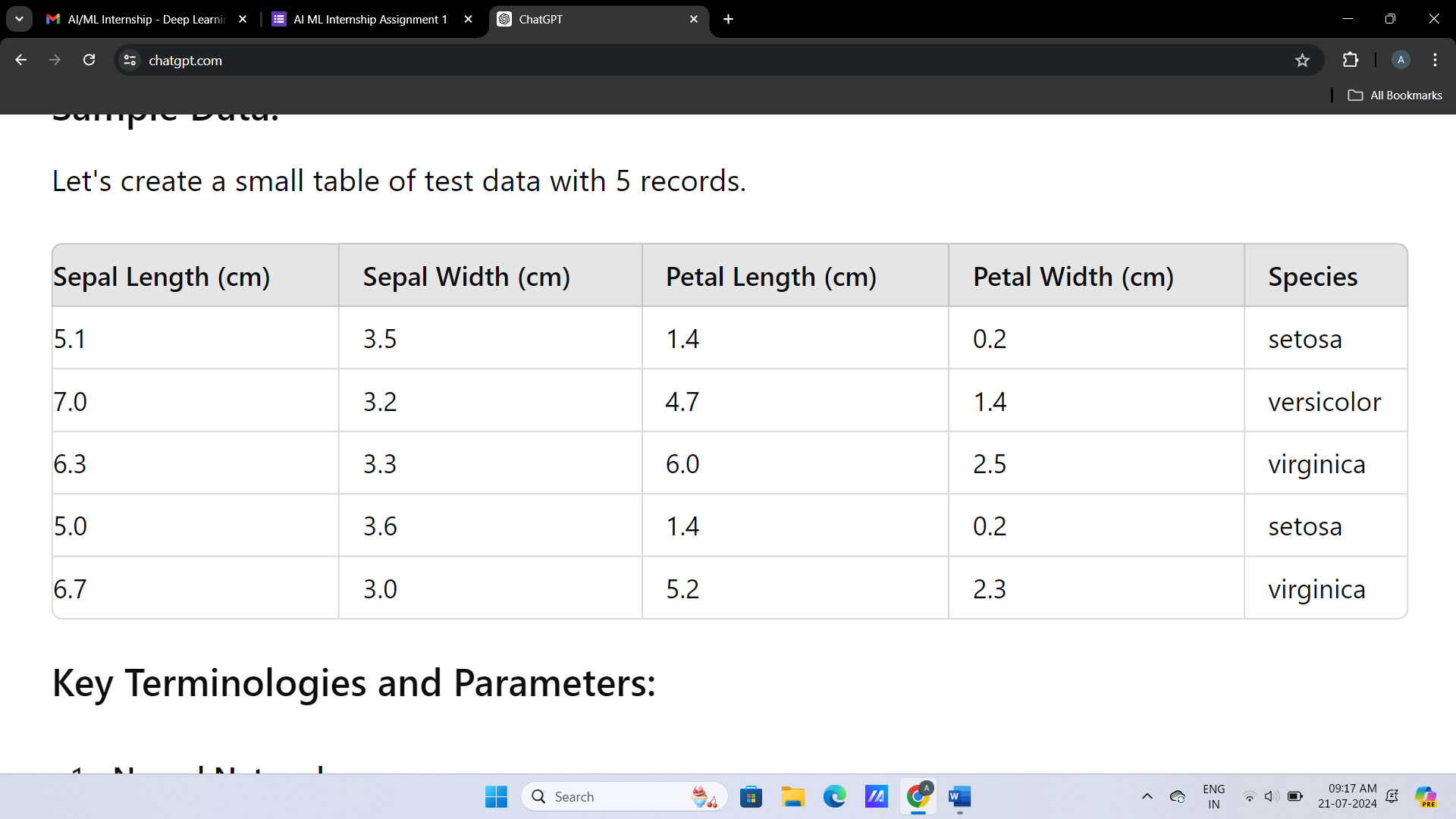
**Problem Description:**

Input: Sepal length, sepal width, petal length, and petal width of Iris flowers.

