

## Dropout Regularization in Machine Learning – Extended Summary

**Dropout** is a regularization technique used in neural networks to reduce **overfitting**. It works by **randomly "dropping out" (disabling)** a subset of neurons during each training iteration. This means that on every forward pass, different neurons are temporarily removed from the network, preventing the model from becoming too dependent on any particular path or feature.

---



### Dropout Rate and Keep Probability

- **Dropout Rate** refers to the **fraction of neurons that are dropped** (i.e., temporarily ignored) during training.
  - Example: A dropout rate of **0.5** means **50% of the neurons are dropped**.
- **Keep Probability** is the **complement** of the dropout rate — it refers to the **probability that a neuron remains active**.
  - Keep Probability = **1 - Dropout Rate**
  - Example: If dropout rate = **0.3**, then keep probability = **0.7**.

Some libraries (like TensorFlow) use **keep probability**, while others (like PyTorch) use **dropout rate**.

---



### Why It Works

Dropout forces the network to learn redundant representations, since no neuron can rely on another being present during training. This helps prevent overfitting and improves generalization to new data. It's like training an ensemble of subnetworks and averaging their predictions.

---



### During Inference

At test time (or inference), **dropout is turned off**, and the full network is used. To maintain consistency in the output, the weights or activations are usually **scaled** (e.g., multiplying by the keep probability) so the output magnitudes match what the network learned during training.