PS5841

Data Science in Finance & Insurance

FFNN

Yubo Wang

Autumn 2021

Overview

- Approximate $f(\boldsymbol{x})$
- Composite functions

$$\hat{f}(\boldsymbol{x}) = f^{(n)} \circ f^{(n-1)} \circ \cdots \circ f^{(1)}(\boldsymbol{x})$$

- Extending the linear model

$$= \boldsymbol{\phi}(\boldsymbol{x}; \boldsymbol{\theta})^T \boldsymbol{w} + b$$

- At layer j, typically
  - An activation function is applied, element-wise, to an affine transformation of inputs

$$f_i^{(j)} = g(x^T W_{:,i} + c_i)$$

Network

$$\hat{f}(\boldsymbol{x}) = f^{(n)} \circ \underbrace{f^{(n-1)} \circ \cdots \circ f^{(1)}}_{\text{hidden layer}}(\boldsymbol{x})$$

output layer

input layer

depth

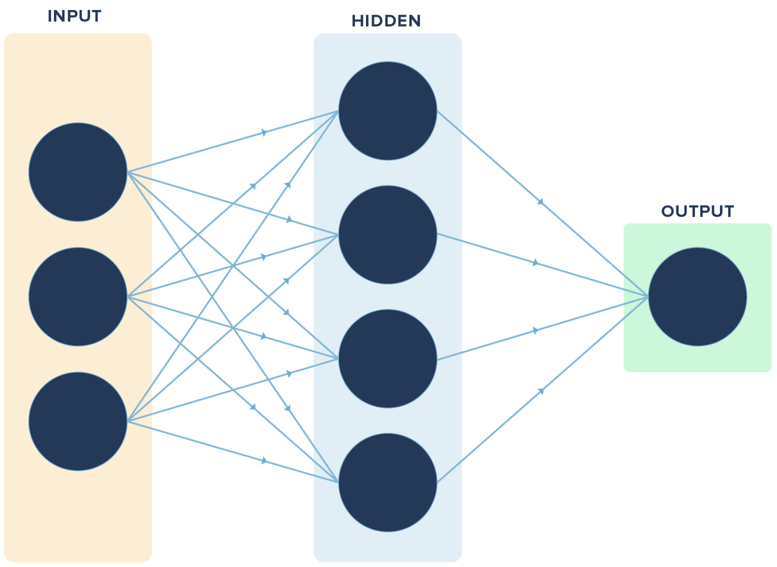
$$\hat{f}(\boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{x};\boldsymbol{\theta})^T\boldsymbol{w} + b$$

bias

H=hidden layer:
nonlinear transformation of x

Width = # of
neurons

Network



width of a layer:
# of neurons

depth

- At layer j, typically
  - An activation function is applied, element-wise, to an affine transformation of inputs

$$f_i^{(j)} = g(x^T W_{:,i} + c_i)$$

Activation Functions

- Linear

$$g(z) = z$$

- Relu (rectified linear unit)

$$g(z) = \max(z, 0)$$

- Sigmoid

$$g(z) = \frac{1}{1 + e^{-z}}$$

- Softmax

$$g(\boldsymbol{z})_i = \frac{e^{z_i}}{\sum_j \ e^{z_j}}$$

- Hyperbolic tangent

$$g(z) = \tanh(z)$$

- …

Loss Functions

- Mean Squared Error

$$R_i = (y_i - \hat{y}_i)^2$$

- Binary Cross Entropy
- Categorical Cross Entropy
- …

Optimizers

- SGD (stochastic gradient descent)
- …

That was