DYNAMIC SYSTEM AND DEEP LEARNING

PINGCHUAN MA

Dynamic Systems

Definition.
A deterministic discrete dynamic system $F : X \to X$ is the action of a continuous map F on a metric space (X, d), usually a subset of $\mathbb{R}^n$.

Theorem.
If f: $\mathbb{R} \to \mathbb{R}$ has continuous derivative $f^{'}$, then a fixed point x_0 is **attracting** if $|f^{'}(x_0)| < 1$. If $|f^{'}(x_0)| > 1$, then x_0 is **repelling**. In both cases we say x_0 is **hyperbolic**.

Hopfield Networks

Definition.
At each point in time, update one node chosen randomly or according to some rule is called asynchronously update while all nodes together are updated together is called synchronously update.

Definition.
Energy Function: For a given state $x \in \{-1, 1\}^N$ of the network and for any set of connection weights $w_{ij} = w_{ji}$ and $w_{ii} = 0$, let

$$E = -\frac{1}{2} \sum_{i,j=1}^{N} w_{ij} x_i x_j$$

Definition.
Central Limit Theorem: if z_m is a sequence of i.i.d. random variables each with mean μ and variance σ^2 then for large n

$$X_n = \frac{1}{n} \sum_{m=1}^{n} z_m$$

has approximately a normal distribution with mean $\bar{X}_n = \mu$ and variance $X_n^2 - \bar{X}_n^2 = \sigma^2/n$

Markov Chains and MCMC

Definition.
A Markov chain Y_0, Y_1,... is a sequence of random variables, with $Y_t \in S$ for all points in time $t \in$ N, that satisfies the **Markov property**, namely, given the present state and past states are independent:

$$Pr(Y_{n+1} = x | Y_0 = y_0, ..., Y_n = y_n) = Pr(Y_{n+1} | Y_n = y_n)$$

Theorem.
A Markov chain is **homogeneous** if for all $n \leq 1$:

$$Pr(Y_{n+1} = j | Y_n = i) = Pr(Y_n = j | Y_{n-1} = i)$$

Theorem.
Fundamental Theorem of Markov chains: An irreducible and aperiodic Markov chain has a unique stationary distribution π which satisfies:

- $lim_{n\to\infty} x P^n = \pi$ for all $x \in X$
- $lim_{n\to\infty} P^n$ exists and is the matrix with all rows equal to π.

Definition.
A Markov chain is **reversible** if there is $\pi \in X$ that satisfies the **detailed balanced condition:**

$$\pi_i P_{ij} = \pi_j P_{ji}, \qquad 1 \leq i, j \leq N$$

Date: April 2017.

Definition.
Gibbs sampling in a Markov random field with a graph $G = (V, E)$ updates each variable based on its conditional distribution given the state of the other variables.

Restricted Boltzmann Machines

Definition.
A Boltzmann Machine has binary (0 or 1) visible vector unit x and hidden (latent) vector unit h that detects features in the visible vector x.

Definition.
The energy by Hammersley-Clifford theorem, is presented:

$$E(v,h) = -\sum_{i=1}^{n}\sum_{j=1}^{m} w_{ij}h_i v_j - \sum_{j=1}^{m} b_j v_j - \sum_{i=1}^{n} c_i h_i$$

where $1 \le j \le m, 1 \le i \le n$.

Definition.
An asymmetric measure of difference between p and q is given by **Kullback-Leiber divergence** or the **relative entropy** of q wrt p given for a finite state space S by:

$$KL(q||p) = \sum_{x\in S} q(x) ln \frac{q(x)}{p(x)} = \sum_{x\in S} q(x) lnq(x) - \sum_{x\in S} q(x) lnp(x)$$

Definition.
For RBM, start from an initial value $\theta^{(0)}$. Let

$$\theta^{(t+1)} = \theta^{(t)} + \frac{\alpha}{l}\frac{\partial}{\partial\theta}(\sum_{k=1}^{l} lnp(x_k|\theta^{(t)})) - \lambda\theta^{(t)} + v\Delta\theta^{(t-1)}$$

where $\Delta\theta^{(t)} = \theta^{(t+1)} - \theta^{(t)}$ and $\alpha > 0$ is the learning rate. Two terms in the above equation aims to optimise the algorithm: (1)$-\lambda\theta^{(t)}$ is the decay weight, with $\lambda > 0$ a constant. (2) $v\Delta\theta^{(t-1)}$ is the momentum, with $v > 0$ a constant.

