Dropout is a regularization technique used in neural networks to prevent overfitting. Overfitting occurs when a model learns the training data too well, including its noise and outliers, leading to poor generalization on new, unseen data. Dropout helps mitigate this by randomly "dropping out" (i.e., setting to zero) a fraction of the neurons during each training iteration. This forces the network to learn redundant representations and thus become more robust.

**Reasons for Using Dropout:**

1. **Prevent Overfitting**:
   * By randomly dropping neurons, dropout prevents the model from becoming too dependent on any single neuron, which can lead to overfitting.
2. **Promote Redundancy**:
   * Encourages the network to learn multiple independent features, making the model more robust.
3. **Regularization**:
   * Acts as a form of regularization similar to L1 or L2 regularization by adding noise to the training process.
4. **Improve Generalization**:
   * Helps the model generalize better to unseen data by reducing the chance of overfitting.

**How Dropout Works:**

* During training, each neuron is kept with a probability **p** (e.g., 0.5), and with probability **1-p**, it is set to zero.
* During testing, dropout is not applied. Instead, the weights are scaled by the dropout probability **p** to maintain the same expected output.

**Dropout in Practice:**

* Dropout is typically applied after fully connected (Dense) layers or between convolutional layers.
* The dropout rate is usually between 0.2 to 0.5. The choice of dropout rate can be tuned based on the specific dataset and model.