LECTURER: Nghia Duong-Trung

DEEP LEARNING

MACHINE LEARNING
TOPIC OUTLINE

UNIT 2

Neural Network Training

STUDY GOALS

- After completing this unit you will be able to ...

  … describe how neural networks are trained using backpropagation.

  … explain the gradient descent method.

  … explain how neural networks are initialized.

  … optimize the learning rate in neural network training.

  … use regularization methods to avoid overtraining and improve the generalization ability of neural networks.

INTRODUCTION TO NEURAL NETWORK TRAINING

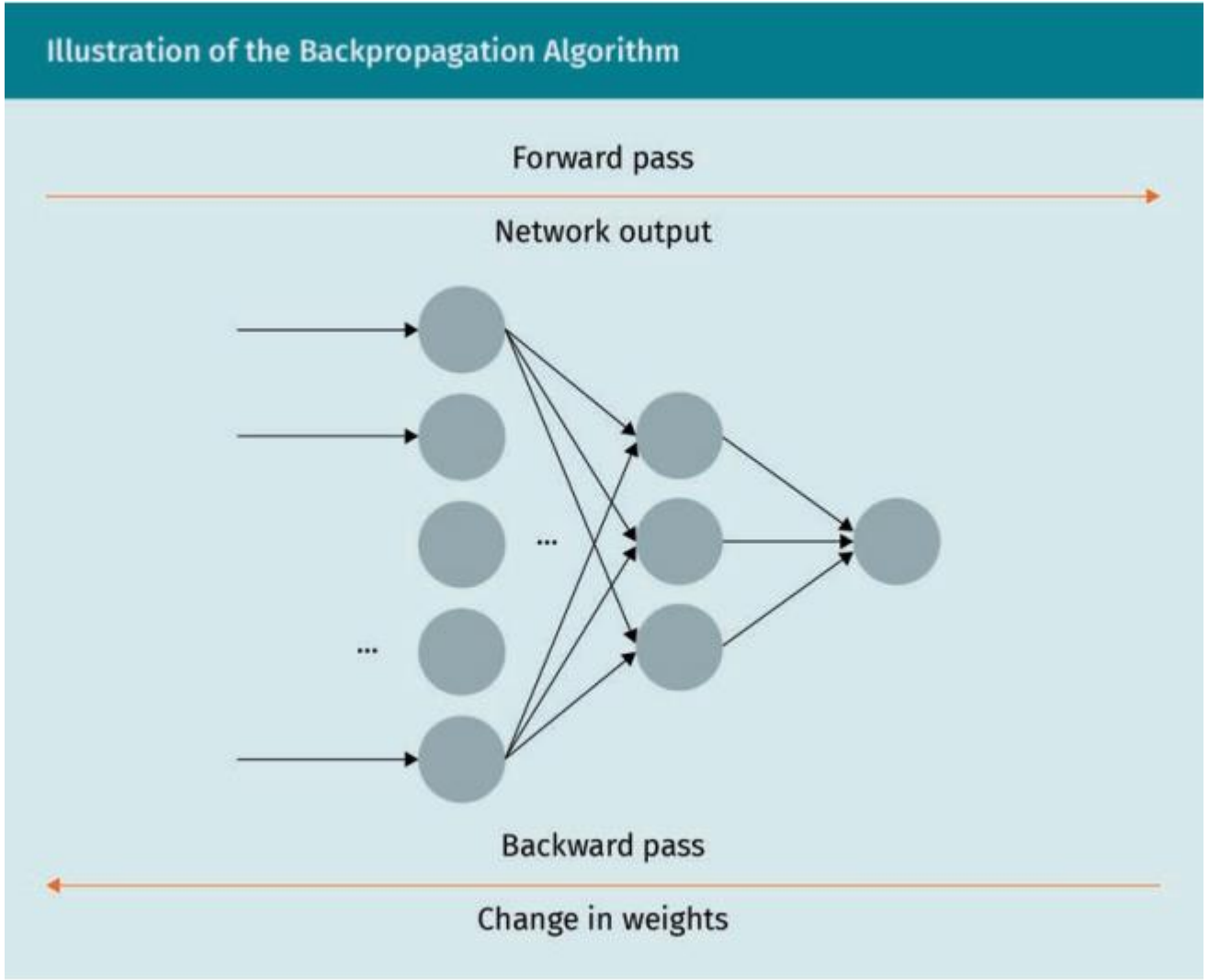
- Neural networks are trained using supervised learning, where the network parameters are iteratively adjusted so that a set of known training patterns consisting of input signals and an output signal is reproduced with a small error.
- During the training process, the input patterns are presented to the network, and the computed output is compared to the true or desired output .
- Then, the network parameters are adjusted, and the process is repeated until the deviation between the network response and true signal output falls below a dedicated threshold or the preset number of training iterations has been reached.

