1. What is the COVARIATE Shift Issue, and how does it affect you?

**- Covariate shift refers to a situation in machine learning where the distribution of the input features (covariates) in the training data differs significantly from the distribution in the test data.**

**- This can affect the model's performance because it is trained on one distribution and tested on another, causing a mismatch. The model may perform well on the training data but poorly on the test data due to this distribution shift.**

**- Techniques like domain adaptation and data preprocessing are used to mitigate the effects of covariate shift.**

2. What is the process of BATCH NORMALIZATION?

**- Batch normalization is a technique used to improve the training stability and convergence of deep neural networks. The process involves the following steps:**

**1. For each mini-batch during training, calculate the mean and standard deviation of the activations within the batch for each feature channel.**

**2. Normalize the activations by subtracting the mean and dividing by the standard deviation.**

**3. Scale and shift the normalized activations using learnable parameters (gamma and beta) to introduce flexibility.**

**4. Update the running statistics (moving averages of mean and standard deviation) during training for later use during inference.**

**- Batch normalization helps alleviate issues like internal covariate shift and can accelerate training.**

3. Using our own terms and diagrams, explain LENET ARCHITECTURE.

**- LeNet is one of the early convolutional neural network (CNN) architectures designed for image classification. It consists of the following layers:**

**- Convolutional Layer: Detects local features using a small kernel.**

**- Activation Function (e.g., Sigmoid or Tanh): Introduces non-linearity.**

**- Pooling Layer: Reduces spatial dimensions and retains important information.**

**- Fully Connected Layer: Combines features and makes final predictions.**

**- Diagram:**

**```**

**[Input Image] -> [Conv Layer] -> [Activation] -> [Pooling] -> [Conv Layer] -> [Activation] -> [Pooling] -> [Fully Connected] -> [Fully Connected] -> [Output]**

```

4. Using our own terms and diagrams, explain ALEXNET ARCHITECTURE.

**- AlexNet is a deep CNN architecture that revolutionized image recognition. It consists of the following key components:**

**- Convolutional Layers with Increasing Depth: These layers learn hierarchical features.**

**- Rectified Linear Unit (ReLU) Activation: Introduces non-linearity.**

**- Max-Pooling Layers: Reduces spatial dimensions.**

**- Local Response Normalization (LRN): Normalizes local activations.**

**- Fully Connected Layers: Combines features for classification.**

**- Diagram:**

**```**

**[Input Image] -> [Conv Layer] -> [ReLU] -> [Max-Pooling] -> [Conv Layer] -> [ReLU] -> [Max-Pooling] -> [Conv Layer] -> [Conv Layer] -> [Conv Layer] -> [ReLU] -> [Fully Connected] -> [Fully Connected] -> [Output]**

```

5. Describe the vanishing gradient problem.

**- The vanishing gradient problem occurs during the training of deep neural networks, particularly recurrent neural networks (RNNs) and deep feedforward networks. It happens when the gradients of the loss function with respect to the model's parameters become extremely small as they are backpropagated through layers.**

**- This problem affects the ability of the model to learn long-range dependencies and can lead to slow convergence or stagnation. It is often associated with activation functions like sigmoid or tanh, which squash values into a limited range, causing gradients to diminish during backpropagation.**

**- Techniques like ReLU activations and gradient clipping are used to mitigate the vanishing gradient problem.**

6. What is NORMALIZATION OF LOCAL RESPONSE?

**- Local Response Normalization (LRN) is a layer used in some neural network architectures, including AlexNet. It performs normalization on local activations within a convolutional layer. The normalization is applied to each pixel in the feature map independently.**

**- LRN is designed to increase the selectivity of neurons by enhancing the response of neurons with strong activations relative to their local neighbors.**

**- It uses a scaling parameter and a local normalization window to perform the normalization.**

7. In AlexNet, what WEIGHT REGULARIZATION was used?

**- In AlexNet, weight regularization was applied to the fully connected layers using L2 weight decay. L2 weight decay adds a regularization term to the loss function that penalizes large weights. This regularization helps prevent overfitting by encouraging**