Early stopping is a regularization technique used in **deep learning (DL)** to prevent overfitting by stopping the training process when the model's performance on a validation dataset stops improving. It is particularly useful when training deep neural networks, as these models often tend to overfit with prolonged training.

How Early Stopping Works

1. **Monitor a Metric**: Early stopping monitors a specific metric on the validation set, such as validation loss or accuracy.
2. **Set a Patience Parameter**: Define a number of epochs (patience) to wait for improvement in the monitored metric.
3. **Stop Training**: If the metric does not improve for the specified patience period, training is stopped early, and the model's weights from the best epoch are restored.

Steps in Early Stopping

1. Train the model on the training dataset while evaluating it on the validation dataset.
2. Track the metric of interest (e.g., validation loss).
3. When the metric stops improving for `patience` epochs:
    - Stop training.
    - Roll back the model weights to the point where the metric was the best.

Why Use Early Stopping?

- **Avoid Overfitting**: Stops training before the model starts overfitting the training data.
- **Save Time and Resources**: Prevents unnecessary epochs, saving computation time and energy.
- **Optimize Generalization**: Ensures the model performs well on unseen data.

Implementation in Python (Using Keras)

Here’s an example using the `EarlyStopping` callback in Keras:

```
python
```

```
from tensorflow.keras.callbacks import EarlyStopping # Define early stopping
early_stopping = EarlyStopping( monitor='val_loss', # Metric to monitor (e.g., validation
loss) patience=5, # Number of epochs with no improvement to wait
restore_best_weights=True # Restore weights from the best epoch ) # Train the model with
early stopping history = model.fit( X_train, y_train, validation_data=(X_val, y_val),
epochs=100, # Maximum number of epochs batch_size=32, callbacks=[early_stopping] )
```

Key Parameters in EarlyStopping (Keras)

1. `monitor` : Metric to monitor (e.g., `'val_loss'` , `'val_accuracy'`).
2. `patience` : Number of epochs to wait for improvement before stopping.
3. `restore_best_weights` : Whether to restore the model's weights from the epoch with the best monitored metric.
4. `min_delta` : Minimum change in the monitored metric to qualify as an improvement.

Advantages of Early Stopping

- Reduces overfitting.
- Saves computational resources.
- Simple and easy to implement.

Disadvantages

- Requires a validation set, which can reduce the data available for training.
- Choosing the patience value can sometimes be tricky.

When to Use Early Stopping

- In models with a high risk of overfitting.
- When training deep networks where the validation performance is expected to plateau.