INTRODUCTION TO NEURAL NETWORK TRAINING

- To perform the training using backpropagation, we need to define the following ingredients:
  - a neural network architecture defining the neural network we want to train. This includes the number and types of network layers (e.g., fully-connected, convolutional, or recurrent), the number of nodes in each layer, and other parameters specifying the details of the network architecture.
  - how to initialize the neural network architecture before training starts. Naively, this should not matter; however, experience shows that the initialization scheme can have an impact on the training process.
  - a cost or error function which comparing the computed and desired true outputs of the training patterns
  - the optimal learning rate as the training progresses
  - a “suitable” transfer or activation function
  - a regularization scheme to avoid or limit overtraining

BACKPROPAGATION AND GRADIENT DESCENT

- The main idea of the backpropagation algorithm is to iteratively adjust the weights of the neural networks connecting the neurons in the various layers.

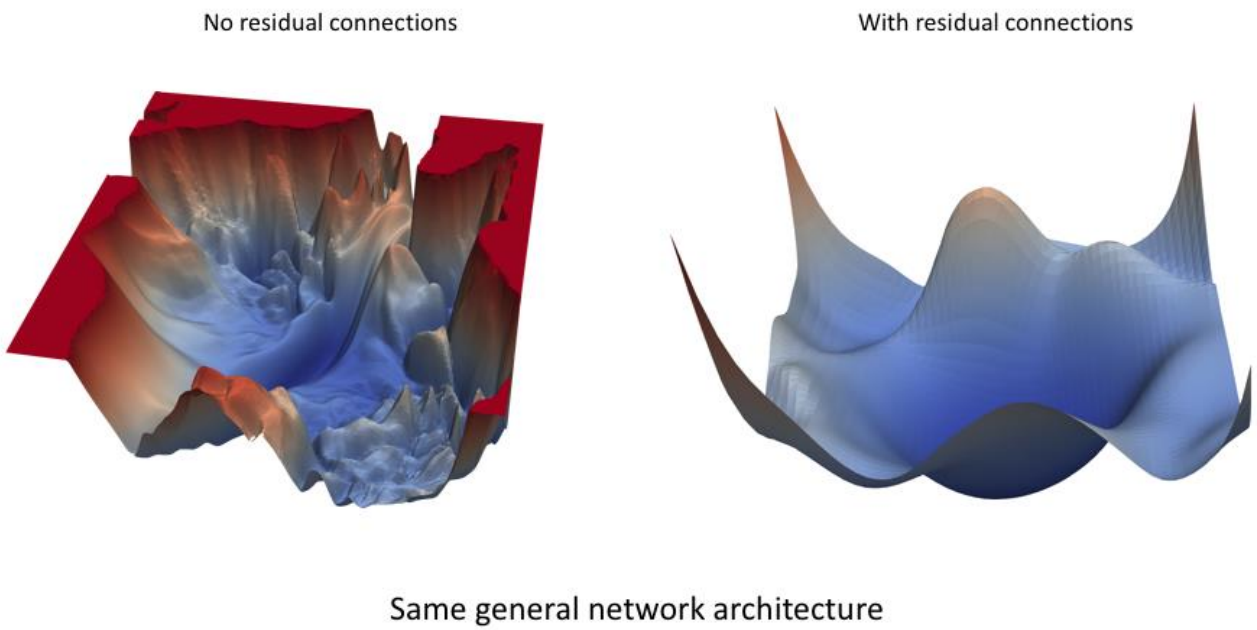


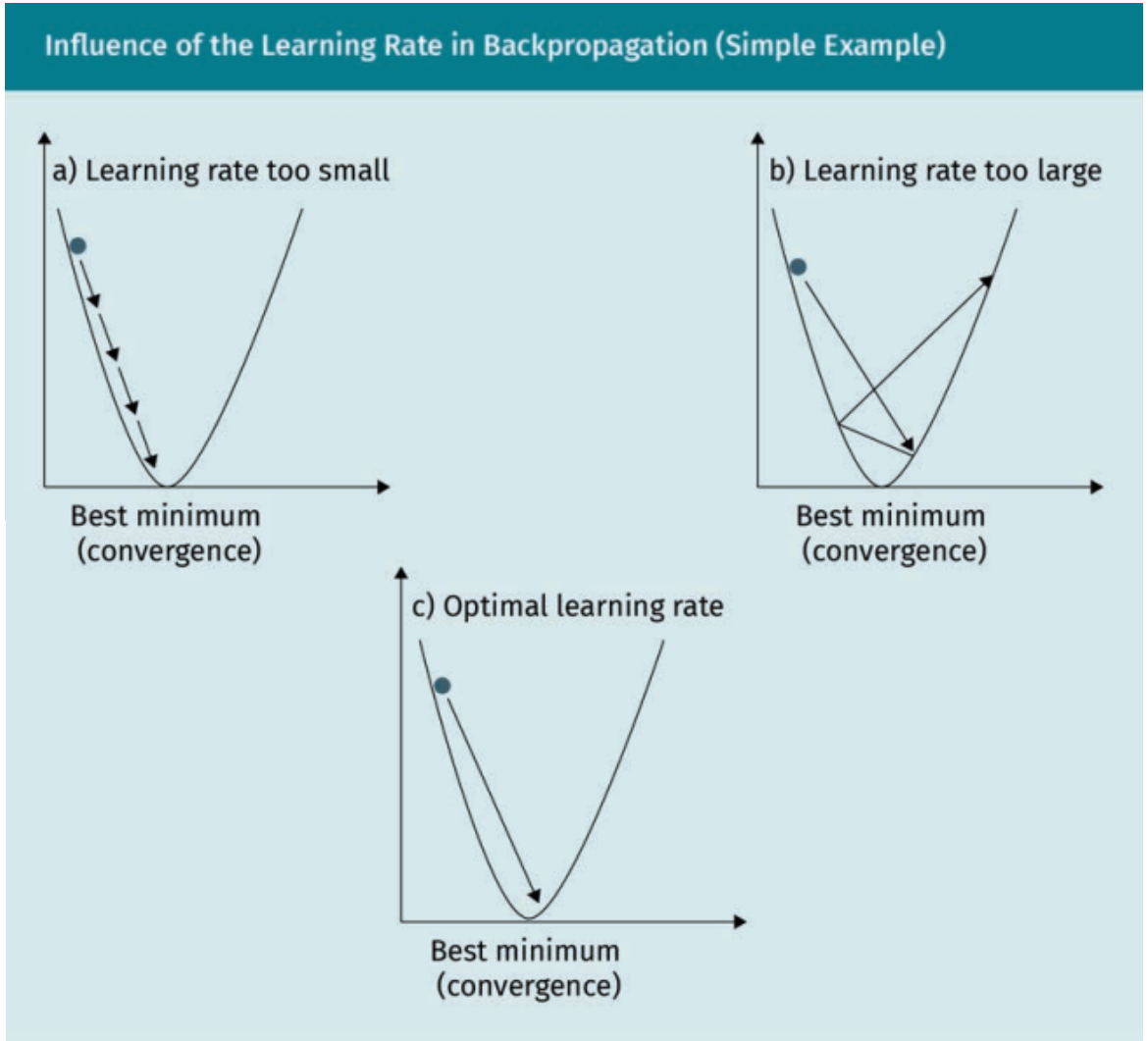
TRANSFER TASK

- https://playground.tensorflow.org/
- https://nnplayground.com/
- https://distill.pub/2020/grand-tour/

OPTIMAL LEARNING RATE

- the learning rate α, which connects the derivatives calculated in the backward pass of the backpropagation algorithm to the actual change in the weights as the neural network is trained





