silde6

Tuesday, June 22, 2021 5:45 PM

TutorialSlide6

STATS5099: Data Mining

Neural networks

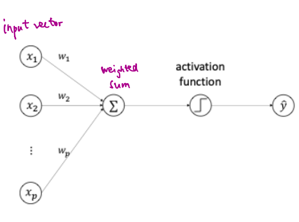
Xiaochen Yang
xiaochen.yang@glasgow.ac.uk

This week's content

- artificial neuron
- activation function
- feedforward neural networks
- gradient descent, backpropagation

Artificial neuron

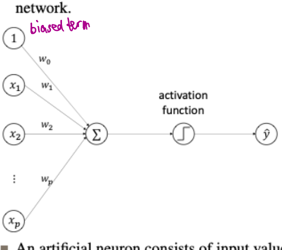
- Artificial neuron, aka perceptron, is the basic unit of a neural network.



$$\hat{y} = \phi(w_0 + \sum w_i x_i)$$

Artificial neuron

- Artificial neuron, aka perceptron, is the basic unit of a neural network.

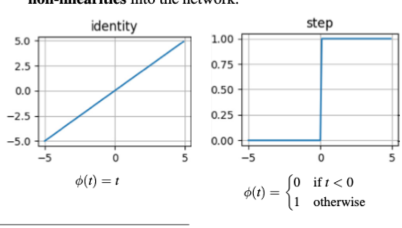


$$\hat{y} = \phi(w_0 + \sum w_i x_i)$$

- An artificial neuron consists of input values, weights and a bias, a weighted sum and activation function.

Activation functions

- The purpose of activation functions is to **introduce non-linearities** into the network.



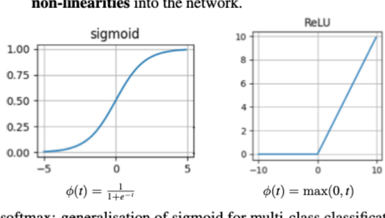
$\phi(t) = t$

$$\phi(t) = \begin{cases} 0 & \text{if } t < 0 \\ 1 & \text{otherwise} \end{cases}$$

Image source: Apicella, Andrea, et al. "A survey on modern trainable activation functions." Neural Networks (2021).

Activation functions

- The purpose of activation functions is to **introduce non-linearities** into the network.



$\phi(t) = \frac{1}{1+e^{-t}}$ $\phi(t) = \max(0, t)$

softmax: generalisation of sigmoid for multi-class classification

$$p_i(y_i) = \frac{e^{y_i}}{\sum_j e^{y_j}}$$

Two special cases of neural networks

- A neural network with no hidden layer and the linear activation function is simply a linear regression model.
- A neural network with no hidden layer and the logistic/sigmoid activation function is a logistic regression model.

Neural network construction

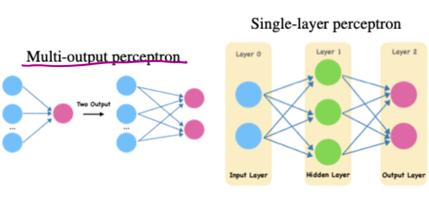
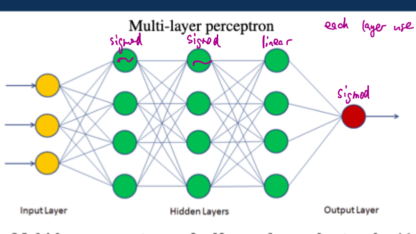


Image source: https://yogayu.github.io/DeepLearningCourse/03/MultilayerPerceptron.html

Neural network construction



- **Multi-layer perceptron**: a **feedforward neural network** with multiple **fully-connected layers** and (at least some nonlinear activation function)

Image source: https://machinelearninggeek.com/multi-layer-perceptron-neural-network-using-python/

Fitting feedforward neural networks

- Empirical loss: total cost incurred from incorrect predictions (how well your network performs)

match =0
mismatch = loss

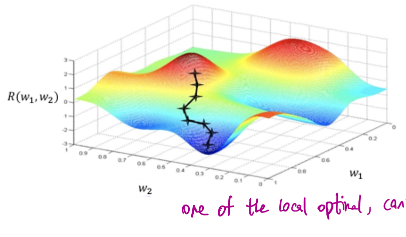
$$R(\boldsymbol{w}) = \frac{1}{n}\sum_{i=1}^{n} L(y_i, \hat{y}_i) = \frac{1}{n}\sum_{i=1}^{n} L(y_i, f(\boldsymbol{x}_i, \boldsymbol{w}))$$

- Objective:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} R(\boldsymbol{w})$$

- Gradient descent:
  1. start with a random guess at the weights
  2. compute gradient, $\frac{\partial R(\boldsymbol{w})}{\partial \boldsymbol{w}}$
  3. update weights, $\boldsymbol{w} \rightarrow \boldsymbol{w} - \underset{\text{learning rate}}{\eta} \frac{\partial R(\boldsymbol{w})}{\partial \boldsymbol{w}}$
  4. repeat steps 2-3 until reaching stopping criteria

