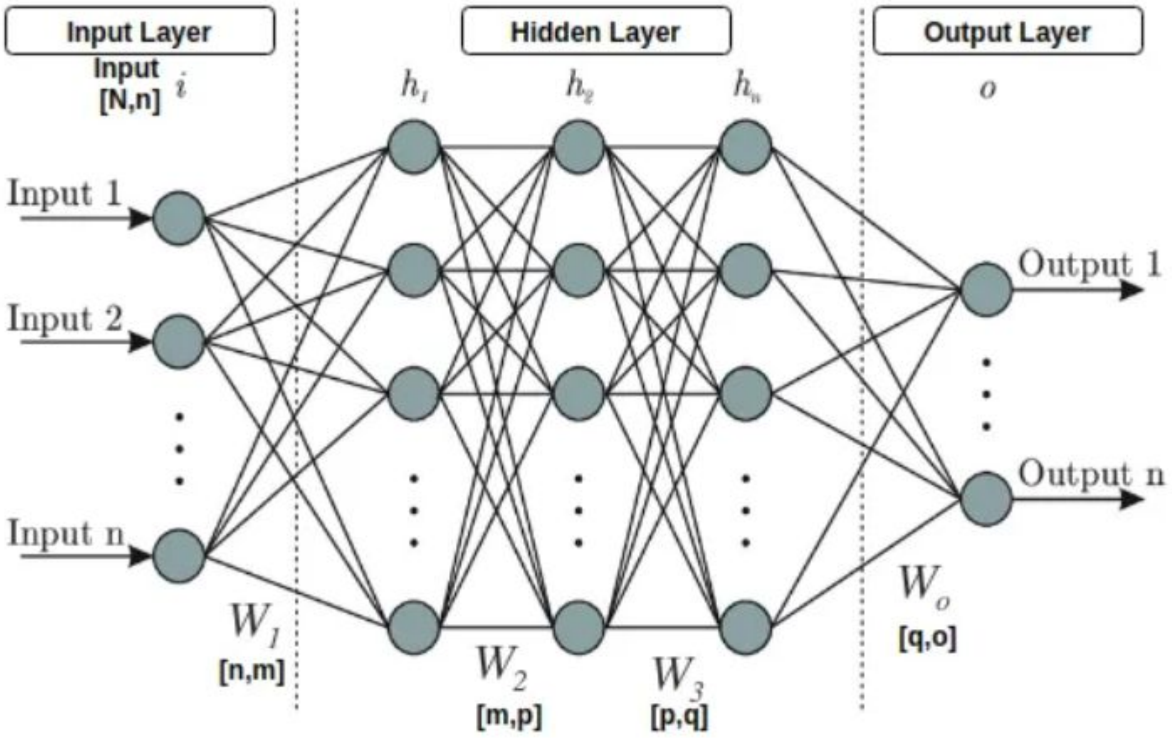
Hubel and Wiesel (1959)
1981 Nobel prize



Experimental setup

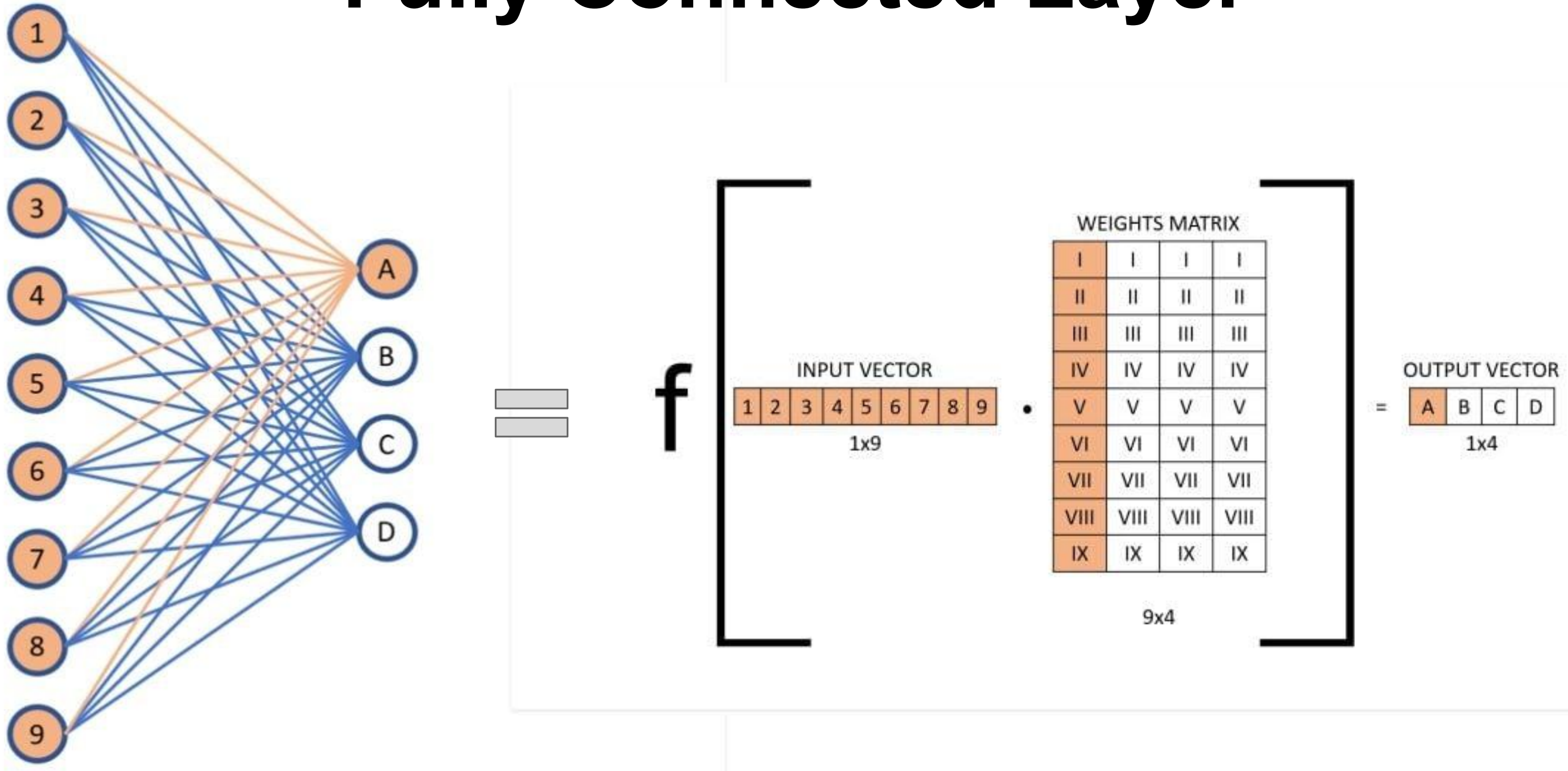
Suggested a ‘hierarchy’ of feature detectors in the mammalian visual cortex.