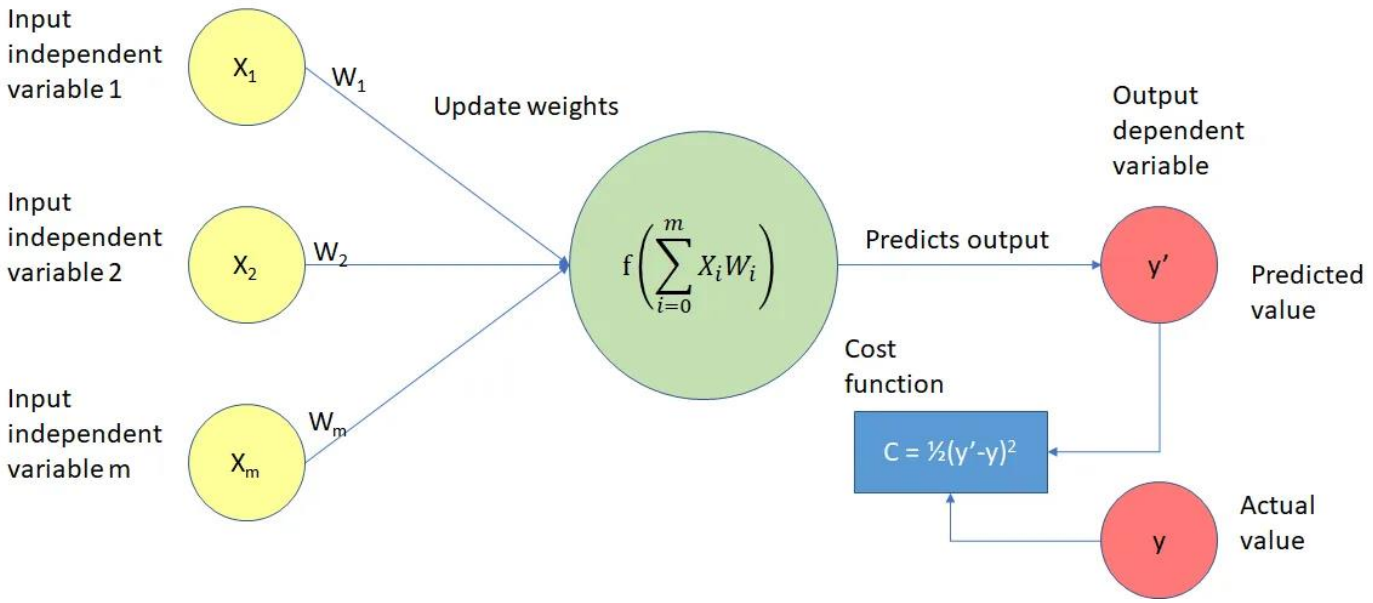
https://www.jeremyjordan.me/nn-learning-rate/

TRANSFER TASK

- Try the notebook CNN > 02. PyTorch_CIFAR.ipynb
  - Make experiments with different learning rates

