Boltzmann Machine

Deep Computational model

1. Deep neural network
2. Probabilistic graphical model

Boltzmann machine is a fully-connected undirected graphical model.

For a restricted Boltzmann Machine (RBM), it consists of weights, biases, and nodes.

Energy function

For two nodes v_i, h_j in visible layer and hidden layer respectively.

$$E(v_i, h_j) = -a_i v_i - b_j h_j - v_i w_{ij} h_j$$

For all the nodes,

$$E(v, h) = -\sum_i a_i v_i - \sum_j b_j h_j - \sum_i \sum_j v_i w_{ij} h_j$$

Map energy function to potential function

$$\psi_Q(X_Q) = e^{-E(X_Q)}$$

For any clique,

$$\psi_Q(v_i, h_j) = e^{a_i v_i + b_j h_j + v_i w_{ij} h_j}$$

From potential function to probability distribution

For any clique in Set C

$$p(v, h) = \frac{1}{z} \prod_{Q \in C} \psi_Q(X_Q)$$

$$z = \sum_{v \in \{0,1\}} \sum_{h \in \{0,1\}} \left(\prod_{Q = (v,h) \in C} \psi_Q(X_Q) \right)$$

$$\psi_Q(X_Q) = \prod_{Q = \{v_i, h_j\}}^{n} e^{a_i v_i + b_j h_j + v_i w_{ij} h_j} = \prod_{i=1}^{n} \prod_{j=1}^{m} e^{a_i v_i + b_j h_j + v_i w_{ij} h_j}$$

$$= e^{\sum_i a_i v_i + \sum_j b_j h_j + \sum_i \sum_j v_i w_{ij} h_j} = e^{-E(v,h)}$$

$$p(v, h) = \frac{1}{z} \prod_{Q \in C} \psi_Q(X_Q) = \frac{e^{-E(v,h)}}{z}$$

Inference

Set

$$h = (h_1, h_2, \ldots, h_m)$$
$$v = (v_1, v_2, \ldots, v_n)$$

Now we derive the formula for p(v)

$$p(v) = \sum_h p(v,h) = \sum_{h\in\{0,1\}^m} \frac{e^{-E(v,h)}}{z}$$

$$\sum_{h\in\{0,1\}^m} e^{-E(v,h)} = \sum_{h\in\{0,1\}^m} e^{\sum_i a_i v_i + \sum_j b_j h_j + \sum_i \sum_j v_i w_{ij} h_j} =$$

$$= e^{a^T v} \sum_{h\in\{0,1\}^m} e^{\sum_j b_j h_j + \sum_i \sum_j v_i w_{ij} h_j}$$

$$= e^{a^T v} \left(\sum_{h_1\in\{0,1\}} e^{b_1 h_1 + \sum_i v_i w_{ij} h_1} \right) \cdots \left(\sum_{h_m\in\{0,1\}} e^{b_m h_m + \sum_i v_i w_{ij} h_m} \right)$$

Take a close look at one term

$$\sum_{h_j\in\{0,1\}} e^{b_j h_j + \sum_i v_i w_{ij} h_j} = 1 + e^{b_j + v^T w_{*j}}$$

Hence,

$$e^{a^T v} \left(\sum_{h_1\in\{0,1\}} e^{b_1 h_1 + \sum_i v_i w_{ij} h_1} \right) \cdots \left(\sum_{h_m\in\{0,1\}} e^{b_m h_m + \sum_i v_i w_{ij} h_m} \right)$$

$$= e^{a^T v} \prod_{j=1}^{m} \left(1 + e^{b_j + v^T w_{*j}} \right)$$

$$= e^{a^T v + \sum_{j=1}^{m} In\left(1 + e^{b_j + v^T w_{*j}}\right)}$$

Hence, we get p(v)

$$p(v) = \frac{e^{a^T v + \sum_{j=1}^{m} In\left(1 + e^{b_j + v^T w_{*j}}\right)}}{z} = \frac{e^{-F(v)}}{z}$$

And similarly, we get

$$p(h) = \frac{e^{b^T h + \sum_{i=1}^{n} In\left(1 + e^{a_i + h^T w_{i*}}\right)}}{z}$$

The conditional probability p(h|v)