Hierarchical Feedforward ANN

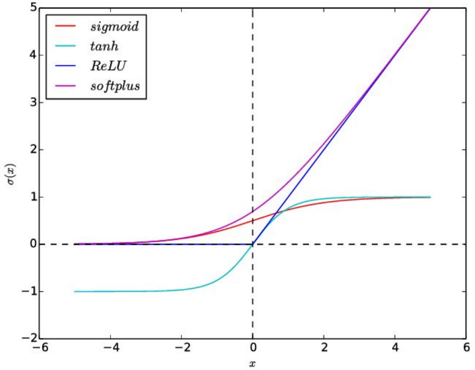


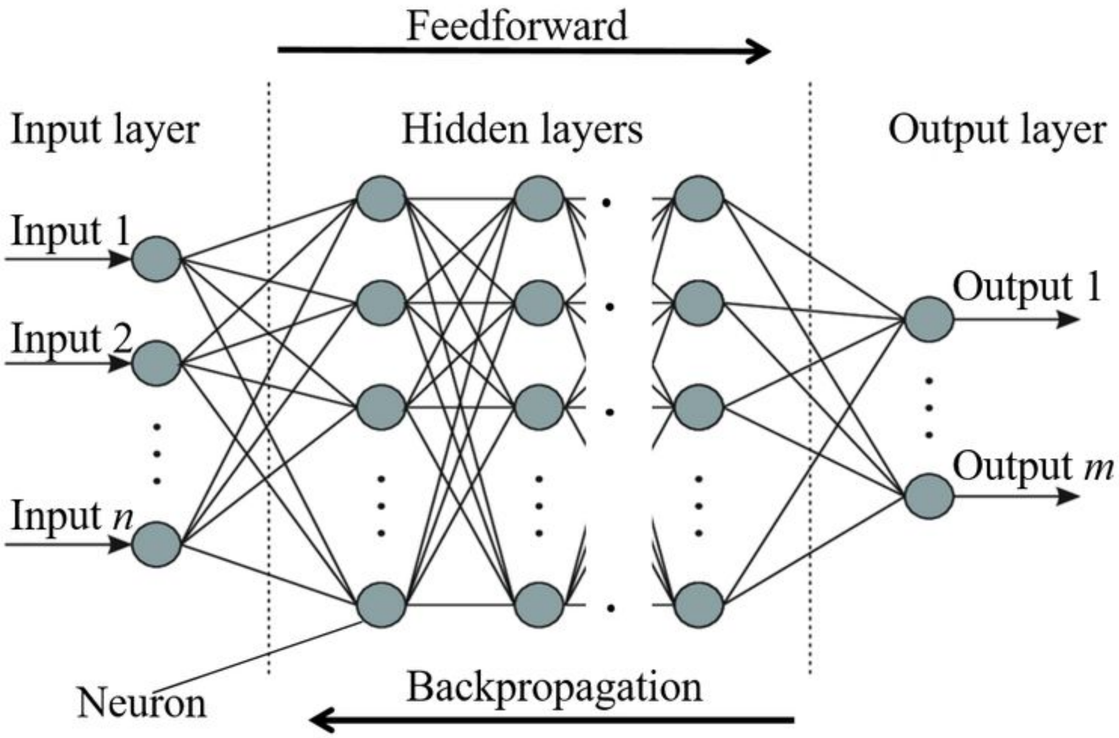
Common Neural Network Architecture

Fully Connected Layer

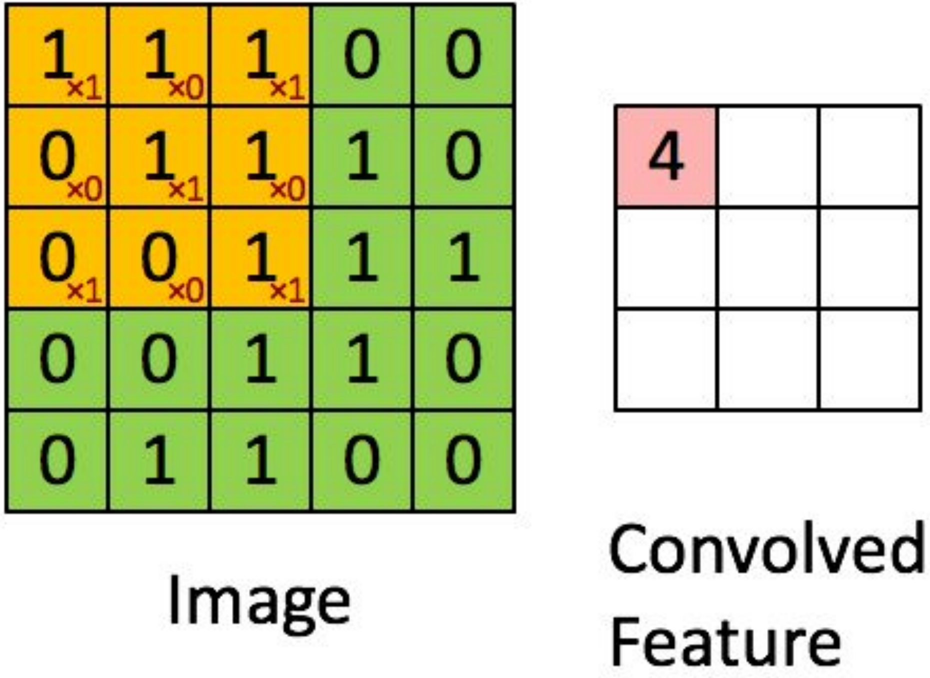


Why We Need Non-Linearities?

