Neural Networks

IKHAN CHOI

Contents

The AI paradigm changes when a new approximating method is discovered.

1. Motivations

1.1. **Statistical model.** The following is not mathematically sound.

Definition 1.1. A *statistical model* is an approximation scheme for unknown probability distribution.

In particular, the general purpose of many statistical models is to estimate the joint probability distribution of two random variables X and Y. The joint probability distrbution contains data about relation between X and Y. Suppose our goal is to obtain the most possible value of Y when given $X = x$, and we have estimated the joint distribution function $f_{X,Y}$. Then, the function $y \mapsto f_{X,Y}(x, y)$ describes the distribution of $Y|X = x$, so what we wanted is reasonably defined as

$$\widehat{y} = \arg\max_y f_{X,Y}(x, y).$$

1.2. **Statistical mechanics.**

Last Update: July 19, 2019.

2. Models

2.1. Random fields.

Definition 2.1 (Random field)**.** A *random field* is a set of random variables parametrized by a topological space or a (directed or undirected)graph.

Definition 2.2. In this note, we will call the random fields on a graph as *networks.*

Actually, networks and graphs are often used as synonyms.

Example 2.1 (Markov chain)**.** Let $G = (V, E)$ be a directed graph defined by

$$V = \mathbb{Z}_{\geq 0}, \quad E = \{(t, t+1)\}_{t \in V},$$

that is, an element $t \in V$ denotes the time t.

Let $\mathcal{S}$ be a finite set of states such that every subset is measurable. Then, the set of $\mathcal{S}$-valued random variables $\{X_t\}_{t \in V}$ indexed by V defines a random field. The Markov property is given by

$$X_t \perp\!\!\!\perp X_s \mid X_{t-1}$$

for $s \leq t$. Since $\mathcal{S}$ is finite, alternatively we may rewrite it by

$$\Pr(X_t = x_t \mid X_{t-1} = x_{t-1}) = \Pr(X_t = x_t \mid X_{t-1} = x_{t-1}, \cdots, X_0 = x_0).$$

Example 2.2 (Maxwell-Boltzman distribution)**.** Let $G = (V, E)$ be a graph defined by

$$V = \{n\}_{n=1}^N, \quad E = \varnothing$$

for a large natural number N; the set V is considered to be the set of ideal gas particles. Define the space of microstates $\mathcal{S} = \mathbb{Z}_x^3 \times \mathbb{Z}_p^3$, which embodies the discretized phase space. At each particle $n \in V$, an $\mathcal{S}$-valued random variable denoted by X_n is attached.

Let $m > 0$ be a constant. Define the *Boltzmann factor* as a function $\phi : \mathbb{R}_{>0} \times V \to \mathbb{R}$ such that

$$\phi(\beta, i) := e^{-\beta E_i},$$

where $\beta = \frac{1}{k_B T} > 0$ is called *coldness* and the *energy function* $E : \mathcal{S} \to \mathbb{R}$ is defined by

$$E(x, p) := \frac{\|p\|^2}{2m}.$$

The assumption for Boltzmann factors states that given β, the probability for a particle to be in the state i is proportional to the Boltzmann factor:

$$\Pr(X_n = i) \propto e^{-\beta E_i}$$

for each state $i \in \mathcal{S}$ and particle $n \in V$. Thus, we can write

$$\Pr(X_n = i) = \frac{e^{-\beta E_i}}{\sum_{j=1}^N e^{-\beta E_j}} =: \frac{1}{Z(\beta)}\, e^{-\frac{\beta}{2m}\|p_i\|^2}.$$

The denominator Z is called the *partition function.* Note that it depends on the data of the Boltzmann factor.

2.2. **Bayesian networks.**

Definition 2.3 (Bayesian network)**.** Let G be a directed acyclic graph.

The graph acts as a parameter space. We want to investigate mutual effects among the paramtrized random variables.

Theorem 2.3 (Factorization of probablity)**.**

Example 2.4 (NBC, Naive Bayesian Classifier)**.**

Example 2.5 (HMM, Hidden Markov Model)**.**

2.3. **Markov networks.**

Definition 2.4 (Markov network)**.**

Markov networks are sometimes called MRF, Markov random field.

Example 2.6 (CRF, Conditional Random Field)**.** Consider a network with a graph G such that vertices are divided into two classes.

2.4. **Neural networks.** Probabilistic graphical models provide effective explanations of the neural networks, but neural networks are not confined only to graphical models.

Definition 2.5 (Neural network)**.** *Neural network* cannot be defined mathematically. It indicates statistical models that can solve problems with a collection of artificial neurons by adjusting connection strength among them.

Example 2.7 (MLP, Multi-layer Perceptron)**.**

Example 2.8 (RNN, Recurrent Neural Network)**.**

3. Inference

3.1. **Viterbi algorithm.**

4. Learning

4.1. **Gradient descent method.**

4.2. **Back propagation.** Backpropagation refers to algorithms to train the weight matrices for minimizing the cost function J, which does not depend explicitly on any variables except the last layer vector $a^{(n)}$. However, since J is a function of the weight matrices implicitly, via $a^{(n)}$, we may find the representation of the gradiant of J as viewing it as a function on the space of weight matrices of each given layer. In other words, we want to find the coefficients of the differential form dJ on the basis $\{dW_{ij}^{(n-1)}\}_{i,j}$, $\{dW_{jk}^{(n-2)}\}_{j,k}$, or $\{dW_{kl}^{(n-3)}\}_{k,l}$, and so on.