COST FUNCTIONS

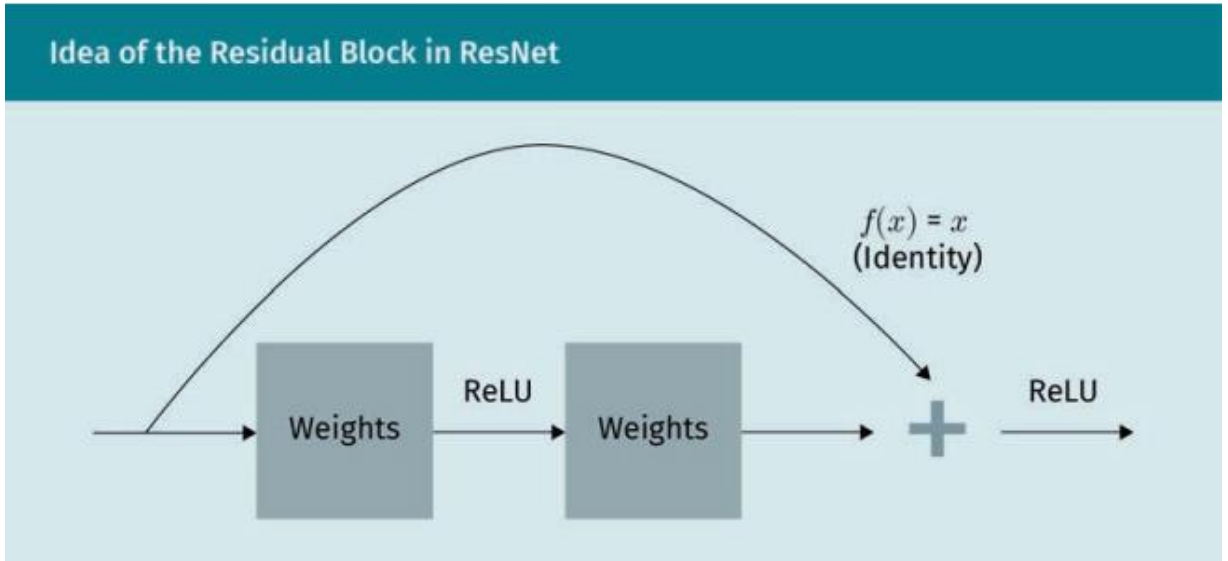
- A cost function is a measure of error between what value your model predicts and what the value actually is.
- Cost and loss functions are synonymous (some people also call it error function). The more general scenario is to define an objective function first, which you want to optimize. This objective function could be to.
  - maximize the posterior probabilities (e.g., naive Bayes)
  - maximize a fitness function (genetic programming)
  - maximize the total reward/value function
  - maximize information gain/minimize child node impurities
  - minimize a mean squared error cost (or loss) function
  - maximize log-likelihood or minimize cross-entropy loss (or cost) function
  - minimize hinge loss (support vector machine)



https://medium.com/@zeeshanmulla/cost-activation-loss-function-neural-network-deep-learning-what-are-these-91167825a4de