Output: The species of Iris flower (setosa, versicolor, or virginica).

**Sample Data:**



**Key Terminologies and Parameters:**

1. Neural Network:

Artificial Neural Networks (ANN) A neural network is a computer model that has the ability to perform tasks in generalization, based on how biological neurons in human brains receive input and send output. It assembled with Interconnected nodes vis neurons that work together to Recognize patterns in data. Example: A neural network may be trained on the Iris dataset in order to predict what species a flower belongs to, given its sepal and petal dimensions.

2. Neuron:

A Neuron: A single processing unit in the neural network It has input that goes through some activation function and we will get an output. They are layered with each neuron connected to those in layers above and below.

3. Layer:

A layer is a set of neurons. There are three types of layers.

Input Layer: It is the first layer to which data is fed.

Hidden Layer: Layers between input and output that perform division to identify patterns.

Output layer: this is the last one that gives us our output.

4. Input Layer:

Input layer: The first neural network which is takes the input data. 4 neurons (one for each feature; sepal length,sepal width, petal length and petalwidth) - For the Iris dataset.

5. Hidden Layer:

Intermediate layers connecting the input & output layer These networks take the input data and performs various high level transformations on it using neural network with hidden layers.

The number and size of hidden layers are model hyperparameters that must be defined based on the specific problem.

6. Output Layer:

The output node of the final layer in a neural network It will be 3 for the Iris dataset (one per species).

7. Convolutional Layer:

Convolution Operation: A layer in Convolutional Neural Networks (CNNs) that applies convolution operations to detect features like edges, textures and blobs from the input. Convolutions are mostly used with images and that's why there is your Convolutional layer.

8. Convolutional Neural Network (CNN):

A kind of Neural Network designed for structured grid data such as images. With CNN, convolutional layers are used to automatically and adaptively learn the spatial hierarchies of features from input images. Not typically used for datasets like Iris.

9. Recurrent Neural Network (RNN):

- A neural network designed for sequential data. RNNs are designed with recurrent connections that form directed cycles, and they are able to maintain an internal 'memory' of the previous inputs. They are often used in case of time series data and natural language processing.

10. Activation Function:

Mathematical function applied to each neuron's output, thus deciding whether or not the neuron has to be activated. The Activation Functions drive non-linearity into the network, and allows it to learn complex patterns.

11. ReLU — Rectified Linear Unit

Activation function returning the input itself if it is positive, zero otherwise. ReLU gained enormous fame due to its high computational efficiency and resolving issues with vanishing gradients, and has since been applied widely in deep neural networks.

12. Sigmoid:

Activation function that conveys the input to values ranging between 0 and 1. It is normally used in

the last layer for binary classification problems. The sigmoid function, often denoted by σ, is defined as $\sigma\left(x\right)=\frac{1}{1+e^{-x}}$.

13. Tanh:

- An activation function that squashes the input to a value between -1 and 1. It is very similar to the sigmoid function, but it has an output that ranges from -1 to 1, instead of from 0 to 1, and that has an output that is centered around zero. This may make training easier. The tanh function is defined as $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$.

14. Softmax:

- Activation function used in the final layer for multiclass problems. The raw scores are transformed into probabilities adding up to one: The softmax function is defined as $ \text{softmax}(z\_i) = \frac{e^{z\_i}}{\sum\_{j} e^{z\_j}}. $ intervened

15. Forward Propagation:

- The process of passing the input data through the network layers to get the output prediction. In forward propagation, the output for every neuron is computed and the result passed on to the next layer until the final output is reached.

16. Backpropagation:

Backpropagation is a process of tuning neural network weights based on the error made in output prediction. Computation of a gradient of the loss function, with respect to each weight, and update of weights in order to reduce or minimize loss, is all part of the training process for a neural network.

17. Loss Function:

- A function that quantifies how far the model's output is from the ground truth. Commonly used loss functions are mean squared error in the case of regression and cross-entropy loss for classification.

18. Cost Function:

- Cost function of Neural Network: an overall function being optimized during training, which is usually the sum of the loss function over all training examples. Minimizing the cost function will improve the performance of the network.

