# PROJECT SUMMARY REPORT: IRIS FLOWER CLASSIFICATION USING ANN

# UNDERSTANDING ARTIFICIAL NEURAL NETWORKS

## 1. What is an Artificial Neural Network (ANN)?

An Artificial Neural Network (ANN) is a computational model inspired by the way biological neural networks in the human brain process information. It consists of layers of interconnected 'neurons' or nodes that transform input data into meaningful outputs by learning patterns from data through training.

## 2.Describe the structure of an ANN (input layer, hidden layers, output layer).

An ANN consists of:

- Input Layer: Receives the input data (features).

- Hidden Layer(s): Perform computations and learn representations using weights and activation functions.

- Output Layer: Produces the final prediction or classification.

## 3.What is the role of activation functions? List commonly used activation functions.

Activation functions introduce non-linearity into the model, allowing the network to learn complex patterns. Common activation functions:

- ReLU (Rectified Linear Unit)

- Sigmoid

- Tanh

- Softmax (used in output layer for multi-class classification)

## 4.Explain the concept of forward propagation and backpropagation.

Forward propagation: Data is passed from input to output layer through weighted connections to generate predictions.

Backpropagation: Errors between predicted and actual output are used to adjust weights using gradient descent, minimizing the loss function.

## 5.What is the difference between a feedforward neural network (FFNN) and a convolutional neural network (CNN)?

FFNN: Data flows only in one direction (forward) without loops. Best for structured/tabular data.

CNN: Special type of network designed to process grid-like data such as images using convolutional layers that detect spatial hierarchies (edges, shapes, objects).

## 6.Why is backpropagation important in training neural networks?

Backpropagation is essential because it enables the network to learn by minimizing the loss function. It computes gradients of the loss with respect to weights and updates them efficiently to improve model accuracy over time.

THEORETICAL UNDERSTANDING OF TRAINING A NEURAL NETWORK

## 7.What are weights, biases, and the learning rate in an ANN?

- Weights: Control the influence of one neuron on another.

- Biases: Allow the activation function to be shifted up or down.

- Learning Rate: A hyperparameter that determines how much weights are adjusted during each step of learning.

## 8.How do the gradient descent and stochastic gradient descent (SGD) algorithms work?

Gradient Descent: Uses all training data to compute the gradient and update weights.

Stochastic Gradient Descent (SGD): Updates weights using one sample at a time. It's faster and introduces randomness which can help escape local minima.

## 9.What are the advantages and disadvantages of common activation functions like ReLU, Sigmoid, and Tanh

ReLU:

+ Efficient and simple

- Can die (output stuck at 0)

Sigmoid:

+ Good for probabilities (0 to 1)

- Can cause vanishing gradients

Tanh:

+ Zero-centered and smooth

- Still suffers from vanishing gradients in deep network.

### **IMPLEMENTATION TASK**

## Objective

The goal of this project was to build a simple Artificial Neural Network (ANN) to classify iris flowers into three species using the famous Iris dataset.

## Dataset

- The Iris dataset contains 150 samples of iris flowers.

- Each sample has 4 features: sepal length, sepal width, petal length, petal width.

- Target classes: Setosa (0), Versicolor (1), Virginica (2).

## Preprocessing

- Used train\_test\_split to split the data (80% train / 20% test).

- Standardized the features using StandardScaler.

- One-hot encoded the target labels using OneHotEncoder (with sparse\_output=False).

## Model Architecture

Built using Keras (TensorFlow backend) as a Sequential model:

- Input Layer: 4 neurons (features)

- Hidden Layer: 5 neurons, ReLU activation

- Output Layer: 3 neurons, Softmax activation (multi-class)

## Training

- Loss Function: categorical\_crossentropy

- Optimizer: sgd (Stochastic Gradient Descent)

- Epochs: 100

- Tracked both training and validation loss/accuracy.

## Evaluation

- Achieved accuracy over 90% on the test set.

- Visualized the confusion matrix to show class-level performance.

- Plotted training and validation loss/accuracy curves to verify learning behavior.

## Insights

- The model successfully learned to classify iris species with high accuracy.

- Loss and accuracy plots showed steady improvement without overfitting.

- The confusion matrix confirmed strong class separation.

Curve Behavior -What it means

Loss decreases - The model is learning

Accuracy increases - The model is getting better at classifying

Val & Train curves close Good generalization (not overfitting!)

## Tools Used

- Python, NumPy, Pandas, Matplotlib, Seaborn

- Scikit-learn for preprocessing and evaluation

- TensorFlow/Keras for model building and training

## Conclusion

This project demonstrated the full machine learning workflow using a simple yet powerful ANN. It covered dataset preparation, model design, training, evaluation, and result visualization. The workflow can now be extended to more complex datasets or deep learning tasks.