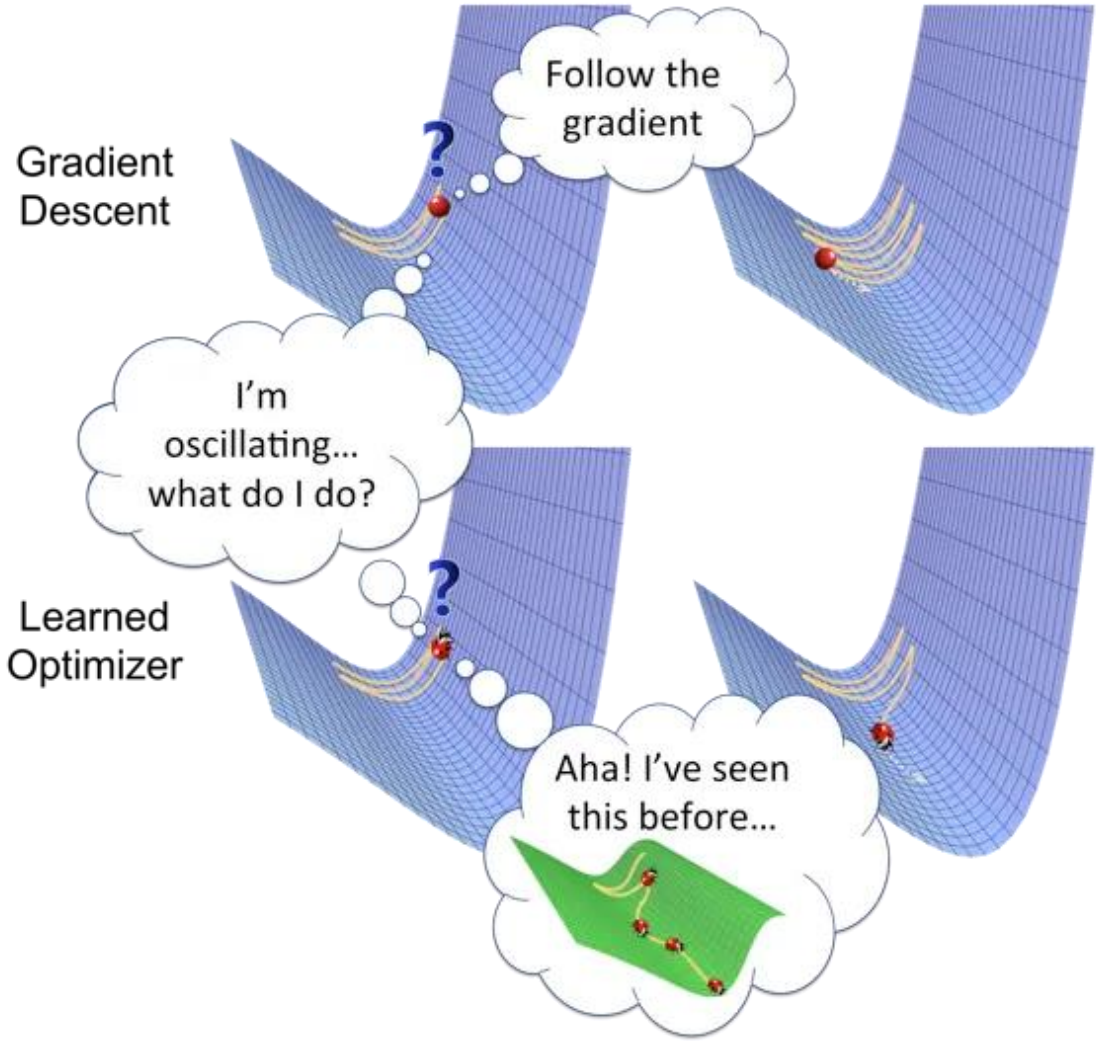
THE VANISHING GRADIENT PROBLEM

- As more layers using certain activation functions are added to neural networks, the gradients of the loss function approaches zero, making the network hard to train.
- Certain activation functions, like the sigmoid function, squishes a large input space into a small input space between 0 and 1. Therefore, a large change in the input of the sigmoid function will cause a small change in the output. Hence, the derivative becomes small.
- For shallow network with only a few layers that use these activations, this isn't a big problem. However, when more layers are used, it can cause the gradient to be too small for training to work effectively.
- The simplest solution is to use other activation functions, such as ReLU, which doesn't cause a small derivative.
- Residual networks are another solution, as they provide residual connections straight to earlier layers.



OPTIMIZERS

- Optimizers are algorithms or methods used to change the attributes of your neural network such as weights and learning rate in order to reduce the losses.
- Adam: https://pythonguides.com/adam-optimizer-pytorch/



https://towardsdatascience.com/optimizers-for-training-neural-network-59450d71caf6