19. Gradient Descent:

It is an optimization algorithm to reduce the cost function by moving iteratively in the weights descent direction. There are several variants of gradient descent like batch gradient descent, stochastic gradient descent, and mini-batch gradient descent.

20. Learning Rate:

A hyperparameter that says how big the steps are that gradient descent takes. This controls how fast or slow the model is going to learn. If too high, the learning rate will make the model converge too fast to a suboptimal solution; if it is too low, it will make the training process very slow.

21. Batch Size:

Batch size: the number of training examples used in one iteration of training. It determines how many samples are processed before updating the model's parameters. The typical batch sizes are powers of 2; for example, 32, 64, or 128.

22. Epoch:

- One pass through the entire training dataset. Training includes many epochs because obviously the model will learn progressively from the data.

23. Overfitting:

- When a neural network learns the examples in the training data too well, with all the noise and details, it learns at the cost of generalizing well to new, unseen data. Methods which will help prevent overfitting include regularization, dropout and early stopping.

24. Underfitting:

- A situation where the neural network is not complex enough to model the trends in the training data. Underfitting can be removed by either increasing the complexity of the model or using more training data.

25. Training Set:

- The subset of the dataset used in training the neural network; it learns from this data to tune its weights and minimize the loss function.

26. Validation Set:

- The subset of a dataset that will be used for tuning the neural network's hyperparameters to prevent overfitting. It gives an unbiased assessment of the model at training time.

27. Test Set:

- The subset of the dataset that holds what the final performance of the trained neural network is supposed to be. The test set is not used during training, and hence gives an unbiased estimate of how well it generalizes to new data.

28. Cross-Validation:

A method to estimate the performance of a model based on training and testing a model on different combinations of folds of a split dataset. Cross-validation allows for checking how robust and generalizable a model is.

29. Hyperparameters:

It is the parameter set before training of the neural network; for example, learning rate, batch size, and number of hidden layers. Thus, hyperparameters are not learned from the data themselves, and so therefore need to be tuned for optimal performance.

30. Model Parameters:

The parameters learned by the neural network in training are exactly the weights and biases.

31. Regularization

- To prevent overfitting, methods that add a penalty to the loss function are used in case of large weights. Commonly used regularization methods are L1 (Lasso) and L2 (Ridge) regularization.

32. Dropout

- A regularization method that randomly ignores neurons while training to make sure that the network doesn't get dependent on specific neurons, hence learning robust features.

33. Weight Initialization:

- It is the process by which the weights in a neural network are set to their preliminary values before training. If done correctly, weight initialization will guarantee that it converges more quickly and never experiences the vanishing or exploding gradient problem.

34. Normalization:

It is a process for rescaling input features to a standard range—usually between 0 and 1—with the purpose of making sure that the network trains well. Normalization boosts the speed of training and gives better performance.

35. Standardization:

Zero-mean and unit-variance scaling of input features. This way, standardization ensures that all of the features take equal contribution in the learning process and accelerates the process of convergence.

**Example:**

Record 1:

Sepal Length, cm = 5.1

Sepal Width, cm = 3.5

Petal Length, cm = 1.4

Petal Width (cm) = 0.2

Species = Setosa

- Neural Network: The neural network will take 4 input neurons, one for each feature in the input layer, some number of neurons in a hidden layer, and 3 output neurons for each species.

- Forward Propagation: The input features (5.1, 3.5, 1.4, 0.2) flow through the input layer, get transformed in the hidden layer by activation functions like ReLU, and finally pass through the output layer, where the output will be returned with an activation function of softmax, returning the probability of each class.

• Backpropagation: In case the result predicted by this species does not correspond to the real species, setosa, then it will calculate an error by the loss function, and the neural network is going to adjust its weights to minimize this error in future predictions.

• Training Set: This record and others make up the training set used to train the neural network.

• Validation Set: Another subset of the dataset is used for tuning the hyper-parameters to avoid overfitting.

• Test Set: Measurements on the test set are used to evaluate how well the neural network generalizes to completely new data.