# ▾ Diagnose and Address Underfitting and Overfitting

## 1. Diagnosing Underfitting:

Underfitting occurs when a model is too simplistic to capture the underlying patterns in the data. It performs poorly on both the training and test sets. Signs of underfitting include:

- Low training and test performance (low accuracy, high error).

- Consistently poor performance across different datasets or folds in cross-validation.

- Model doesn't seem to learn from the training data.

**Addressing Underfitting:**

- **Increase Model Complexity:** Consider using a more complex model with more parameters, such as using deeper neural networks, higher-degree polynomial regression, or more complex algorithms.

- **Feature Engineering:** Add more relevant features to the dataset to provide the model with more information.

- **Fine-tuning Hyperparameters:** Adjust hyperparameters like learning rate, regularization strength, or the number of hidden units/layers in a neural network.

- **Reduce Regularization:** If you're using regularization techniques, consider reducing the strength of regularization or using a different type.

**Reasons for Underfitting:**

- High bias and low variance.

- The size of the training dataset used is not enough.

- The model is too simple.

- Training data is not cleaned and also contains noise in it.

**Techniques to Reduce Underfitting:**

- Increase model complexity.

- Increase the number of features, performing feature engineering.

- Remove noise from the data.

- Increase the number of epochs or increase the duration of training to get better results.

## 2. Diagnosing Overfitting:

Overfitting occurs when a model becomes too flexible and fits the training data noise and outliers. It performs very well on the training set but poorly on the test set. Signs of overfitting include:

- High training performance but significantly lower test performance.
- Large differences between training and test performance.
- Model captures noise and fluctuations in the training data.

### Addressing Overfitting:

- **Regularization:** Apply regularization techniques to penalize overly complex models. Common methods include L1 regularization (Lasso), L2 regularization (Ridge), and dropout in neural networks.
- **Feature Selection:** Remove irrelevant or noisy features that might be contributing to overfitting.
- **More Data:** Increase the size of your training dataset to provide the model with more examples to learn from.
- **Early Stopping:** Monitor the performance on the validation set during training and stop training when performance starts to degrade.
- **Simpler Model:** Consider using a simpler model architecture with fewer parameters. Ensemble Methods: Combine predictions from multiple models to reduce overfitting.

### Reasons for Overfitting:

- High variance and low bias.
- The model is too complex.
- The size of the training data.

### Techniques to Reduce Overfitting:

- Increase training data.
- Reduce model complexity.

- Early stopping during the training phase (have an eye over the loss over the training period as soon as loss begins to increase stop training).

- Ridge Regularization and Lasso Regularization.

- Use dropout for neural networks to tackle overfitting.

### 3. Using Cross-Validation:

Cross-validation is a powerful tool to diagnose both underfitting and overfitting. If a model performs poorly on both training and validation sets across multiple folds, it might indicate underfitting. If the model performs very well on training but poorly on validation, it suggests overfitting.