

Reflection Week 9

Summary

This week's lectures introduce more advanced topics in the field of machine learning, including deep learning and its subset of neural networks.

Concepts

- Perceptron: an algorithm for supervised learning of binary classifiers. A perceptron can represent models like AND and OR. It consists of a single layer of input neurons, each of which is connected to a single output neuron. Each neuron is associated with a weight value that determines the strength of the connection, and the output neuron sums up the weighted input value and applies an activation function to determine its output.
- Neural network uses the idea of combining perceptrons in layers to learn more complex functions, which includes layers of linear models and nonlinearities (activation function)
- Activation function: a mathematical function that is applied to the output of each neuron in a hidden layer to transform the output of the neuron into a non-linear representation so that the model can learn complex relationships between the input and output data

Different types of activation functions:

- Sign activation function: not differentiable and no notion of confidence
- Sigmoid function: historically popular, not zero-centered
- Hyperbolic tangent $g(x) = \tanh(x)$ works like sigmoid function but zero-centered
- Rectified linear unit (ReLU) is popular nowadays but fragile during training (neurons "die off" if input to a neuron is negative), which can be fixed with its variants such as "leaky" or "noisy" ReLU.
- among others...
- Neural network overfitting can be prevented by adding more training data, reducing hidden nodes, regularization (drop out), or early stopping.
- Applications: computer vision, classification and regression problems,
- Backpropagation: a supervised learning method that involves updating the weights of a neural network in order to minimize the differences between the predicted outputs and actual outputs given a training dataset. It works by "propagating" the output errors back through the layers and update the weights of the neurons in each layer to reduce the error using gradient descent.
- Convolutional Neural Networks: using convolutions (operations that apply a kernel to an input image or signal to produce a feature map)
- Pooling is used to downsample an image by combining local pixels with some operation (min, max, average)
- Applications of Convolutional Neural Networks: Object Detection, Image detection, recommendation

Concerns

- Is there any potential issue/risk by using convolutions to downsample input data?
- When to use deep learning vs. normal machine learning?