Recall the definitions:

$$a_i^{(n)} = \sigma\left(\sum_j W_{ij}^{(n-1)} a_j^{(n-1)}\right).$$

Since the derivative of the sigmoid function is given by $\sigma' = \sigma - \sigma^2$, we can compute the following auxiliary relations

$$\frac{\partial a_i^{(n)}}{\partial a_j^{(n-1)}} = h(a_i^{(n)})W_{ij}^{(n-1)} \quad \text{and} \quad \frac{\partial a_i^{(n)}}{\partial W_{i'j}^{(n-1)}} = \delta_{ii'} h(a_i^{(n)})a_j^{(n-1)},$$

where $h(x) = x - x^2$.

Then, we can compute

$$dJ = \sum_i \frac{\partial J}{\partial a_i^{(n)}} \sum_j \frac{\partial a_i^{(n)}}{\partial W_{ij}^{(n-1)}} \, dW_{ij}^{(n-1)} = \sum_{i,j} \frac{\partial J}{\partial a_i^{(n)}} h(a_i^{(n)})a_j^{(n-1)} \, dW_{ij}^{(n-1)},$$

which implies

$$\nabla J(W^{(n-1)}) = \left[\frac{\partial J}{\partial a_i^{(n)}} h(a_i^{(n)})a_j^{(n-1)}\right] \frac{\partial}{\partial W_{ij}^{(n-1)}}.$$

Note that it is a function of a_i and a_j. The gradient descent method will take

$$W_{ij}^{(n-1)+} := W_{ij}^{(n-1)} - \alpha \cdot \frac{\partial J}{\partial a_i^{(n)}} h(a_i^{(n)})a_j^{(n-1)}$$

with a proper parameter $\alpha > 0$.

By the same reason,

$$\begin{aligned} dJ &= \sum_{i,j,k} \frac{\partial J}{\partial a_i^{(n)}} \frac{\partial a_i^{(n)}}{\partial a_j^{(n-1)}} \frac{\partial a_j^{(n-1)}}{\partial W_{jk}^{(n-2)}} \, dW_{jk}^{(n-2)} \\ &= \sum_{i,j,k} \frac{\partial J}{\partial a_i^{(n)}} \cdot h(a_i^{(n)})W_{ij}^{(n-1)} \cdot h(a_j^{(n-1)})a_k^{(n-2)} \, dW_{jk}^{(n-2)}, \end{aligned}$$

which implies

$$\nabla J(W^{(n-2)}) = \left[\sum_i \frac{\partial J}{\partial a_i^{(n)}} \cdot h(a_i^{(n)})W_{ij}^{(n-1)} \cdot h(a_j^{(n-1)})a_k^{(n-2)}\right] \frac{\partial}{\partial W_{jk}^{(n-2)}}.$$

Therefore, the gradient descent method will take

$$\begin{aligned} W_{jk}^{(n-2)+} &: = W_{jk}^{(n-2)} - \alpha \cdot \sum_i \frac{\partial J}{\partial a_i^{(n)}} h(a_i^{(n)})W_{ij}^{(n-1)} h(a_j^{(n-1)})a_k^{(n-2)} \\ &= W_{jk}^{(n-2)} + (1 - a_j^{(n-1)})a_k^{(n-2)} \sum_i (W_{ij}^{(n-1)+} - W_{ij}^{(n-1)})W_{ij}^{(n-1)}. \end{aligned}$$

In similar way,

$$W_{kl}^{(n-3)+} := W_{kl}^{(n-3)} + (1 - a_k^{(n-2)})a_l^{(n-3)} \sum_i (W_{jk}^{(n-2)+} - W_{jk}^{(n-2)})W_{jk}^{(n-2)} (?)$$

4.3. Maximum likelihood estimate.

Definition 4.1. Let f be a distribution function on a measure space X. Let $\{f_\theta\}_\theta$ be a parametrized family of distrubution functions on X. The *likelihood* $L_n(\theta) : \Omega^n \to \mathbb{R}_{\geq 0}$ for a fixed parameter θ is a random variable defined by

$$L_n(\theta) := \prod_{i=1}^{n} f_\theta(x_i)$$

where $\{x_i\}_i$ is a family of i.i.d. X-valued random variables with a distriburion f.

The objective of the likelihood function is to find θ such that f_θ approximates the unknown distribution f. Write

$$\frac{1}{n} \log L_n(\theta) = \frac{1}{n} \sum_{i=1}^{n} \log f_\theta(x_i).$$

By the law of large numbers, $\frac{1}{n} \log L_n(\theta)$ converges to a constant function

$$\mathbb{E}(\log f_\theta(x)) = \int_X f \log f_\theta$$

in measure as $n \to \infty$. This constant function is exactly what we call *cross entropy.*

The *Kullback-Leibler divergence* is a kind of asymmetric distance function defined from the difference with cross entropy

$$D_{KL}(f \| f_\theta) := \int_X f \log f - \int_X f \log f_\theta.$$

It is proved to be always nonnegative by the Jensen inequality:

$$\int_X f \log f_\theta - \int_X f \log f = \int_X f \log \frac{f_\theta}{f} \leq \log \left(\int_X f \frac{f_\theta}{f} \right) = 0.$$

Here, we exclude the region $f = 0$ from the integration region. Then, we can say, bigger $L_n(\theta)$ is, closer f_θ and f are.