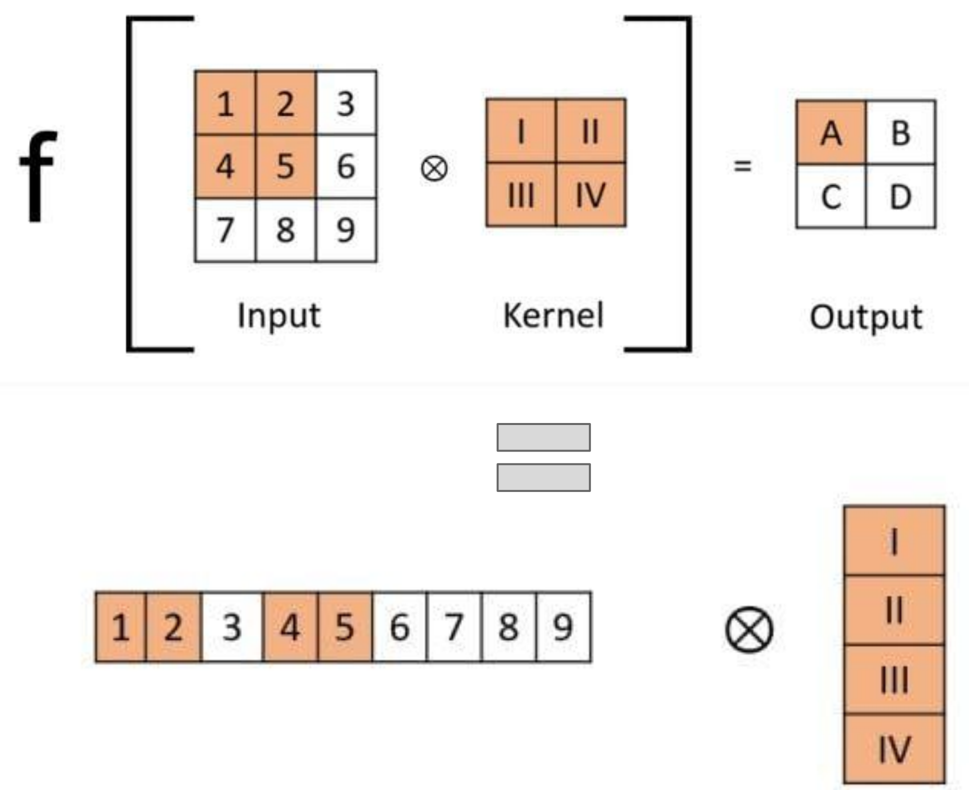




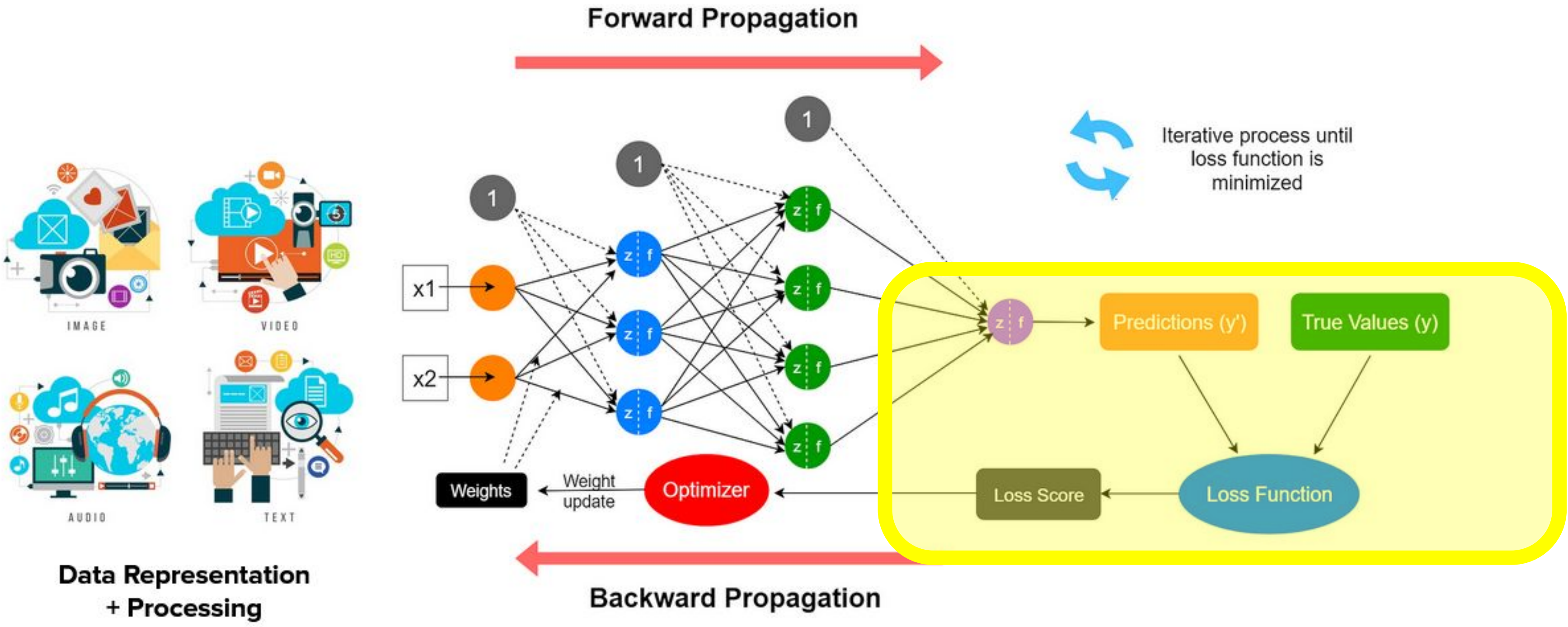
Convolutional Layer



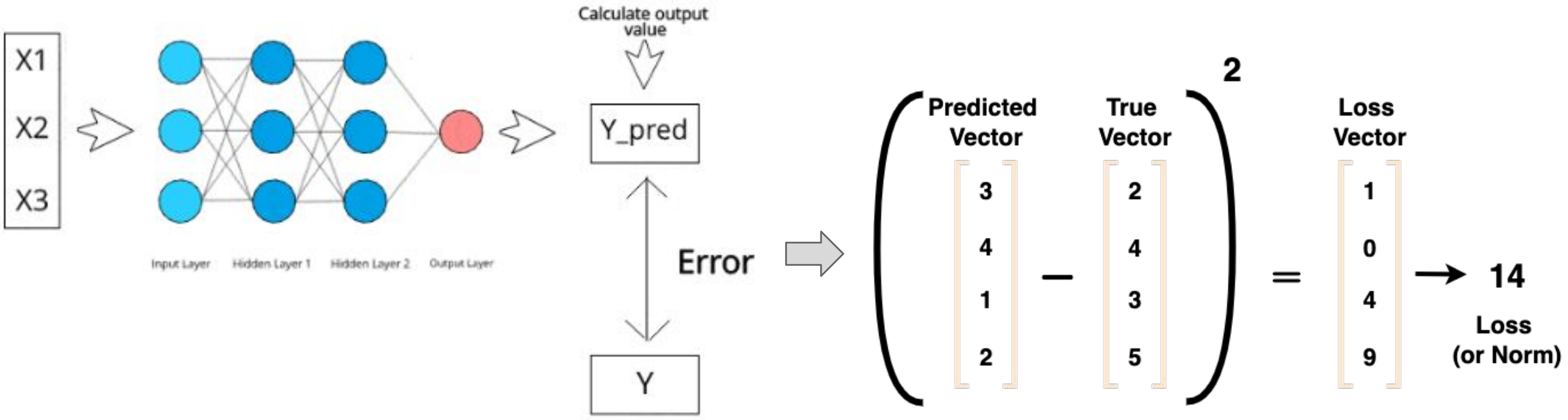
Convolutional Layer



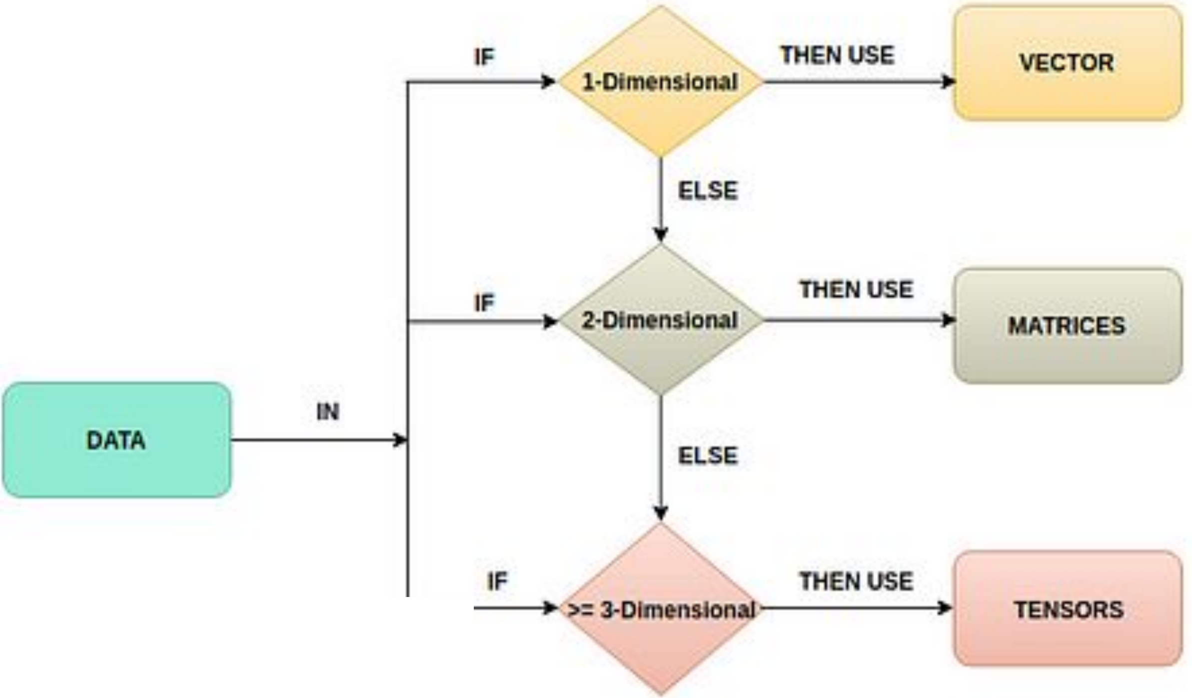
Linear Algebra for ANN Losses



Linear Algebra for ANN Losses

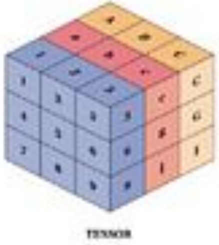


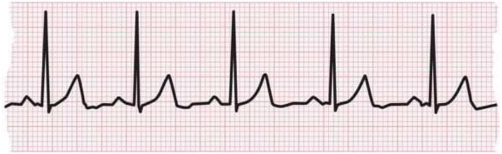
Error Measures (Norms)

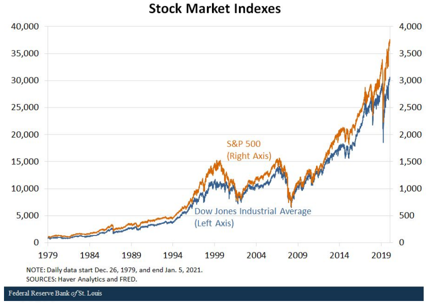


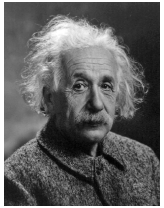
$$\begin{bmatrix} v_1 \\ v_2 \\ \cdot \\ \cdot \\ \cdot \\ v_n \end{bmatrix}$$

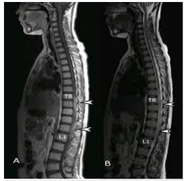
$$\begin{bmatrix} M_{11} & M_{12} & \dots & M_{1n} \\ M_{21} & M_{22} & \dots & M_{2n} \\ \dots & \dots & & \dots \\ M_{m1} & M_{m2} & \dots & M_{mn} \end{bmatrix}$$