Definition.
The **softmax** function takes a vector in $z \in \mathrm{R}^L$ of L real numbers and provides a probability vector with L components:

$$\frac{e^{z_k}}{\sum_{l=1}^{L} e^{z_l}}$$

Definition.

$$p(H_i = h_i|v) = \frac{e^{\sum_{j=1}^{m} w_{ij}v_j h_i + c_i h_i}}{1 + e^{\sum_{j=1}^{m} w_{ij}v_j + c_i}}$$
$$p(V_j = v_j|h) = \frac{e^{\sum_{i=1}^{n} w_{ij}v_j h_i + b_j v_j}}{1 + e^{\sum_{i=1}^{n} w_{ij}h_i + b_j}}$$

Theorem.

$$\begin{aligned}
&\frac{1}{l}\sum_{v\in D}\frac{\partial lnp(v|w_{ij})}{\partial w_{ij}}\\
&= \frac{1}{l}\sum_{v\in D}[-\sum_{h} p(h|v)\frac{\partial E(v,h)}{\partial w_{ij}} + \sum_{v,h} p(v,h)\frac{\partial E(v,h)}{\partial w_{ij}}]\\
&= \frac{1}{l}\sum_{v\in D}[\sum_{h} p(h|v)h_i v_j - \sum_{h} p(v,h)h_i v_j]\\
&= \frac{1}{l}\sum_{v\in D}[\mathbb{E}_{p(h|v)}(h_i v_j) - \mathbb{E}_{p(v,h)}(h_i v_j)]\\
&=< h_i v_j >_{p(h|v)q(v)} - < h_i v_j >_{p(v,h)}\\
&=< h_i v_j >_{data} - < h_i v_j >_{model}
\end{aligned}$$

Deep Belief Networks

Definition.
A Deep Belief Networks (DBN) is a multi-layered generative model that mixes undirected and directed connections between the units. For a DBN with J layers of hidden units, the network contains undirected matrix connection $W^{(J)}$ between the top two hidden unit layers $h^{(J)}$ and $h^{(J-1)}$, forming an RBM. There are directed connections $W^{(j)}$ between the remaining hidden unit layers $h^{(j)}$ and $h^{(j-1)}$, for $1 \leq j \leq J$.

Backpropagation

Definition.
The backpropagation algorithm is given as follows.

- Input: Insert $x \in M$ as the activation a^1 of the first level
- Feedforward: For each $2 \leq l \leq L$, compute a^l, using $a^l = \sigma(z^l)$ and $z^l = W^l a^{l-1} + b^l$
- Output error: Compute $\delta_j^l = \frac{\partial C_x}{\partial a_j^L} \sigma`(z_j^L)$
- Error propagation: For $1 \leq l \leq L-1$, compute $\delta_j^l = ((W^{l+1})\delta^{l+1})_j \sigma`(z_j^l)$
- Output: $\frac{\partial C_x}{\partial w_{jk}^l} = \delta_j^l a_k^{l-1}$ and $\frac{\partial C_x}{\partial b_j^l} = \delta_j^l$
- Repeat for all $x \in M$
- Update w_{jk}^l and b_j^l after each mini-batch is done:

$$w_{jk}^l \leftarrow w_{jk}^l - \frac{\eta}{m} \sum_{x \in M} \frac{\partial C_x}{\partial w_{jk}^l}$$

$$b_j^l \leftarrow b_j^l - \frac{\eta}{m} \sum_{x \in M} \frac{\partial C_x}{\partial b_j^l}$$

Definition.
The **cross entropy cost function** for backpropagation is defined as $C = \frac{1}{|T|} \sum_{x \in T} C_x$ with

$$C_x = -\sum_j [y_j(x) log a_j^L(x) + (1 - y_j(x)) log(1 - a_j^L(x))]$$

Convolutional Neural Nets

Requires four hyperparameters:

- number of filters K
- their spatial extent F
- the stride S
- the amount of zero padding P

Produce a volume of size $W_2 \times H_2 \times D_2$

- $W_2 = (W_1 - F + 2P)/S + 1$
- $H_2 = (H_1 - F + 2P)/S + 1$
- $D_2 = K$

With parameters sharing, it introduces $F \cdot F \cdot D_1$ weights per filter, for a total of $(F \cdot F \cdot D_1) \cdot K$ weights and K biases

Definition.
The field z_{ij} into hidden neuron (i, j) is given by

$$z_{ij} = \sum_{p,q=0}^{F-1} w_{pq} a_{(i+p)(j+q)} + b$$

where the result is called the **convolved feature**.

Definition.
Stride is defined as the number of steps that filters slide across their inputs.

Definition.
Local Connection, Weight Sharing, Pooling

Definition.
Pooling, subsampling or downsampling, reduces the dimensions of the feature maps, and the number of parameters while keeping significant information.

Theorem.
Three significant features are included in convolutional neural networks.

- Local Connectivity: When dealing with high-dimensional inputs such as images, as we saw above it is impractical to connect neurons to all neurons in the previous volume. Instead, we will connect each neuron to only a local region of the input volume.
- Parameter Sharing: It is used to control the number of parameters. It turns out that we can dramatically reduce the number of parameters by making one reasonable assumption: That if one feature is useful to compute at some spatial position (x,y), then it should also be useful to compute at a different position (x2,y2).
- Downsampling: use max to pool

Theorem.
The firing rate of the neuron is modelled with an activation function f. The most commonly used is ReLU, in which $f(x) = \max(0, x)$.

ReLU is found to greatly accelerate the convergence of stochastic gradient descent compared to the sigmoid/tanh functions due to its linear. In addtion to this, it does not involve expensive operations. The disadvantage is that ReLU neuron can die during training and the unit will forever be zero from that point on.