Fitting feedforward neural networks: gradient descent



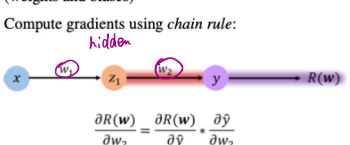
one of the local optimal, can't ensure the best optimal

Image source: MIT 6.S191. Introduction to Deep Learning

Fitting (feedforward) neural networks: backpropagation

- Feed loss backwards to tune the model parameters (weights and biases)
- Compute gradients using *chain rule*:

hidden

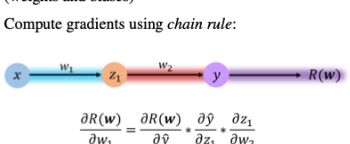


$$\frac{\partial R(\boldsymbol{w})}{\partial w_2} = \frac{\partial R(\boldsymbol{w})}{\partial \hat{y}} * \frac{\partial \hat{y}}{\partial w_2}$$

Image source: MIT 6.S191. Introduction to Deep Learning

Fitting (feedforward) neural networks: backpropagation

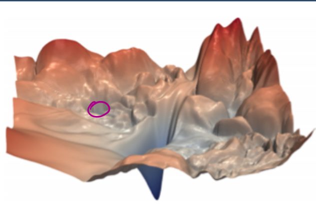
- Feed loss backwards to tune the model parameters (weights and biases)
- Compute gradients using *chain rule*:



$$\frac{\partial R(\boldsymbol{w})}{\partial w_1} = \frac{\partial R(\boldsymbol{w})}{\partial \hat{y}} * \frac{\partial \hat{y}}{\partial z_1} * \frac{\partial z_1}{\partial w_2}$$

Image source: MIT 6.S191. Introduction to Deep Learning

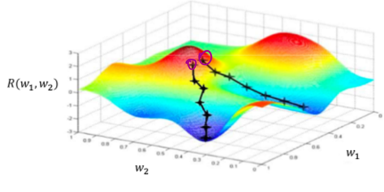
Training in practice



- Gradient descent algorithm may get "stuck" in local optima.

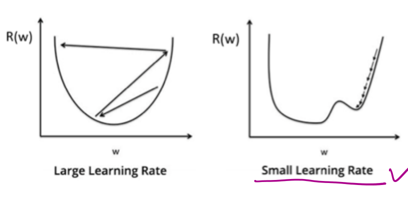
Image source: Li, Hao, et al. "Visualizing the loss landscape of neural nets."

Training in practice: random initialisation



- Different starting values may lead to different local optima.
- Common practice 1: test multiple starting values
- Common practice 2: Initialise weights to random values near zero; initialise bias to 0

Training in practice: learning rate



- Advanced techniques: adaptive learning rate

Image source: https://saugatbhattarai.com.np/what-is-gradient-descent-in-machine-learning/

Training in practice: additional considerations

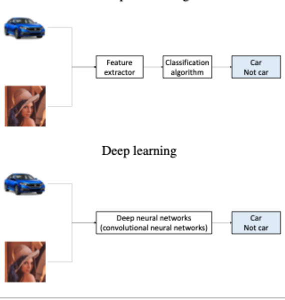
- Stochastic gradient descent dataset be random
- Regularisation avoid overfit

Pros of deep learning

- Nonlinear activation functions
- Universal Approximation Theorem: neural networks with a single hidden layer can be used to approximate any *continuous* function to any desired precision

  Caveat: the layer may be infeasibly large and may fail to learn and generalize correctly
- Particularly suitable for complex problems such as image classification, natural language processing, and speech recognition

Pros of deep learning



Cons of deep learning

- may not be easily interpreted
- typically require a large amount of data to perform well
- can be computationally expensive to train
- training a good neural network can be hard

Next week's plan

Next week (Week 7) will be a reading week.

1. Training neural networks in practice
   - learning rate, optimizers
   - regularisation techniques
2. Deep learning in R
3. Optional topic 1: convolutional neural networks
   Optional topic 2: recurrent neural networks
4. Revision quizzes
5. Additional topics: deep reinforcement learning, etc.

No tutorial; drop-in Q&A on Friday 12-1pm.

Next week's plan

Next week (Week 7) will be a reading week.

1. Training neural networks in practice
   - learning rate, optimizers
   - regularisation techniques
2. Deep learning in R
3. Optional topic 1: convolutional neural networks
   Optional topic 2: recurrent neural networks
4. Revision quizzes
5. Additional topics: deep reinforcement learning, etc.

No tutorial; drop-in Q&A on Friday 12-1pm.

Tutorial in Week 8 will take place on **Wednesday 12-1pm**. The lecture material will be provided early next week.