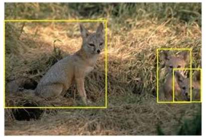


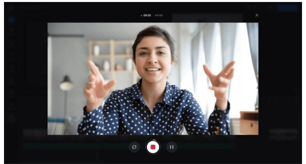




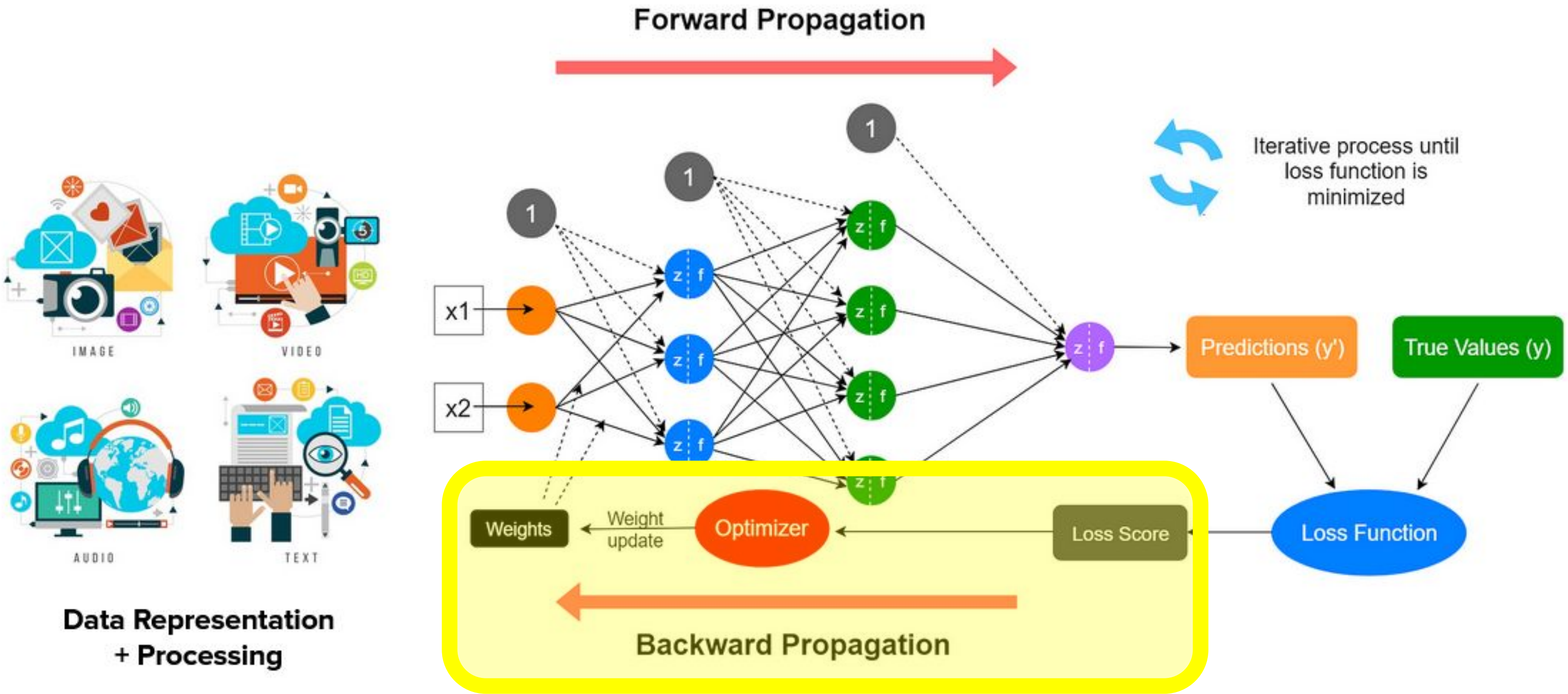




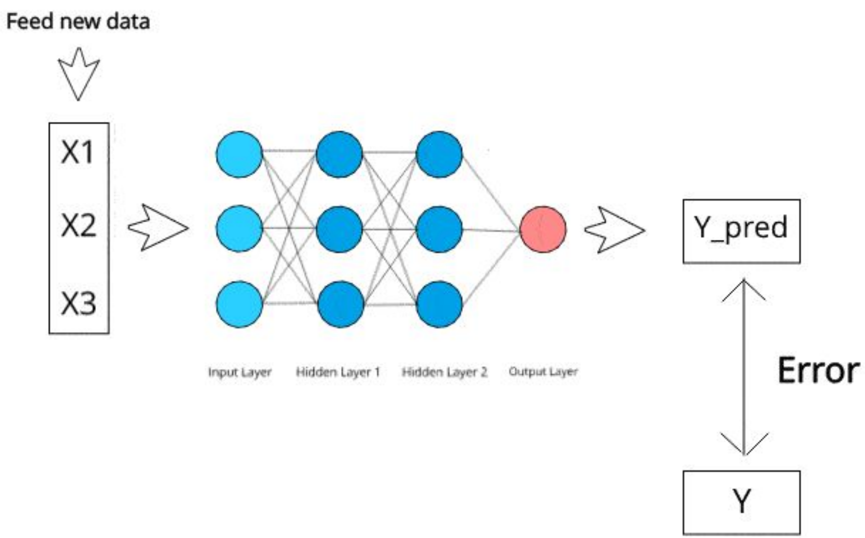


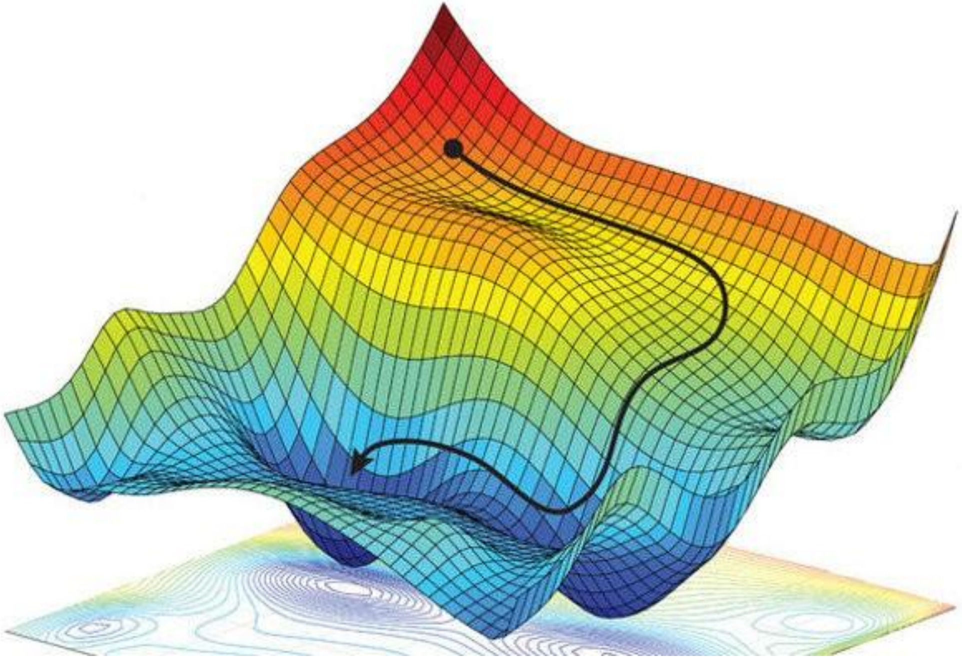


Linear Algebra for ANN Training

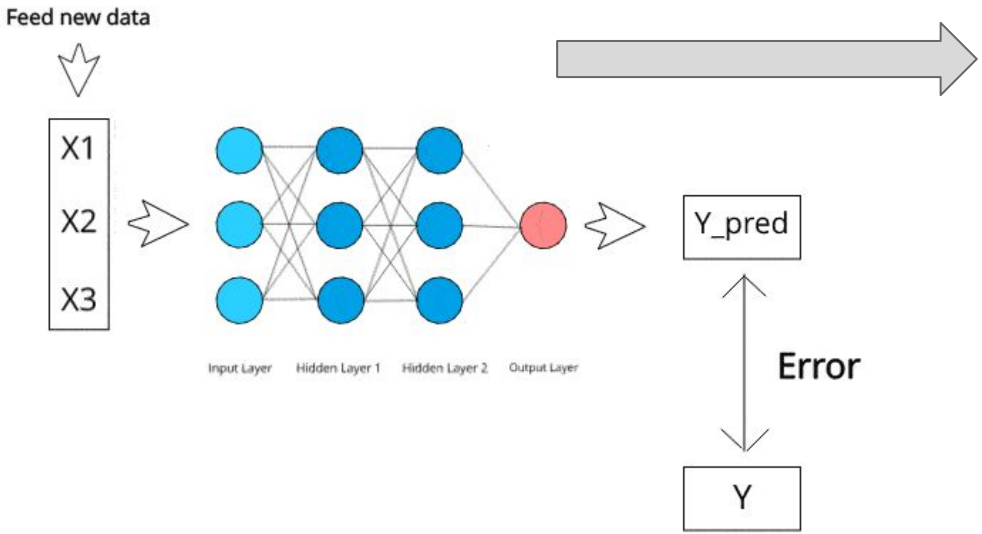