According to Markov Independence,

$$p(h \mid v) = \frac{p(h,v)}{p(v)} = \prod_{j=1}^{m} p(h_j \mid v)$$

$$= \frac{e^{\sum_i a_i v_i + \sum_j b_j h_j + \sum_i \sum_j v_i w_{ij} h_j} / z}{e^{a^T v + \sum_{j=1}^{m} In\left(1 + e^{b_j + v^T w_{*j}}\right)} / z} = \frac{e^{\sum_i a_i v_i + \sum_j b_j h_j + \sum_i \sum_j v_i w_{ij} h_j}}{e^{a^T v + \sum_{j=1}^{m} In\left(1 + e^{b_j + v^T w_{*j}}\right)}}$$

$$= \frac{e^{\sum_j b_j h_j + \sum_i \sum_j v_i w_{ij} h_j}}{e^{\sum_{j=1}^{m} In\left(1+e^{b_j + v^T w_{*j}}\right)}} = \frac{e^{\sum_j b_j h_j + \sum_i \sum_j v_i w_{ij} h_j}}{\prod_{j=1}^{m}\left(1+e^{b_j + v^T w_{*j}}\right)}$$

$$= \frac{\prod_{j=1}^{m} e^{b_j h_j + v^T w_{*j} h_j}}{\prod_{j=1}^{m}\left(1+e^{b_j + v^T w_{*j}}\right)} = \prod_{j=1}^{m} \frac{e^{b_j h_j + v^T w_{*j} h_j}}{1+e^{b_j + v^T w_{*j}}}$$

$$p(h_j \mid v) = \frac{e^{b_j h_j + v^T w_{*j} h_j}}{1+e^{b_j + v^T w_{*j}}}$$

Hence,

$$p(h_j \mid v) = \begin{cases} \dfrac{1}{1+e^{b_j + v^T w_{*j}}} & h_j = 0 \\ \dfrac{1}{1+e^{-b_j - v^T w_{*j}}} & h_j = 1 \end{cases}$$

Similarly,

$$p(v_i \mid h) = \begin{cases} \dfrac{1}{1+e^{a_j + w_{*j} h}} & v_j = 0 \\ \dfrac{1}{1+e^{-b_j - w_{i*} h}} & v_j = 1 \end{cases}$$

Cost function

$$\max_\theta \left(\prod_{i=1}^{s} p(v^i)\right) = \min_\theta - In\left(\prod_{i=1}^{s} p(v^i)\right)$$

$$= \min_\theta - \sum_{i}^{s} In\left(p(v^i)\right)$$

Gradient

$$In(p(v)) = -F(v) - In(z)$$

$$\frac{\partial In(p(v))}{\partial \theta} = \frac{\partial(-F(v))}{\partial \theta} - \frac{\partial(In(z))}{\partial \theta}$$

$$\frac{\partial(-F(v))}{\partial \theta} = \frac{\partial\left(-a^T v - \sum_{j=1}^{m} In\left(1+e^{b_j + v^T w_{*j}}\right)\right)}{\partial \theta}$$

$$\frac{\partial(-F(v))}{\partial\theta}=\begin{cases}\dfrac{e^{b_j+v^T w_{*j}}}{1+e^{b_j+v^T w_{*j}}}v_i & \theta=w_{ij}\\ v_i & \theta=a_i\\ \dfrac{e^{b_j+v^T w_{*j}}}{1+e^{b_j+v^T w_{*j}}} & \theta=b_j\end{cases}$$

$$\frac{\partial(In(z))}{\partial\theta}=\frac{\partial\left(In\left(\sum_v\sum_h e^{-E(v,h)}\right)\right)}{\partial\theta}$$

$$=\frac{\sum_v\sum_h\left(e^{-E(v,h)}\frac{\partial(-E(v,h))}{\partial\theta}\right)}{\sum_v\sum_h e^{-E(v,h)}}$$

$$=\sum_v\sum_h\left(\frac{e^{-E(v,h)}}{\sum_v\sum_h e^{-E(v,h)}}\frac{\partial(-E(v,h))}{\partial\theta}\right)$$

$$=\sum_v\sum_h\left(p(v,h)\frac{\partial(-E(v,h))}{\partial\theta}\right)$$

$$=\sum_v p(v)\left(\sum_h\left(p(h\mid v)\frac{\partial(-E(v,h))}{\partial\theta}\right)\right)$$

