1. What is the difference between TRAINABLE and NON-TRAINABLE PARAMETERS?

**- Trainable Parameters: These are the model parameters that are learned and updated during the training process. They include weights and biases in neural networks. The model learns these parameters to make predictions that minimize a loss function.**

**- Non-Trainable Parameters: These are model parameters that are not learned during training and are typically set beforehand. Non-trainable parameters can include hyperparameters like the architecture of the network, fixed filters in convolutional layers (e.g., pre-trained filters in transfer learning), and constants used for normalization. These parameters are not updated during training.**

2. In the CNN architecture, where does the DROPOUT LAYER go?

- **Dropout layers are typically added after the fully connected (dense) layers in a CNN architecture. They can also be added after specific convolutional layers, especially if those layers have a large number of parameters and are prone to overfitting. Dropout is a regularization technique used to prevent overfitting by randomly deactivating a fraction of neurons during each training iteration.**

3. What is the optimal number of hidden layers to stack?

**- The optimal number of hidden layers in a neural network depends on the specific problem and dataset. There is no one-size-fits-all answer. It's a hyperparameter that needs to be tuned based on experimentation.**

**- For many tasks, shallow networks (with a few hidden layers) are sufficient. However, deeper networks (with more layers) may be necessary for more complex problems or to capture hierarchical features.**

**- Techniques like cross-validation can help determine the appropriate number of hidden layers for a given task.**

4. In each layer, how many secret units or filters should there be?

**- The number of neurons or filters in each layer is a hyperparameter that depends on the problem's complexity, the size of the dataset, and the available computational resources.**

**- There is no fixed number, and it should be determined through experimentation and validation. Common practice is to start with a moderate number and adjust based on performance.**

5. What should your initial learning rate be?

**- The initial learning rate is a hyperparameter that also depends on the problem and the chosen optimization algorithm. There is no one-size-fits-all answer.**

**- Common initial learning rates are in the range of 0.1 to 0.001. It's often a good practice to start with a moderate learning rate and adjust it during training using techniques like learning rate schedules.**

6. What do you do with the activation function?

**- Activation functions introduce non-linearity to the model, allowing it to learn complex patterns. Common activation functions include ReLU, Sigmoid, and Tanh.**

**- You choose an appropriate activation function for each layer based on the problem. ReLU is a popular choice for hidden layers, while Sigmoid or Softmax is used for output layers depending on the task (binary classification or multi-class classification).**

7. What is NORMALIZATION OF DATA?

- **Data normalization is the process of scaling and transforming data so that it falls within a specific range or follows a standard distribution. It's often used to bring data to a common scale, making it easier for machine learning algorithms to converge during training.**

**- Common normalization techniques include Min-Max scaling (scaling to a range like [0, 1]), Z-score normalization (scaling to mean=0, std=1), and feature scaling to a specific range**.

8. What is IMAGE AUGMENTATION and how does it work?

**- Image augmentation is a data preprocessing technique used in computer vision tasks, particularly for deep learning. It involves applying random transformations to training images to increase the diversity of the dataset.**

**- Augmentations can include rotations, flips, translations, zooms, and changes in brightness or contrast. This helps the model generalize better to variations in real-world data and reduces overfitting.**

9. What is DECLINE IN LEARNING RATE?

**- A decline in learning rate refers to the reduction of the learning rate during training. It's often used to fine-tune model convergence. Learning rate schedules can be applied, such as reducing the learning rate by a factor after a fixed number of epochs or when a certain condition is met. The decline in learning rate can help the model converge more accurately in later training stages.**

10. What does EARLY STOPPING CRITERIA mean?

**- Early stopping criteria are conditions set during training to determine when training should be halted before completing all epochs. It is used to prevent overfitting.**

**- Common early stopping criteria involve monitoring a validation metric (e.g., validation loss or accuracy). Training stops when the metric stops improving or starts deteriorating, indicating that further training would lead to overfitting on the validation set.**