Linear Algebra for ANN Training



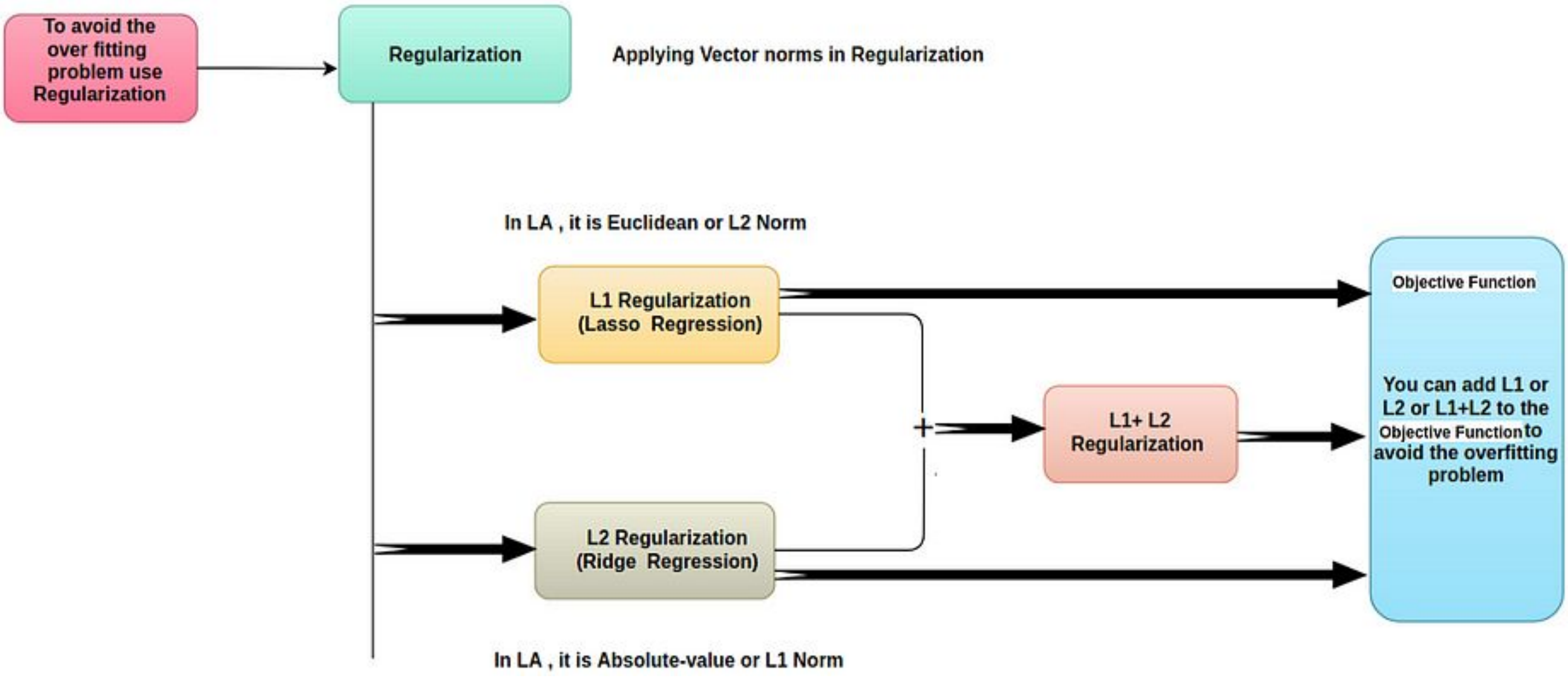


Linear Algebra for ANN Training

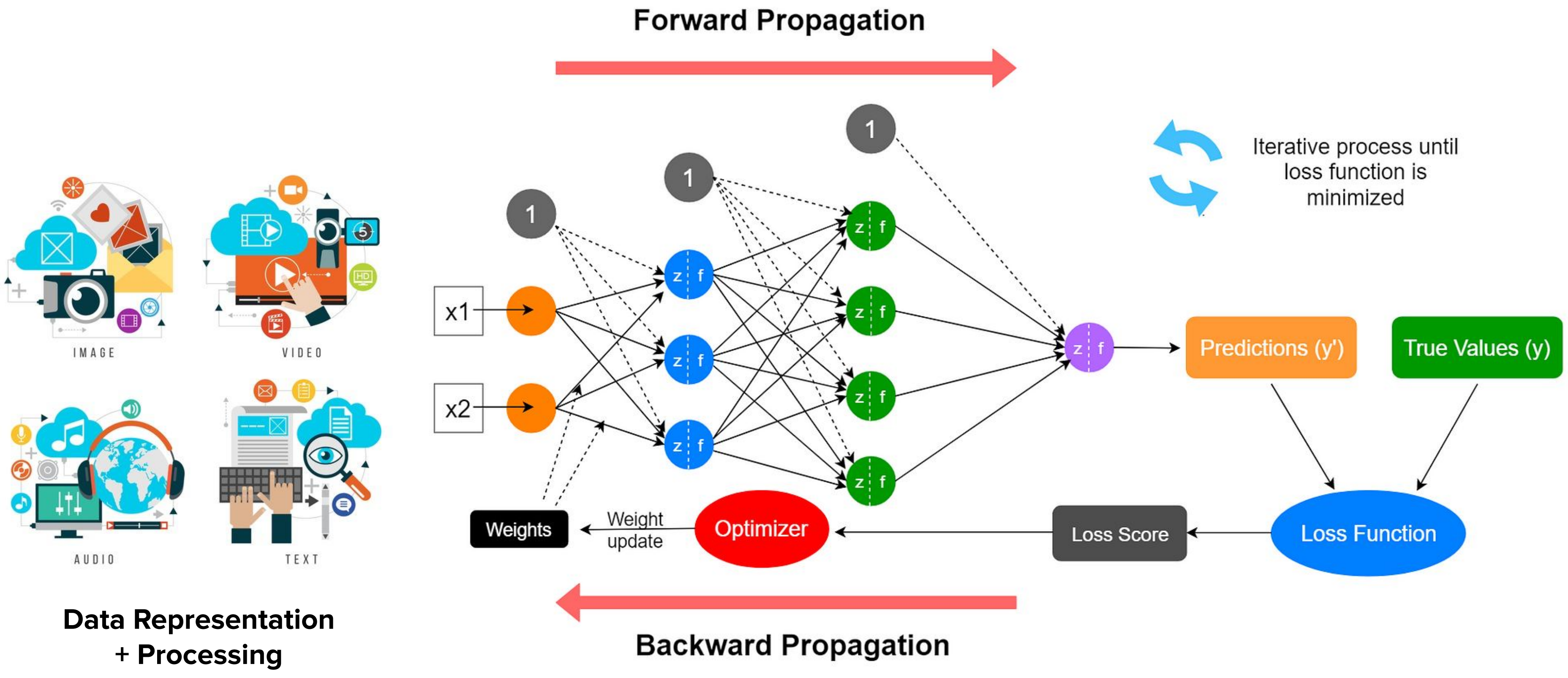


Type	Scalar	Vector	Matrix
Scalar	$\frac{\partial y}{\partial x}$	$\frac{\partial \mathbf{y}}{\partial x}$	$\frac{\partial \mathbf{Y}}{\partial x}$