$$=\sum_v p(v)\left(\frac{\sum_h\left(e^{-E(v,h)}\frac{\partial(-E(v,h))}{\partial\theta}\right)}{\sum_h e^{-E(v,h)}}\right)$$

$$=\sum_v p(v)\frac{\partial(-F(v))}{\partial\theta}$$

$$\frac{\partial(In(z))}{\partial\theta}=\begin{cases}\sum_v p(v)\dfrac{e^{b_j+v^T w_{*j}}}{1+e^{b_j+v^T w_{*j}}}v_i & \theta=w_{ij}\\ \sum_v p(v)v_i & \theta=a_i\\ \sum_v p(v)\dfrac{e^{b_j+v^T w_{*j}}}{1+e^{b_j+v^T w_{*j}}} & \theta=b_j\end{cases}$$

Hence, the gradient

$$\frac{\partial In(p(v))}{\partial\theta}=\begin{cases}p(h_j=1\mid v)v_i-\sum_v p(v)p(h_j=1\mid v)v_i & \theta=w_{ij}\\ v_i-\sum_v p(v)v_i & \theta=a_i\\ p(h_j=1\mid v)-\sum_v p(v)p(h_j=1\mid v) & \theta=b_j\end{cases}$$

Sampling

$$h^{t+1} \sim p\left(h^{t+1} \mid v^{t}\right) = sigmoid\left(Wv^{t} + b\right)$$
$$v^{t+1} \sim p\left(v^{t+1} \mid h^{t+1}\right) = sigmoid\left(Wh^{t+1} + a\right)$$

Contrast Divergence

Input: training data D, iteration_steps, sample_steps cd_k

Output: update weights, biases

1. Initialize *W, a, b*
2. for i = 1, 2,..., n_steps

    v0 = v

    for k = 0, 2, ..., *cd_k*

    $$h^{t+1} \sim p\left(h^{t+1} \mid v^{t}\right) = sigmoid\left(Wv^{t} + b\right)$$
    $$v^{t+1} \sim p\left(v^{t+1} \mid h^{t+1}\right) = sigmoid\left(Wh^{t+1} + a\right)$$

    for i = 1, 2,..., n, j = 1, 2,..., m

    $$w_{ij} \leftarrow w_{ij} + \left(p\left(h_j = 1 \mid v\right)v_i - p\left(h_j = 1 \mid v^{cd\_k}\right)v_i^{cd\_k}\right)$$

    for j = 1, 2,..., n

    $$a_i \leftarrow a_i + \left(v_i - v_i^{cd\_k}\right)$$

    for j = 1,2,...,m

    $$b_j \leftarrow b_j + \left(p\left(h_j = 1 \mid v\right) - p\left(h_j = 1 \mid v^{cd\_k}\right)\right)$$

Optimization

Gradient Descent

Consider a multivariate function

$$f\left(x_1, x_2, \dots, x_n\right)$$

The gradient at any given point will be

$$\nabla = \begin{pmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots & \frac{\partial f}{\partial x_n} \end{pmatrix}^T$$

There are three gradient descent: Batch Gradient Descent (BGD), Stochastic Gradient Descent (SGD), MiniBatch Gradient Descent (MBGD).

Algorithm	Advantage	Disadvantage
BGD	Consider all the errors	Slow convergence due to large dataset Can not update online Costly
SGD	Fast convergence Update online	Data Redundancy Loss function fluctuates
MBGD	Suitable for matrix computation (GPU) Steady results Update online	Limited number of samples n (10 ~ 500)

Update Strategy

Vanilla strategy

$$\theta = \theta - lr \times d\theta$$

Momentum strategy

$$v_i = mu \times v_{i-1} + lr \times d\left(\theta^{i-1}\right)$$
$$\theta^i = \theta^{i-1} - v_i$$

Nesterov Accelerated Gradient

$$\left(\theta^{i-1}\right)' = \theta^{i-1} - mu \times v_{i-1}$$
$$v_i = mu \times v_{i-1} + lr \times d\left(\left(\theta^{i-1}\right)'\right)$$
$$\theta^i = \theta^{i-1} - v_i$$

