

Underfitting

- **What It Is:** Underfitting happens when a model is too simple to understand the patterns in the data. It's like trying to fit a straight line to data that needs a curve.
- **Signs of Underfitting:**
 - The model makes many mistakes on both the training data (the data it learned from) and new data.
 - It doesn't perform well on either the data it was trained on or on data it hasn't seen before.
- **Why It Happens:**
 - The model is too basic (like using a straight line for data that requires something more complex).
 - The model hasn't been trained long enough.
 - The data used to train the model doesn't have enough useful information.

Overfitting

- **What It Is:** Overfitting occurs when a model learns the details and noise in the training data too well. This is like memorizing answers instead of understanding the material, so it struggles with new questions.
- **Signs of Overfitting:**
 - The model does great on training data but fails on new data.
 - It performs well on the data it was trained on but poorly on new or unseen data.
- **Why It Happens:**
 - The model is too complex (like trying to fit a wavy line to data that needs a straight line).
 - The model has been trained for too long and starts to pick up noise along with the important patterns.

Regularization

- **Why It's Important:** Regularization is a set of techniques used to prevent overfitting by simplifying the model or adding constraints.
- **How It's Done:**
 - **L1 Regularization (Lasso):** Adds a penalty to the model based on the absolute value of the parameters, which can make some parameters zero and lead to a simpler model.
 - **L2 Regularization (Ridge):** Adds a penalty based on the squared values of the parameters, discouraging large parameter values and making the model less complex.

- **Dropout:** During training, randomly ignore some neurons in the network, which prevents the model from relying too much on any single neuron.
- **Early Stopping:** Stop training when the model starts to perform worse on validation data, preventing it from overfitting.
- **Data Augmentation:** Increase the variety of the training data by applying random changes, helping the model learn more robust features.
- **Batch Normalization:** Regularly adjusting the inputs of each layer to stabilize and speed up training, which also helps in reducing overfitting.