Vector	$\frac{\partial y}{\partial \mathbf{x}}$	$\frac{\partial \mathbf{y}}{\partial \mathbf{x}}$	
Matrix	$\frac{\partial y}{\partial \mathbf{X}}$		

Linear Algebra for ANN Regularization



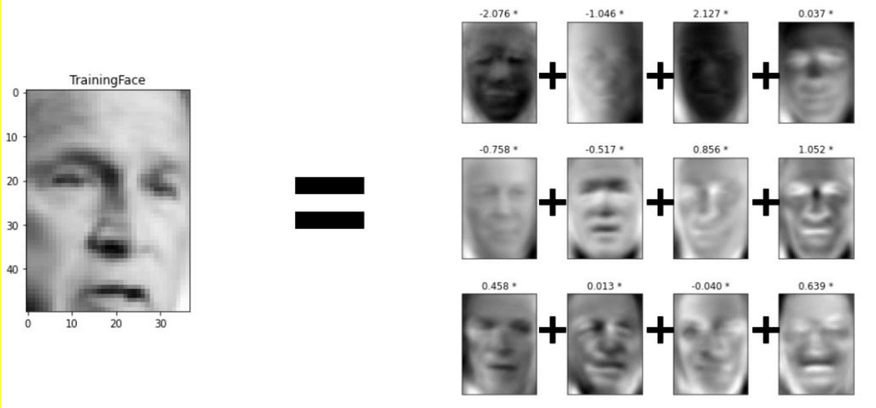
Stages of Neural Network Training



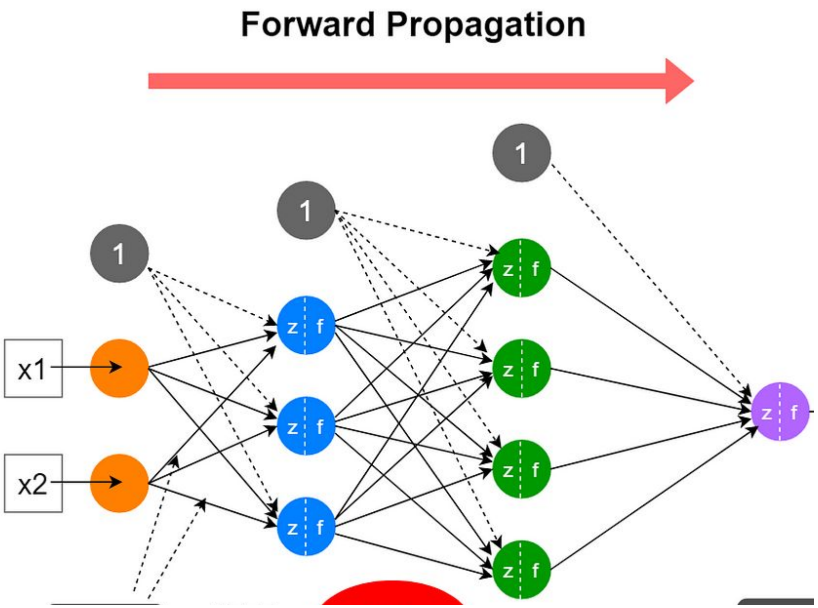
Overview: Stage 1 (Data)