Derivation

$$\left(\theta^{i-1}\right)' = \theta^{i-1} - mu \times v_{i-1}$$
$$v_i = mu \times v_{i-1} + lr \times d\left(\left(\theta^{i-1}\right)'\right)$$
$$\theta^i - mu \times v_i = \theta^{i-1} - v_i - mu \times v_i$$
$$= \theta^{i-1} - (1 + mu) \times \left(mu \times v_{i-1} + lr \times d\left(\theta^{i-1} - mu \times v_{i-1}\right)\right)$$
$$= \theta^{i-1} - mu(1 + mu) \times v_{i-1} - lr(1 + mu)d\left(\theta^{i-1} - mu \times v_{i-1}\right)$$

Let

$$\hat{\theta}^i = \theta^i - mu \times v_i$$
$$\hat{\theta}^{i-1} = \theta^{i-1} - mu \times v_{i-1}$$
$$\hat{v}_i = \hat{\theta}^{i-1} - \hat{\theta}^i$$

Hence,

$$\hat{v}_i = lr(1 + mu)d\left(\hat{\theta}^{i-1}\right) + mu^2 \times v_{i-1}$$
$$= lr(1 + mu)d\left(\hat{\theta}^{i-1}\right) + lr \times mu^2 \times d\left(\hat{\theta}^{i-2}\right) + mu^3 v_{i-2}$$
$$= \cdots$$
$$= lr(1 + mu)d\left(\hat{\theta}^{i-1}\right) + lr \times mu^2 \times d\left(\hat{\theta}^{i-2}\right) + \cdots + lr \times mu^k \times d\left(\hat{\theta}^{i-k}\right) + mu^{k+1} v_{i-k}$$
$$\hat{v}_{i-1} = lr(1 + mu)d\left(\hat{\theta}^{i-2}\right) + lr \times mu^2 \times d\left(\hat{\theta}^{i-3}\right) + \cdots + lr \times mu^k \times d\left(\hat{\theta}^{i-k-1}\right) + mu^{k+1} v_{i-k-1}$$
$$mu \times \hat{v}_{i-1} = lr \times mu(1 + mu) \times d\left(\hat{\theta}^{i-2}\right) + lr \times mu^3 \times d\left(\hat{\theta}^{i-3}\right) + \cdots + lr \times mu^{k+1} \times d\left(\hat{\theta}^{i-k-1}\right) + mu^{k+2} v_{i-k-1}$$
$$\hat{v}_i - mu \times \hat{v}_{i-1} = lr(1 + mu)d\left(\hat{\theta}^{i-1}\right) - lr \times mu \times d\left(\hat{\theta}^{i-2}\right)$$
$$= lr \times d\left(\hat{\theta}^{i-1}\right) + lr \times mu \times d\left(\hat{\theta}^{i-1} - \hat{\theta}^{i-2}\right)$$
$$\hat{v}_i = mu \times \hat{v}_{i-1} + lr \times d\left(\hat{\theta}^{i-1}\right) + lr \times mu \times d\left(\hat{\theta}^{i-1} - \hat{\theta}^{i-2}\right)$$

Adaptive gradient algorithm

$$acc_i = acc_{i-1} + \left(d\left(\theta_{i-1}\right)\right)^2$$
$$\theta_i = \theta_{i-1} - \frac{lr}{\sqrt{acc_i + \varepsilon}} \times d\left(\theta_{i-1}\right)$$

Conjugate gradient

Consider a quadratic optimization problem

$$f(x) = \frac{1}{2} x^T Q x + b^T x + c$$

If we can transform the function into

$$f(x) = f_1(x_1) + f_2(x_2) + \ldots + f_n(x_n)$$

If we can find P = {p1, p2,..., pn} that satisfies the following

$$p_i^T Q p_j = 0, i \neq j$$

Then we can separate the function. That is

$$f(Px) = \frac{1}{2}(Px)^T Q(Px) + b^T Px + c$$

$$= \sum_{i=1}^{n} \left(\frac{1}{2} x_i^T p_i^T Q p_i x_i + b^T p_i x_i \right) + c$$