WEIGHT INITIALIZATION

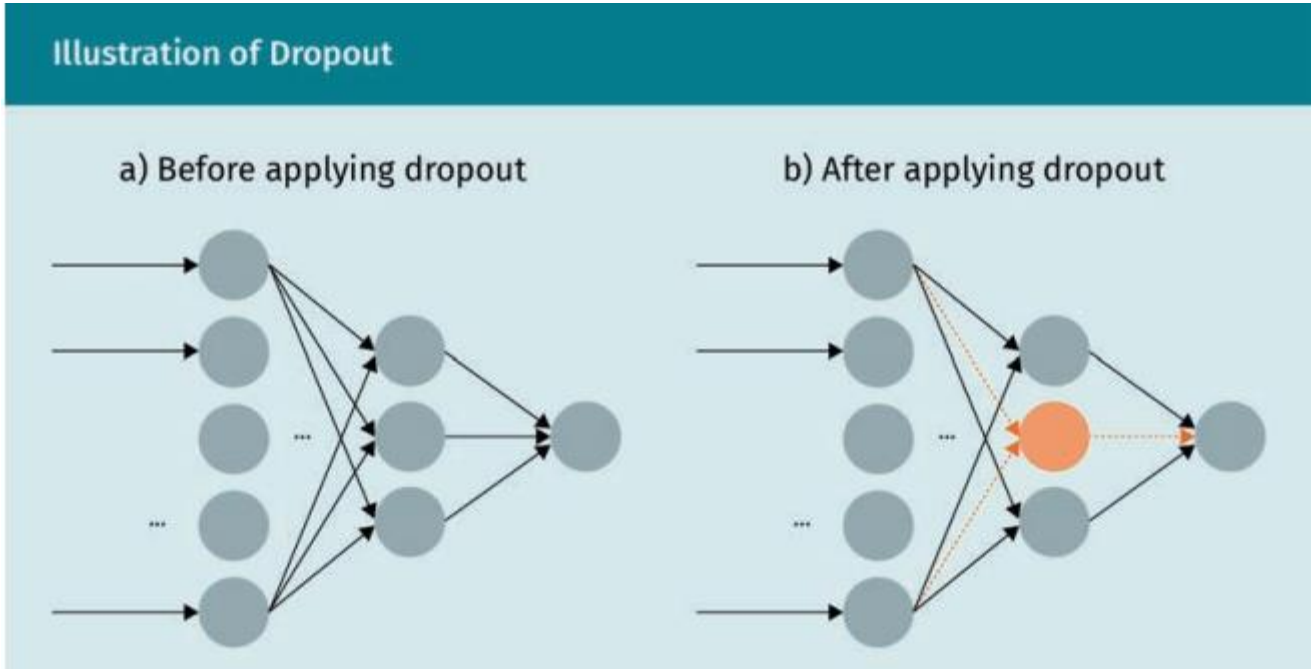
- While building and training neural networks, it is crucial to initialize the weights appropriately to ensure a model with high accuracy. If the weights are not correctly initialized, it may give rise to the Vanishing Gradient problem or the Exploding Gradient problem. Hence, selecting an appropriate weight initialization strategy is critical when training DL models.
  - Zero initialization
  - Random initialization
  - Xavier/Glorot Initialization
  - Normalized Xavier/Glorot Initialization
  - He Uniform Initialization
  - He Normal Initialization

MOMENTUM

- The **learning rate** is a scalar value that determines how much the weights of a neural network are updated during training. A high learning rate can cause the weights to overshoot the optimal values, while a low learning rate can result in slow convergence or getting stuck in a local minimum.
- **Momentum**, on the other hand, is a value that determines how much the previous weight update influences the current update. It can help the network overcome local minima and speed up convergence.
- **Overfitting** occurs when a neural network becomes too complex and starts to fit the **noise** in the training data instead of the underlying patterns. In this case, reducing the learning rate and increasing the momentum can help the network generalize better to new data.
- Change learning rate and momentum over time: Decay, Momentum schedule, adaptive learning rate

DROPOUT

- Dropout builds on the idea that small neural networks are generally preferable to large ones as they minimize the risk of overtraining.
- In this approach, individual neurons are removed from the network (including all their weights or connections connecting the neuron to other neurons) at random with some probability p.
- This way, each training cycle of the input patterns is completed with a small, sparse sub-network of the original setup.



TRANSFER TASK

- Run the notebook CNN\01. PyTorch_CNN_Fashion_MNIST.ipynb

REVIEW STUDY GOALS

- After completing this unit you will be able to ...

  … describe how neural networks are trained using backpropagation.

  … explain the gradient descent method.

  … explain how neural networks are initialized.

  … optimize the learning rate in neural network training.

  … use regularization methods to avoid overtraining and improve the generalization ability of neural networks.

SESSION 1

Neural Network Training

1. What is back propagation?

A. It is another name given to the curvy function in the perceptron.

B. It is the transmission of error back through the network to adjust the inputs.

C. It is the transmission of error back through the network to allow weights to be adjusted so that the network can learn.

D. None of the mentioned.

2. Which of the following functions can be used as an activation function in the output layer if we wish to predict the probabilities of n classes (p1, p2..pk) such that sum of p over all n equals to 1?

A. Tanh.

B. Softmax.

C. ReLu.

D. Sigmoid.

3. Which of the following activation functions can lead to vanishing gradients?

A. ReLU.

B. Tanh.

C. Leaky ReLU.

D. None of the mentioned.

4. After training a neural network, you observe a large gap between the training accuracy (100%) and the test accuracy (42%). Which of the following methods is commonly used to reduce this gap?

A. Generative adversarial networks.

B. Dropout.

C. Sigmoid activation.

D. RMSprop optimizer