Data Representation + Processing

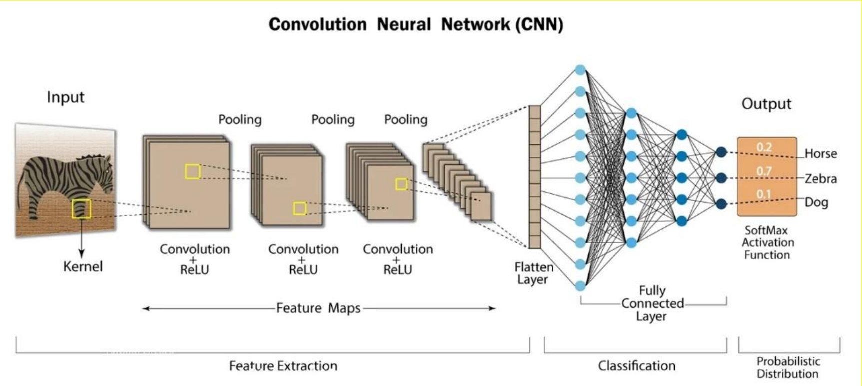
1. Vector Spaces and Subspaces
2. Basis, dimension and Rank
3. Solving Linear Equations
4. Orthogonal Basis and Gram Schmidt Method
5. PCA and SVD
6. Application: Feature Extraction + Dimensionality Reduction



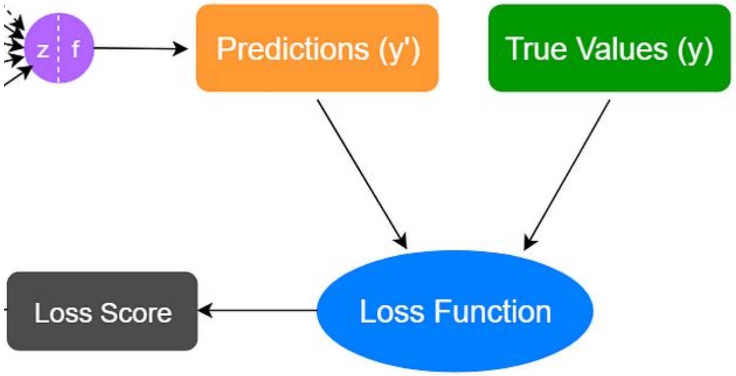
Overview: Stage 2 (ANN Modelling)



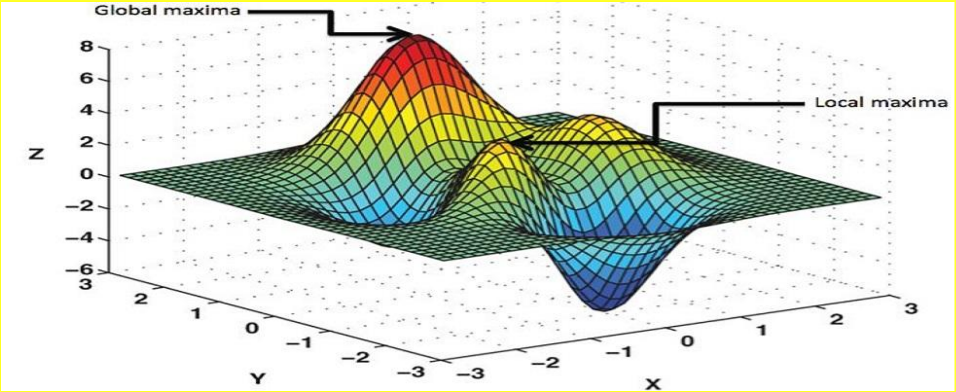
1. Linear Transformations
2. Inner Product and Projections
3. Inequalities
4. Application:
   a. The Construction of Deep Neural Networks
   B. Convolutional Neural Networks



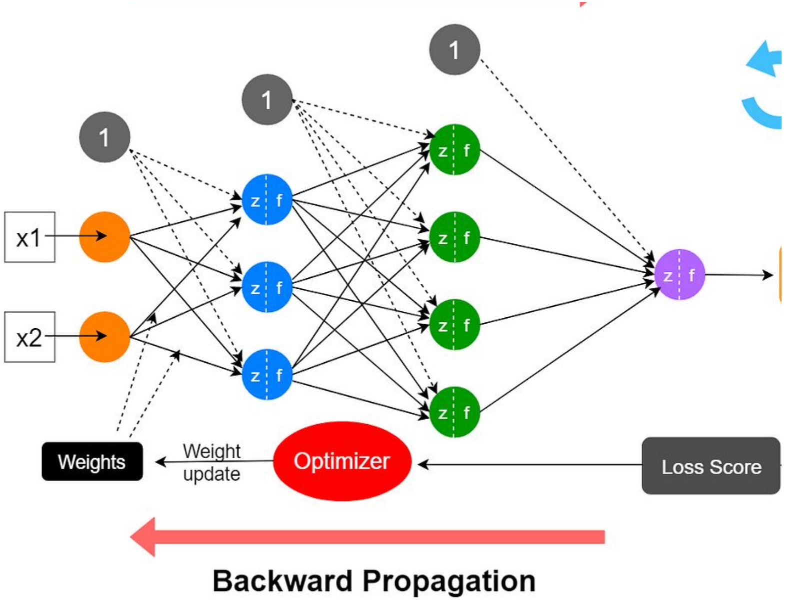
Overview: Stage 3 (Losses)



1. Norms of Vectors, Functions and Matrices
2. L2 and L1 Norms and Interpretations
3. Split Algorithms for L2 and L1
4. Application: Different Loss Functions and Analysis



Overview: Stage 4 (Optimization)



1. Existence and uniqueness of solutions
2. Inverse and pseudo inverse of matrices
3. Introduction to Least Squares
4. Introduction to Gradient Descent and Matrix Differentials
5. Regularization using L2 and L1 Norms
6. Application: Optimization using Closed form and Gradient Descent Algorithms