Find the derivative with respect to x

$$\nabla f(x) = Qx + b$$

Let

$$p_1 = -\nabla f(x_1)$$

And

$$(x^2, p_2), (x^3, p_3), \ldots, (x^k, p_k)$$

Let

$$x^{k+1} = x^k + a_k p_k$$

$$a_k = \arg\min\left(f\left(x^k + a_k p_k\right)\right)$$

$$\frac{df\left(x^k + a_k p_k\right)}{da}\Big|_{a=a_k} = p_k^T \nabla f\left(x^{k+1}\right) = 0$$

$$\Leftrightarrow p_k^T \left(Q a_k p_k + \nabla f\left(x^k\right)\right) = 0$$

$$\Leftrightarrow a_k = \frac{-p_k^T \nabla f\left(x^k\right)}{p_k^T Q p_k}$$

$$\nabla f\left(x^{k+1}\right)-\nabla f\left(x^{j+1}\right)=Q\left(x^{k+1}-x^{j+1}\right)$$
$$\Leftrightarrow p_j^T\nabla f\left(x^{k+1}\right)=p_j^T\nabla f\left(x^{j+1}\right)+p_j^TQ\left(x^{k+1}-x^{j+1}\right)$$
$$\Leftrightarrow p_j^T\nabla f\left(x^{k+1}\right)=p_j^TQ\left(x^{k+1}-x^k+x^{k-1}-x^{k-2}+\cdots+x^{j+2}-x^{j+1}\right)$$
$$\Leftrightarrow p_j^T\nabla f\left(x^{k+1}\right)=p_j^TQ\left(\sum_{i=j+1}^{k}a_i p_i\right)$$
$$\Leftrightarrow p_j^T\nabla f\left(x^{k+1}\right)=\left(\sum_{i=j+1}^{k}a_i p_j^TQp_i\right)$$
$$\Leftrightarrow p_j^T\nabla f\left(x^{k+1}\right)=0$$

Let

$$p_{k+1}=-\nabla f\left(x^{k+1}\right)+\lambda_k p_k$$
$$\Leftrightarrow p_k^TQp_{k+1}=p_k^TQ\left(-\nabla f\left(x^{k+1}\right)\right)+p_k^TQ\lambda_k p_k=0$$
$$\Leftrightarrow \lambda_k=\frac{p_k^TQ\left(\nabla f\left(x^{k+1}\right)\right)}{p_k^TQp_k}$$

Conjugate Gradient

$$\min f(x),\frac{1}{2}x^TQx+b^Tx+c$$

1. $choose\ x^1,\ let\ p_1=\nabla f\left(x^1\right)$
2. $if\ \nabla f\left(x^1\right)=0, stop, otherwise\ x^{k+1}=x^k+a_k p_k$

$$a_k=\frac{-p_k^T\nabla f\left(x^k\right)}{p_k^TQp_k}$$
$$p_{k+1}=-\nabla f\left(x^{k+1}\right)+\lambda_k p_k$$
$$\lambda_k=\frac{p_k^TQ\left(\nabla f\left(x^{k+1}\right)\right)}{p_k^TQp_k}$$

3. $k=k+1, return\ to\ step\ 2$

For non-quadratic equation

$$f(x)\approx f\left(x^k\right)+\nabla f\left(x^k\right)\left(x-x^k\right)+\frac{1}{2}\left(x-x^k\right)^T\nabla^2 f\left(x^k\right)\left(x-x^k\right)$$

$$\lambda_k = \frac{p_k^T Q\left(\nabla f\left(x^{k+1}\right)\right)}{p_k^T Q p_k} = \frac{a_k p_k^T Q\left(\nabla f\left(x^{k+1}\right)\right)}{a_k p_k^T Q p_k}$$

$$= \frac{\left(Q p_k a_k\right)^T \left(\nabla f\left(x^{k+1}\right)\right)}{\left(Q p_k a_k\right)^T p_k} = \frac{\left(Q\left(x^{k+1} - x^k\right)\right)^T \left(\nabla f\left(x^{k+1}\right)\right)}{\left(Q\left(x^{k+1} - x^{j+1}\right)\right)^T p_k}$$

$$= \frac{\left(\nabla f\left(x^{k+1}\right) - \nabla f\left(x^k\right)\right)^T \left(\nabla f\left(x^{k+1}\right)\right)}{\left(\nabla f\left(x^{k+1}\right) - \nabla f\left(x^k\right)\right)^T p_k}$$

$$= \frac{\left\|\nabla f\left(x^{k+1}\right)\right\|^2}{\left\|\nabla f\left(x^k\right)\right\|^2}$$

Non-quadratic equation

$$\min f(x)$$

$$1.\ choose\ x^1,\ let\ p_1 = \nabla f\left(x^1\right)$$

$$2.\ if\ \nabla f\left(x^1\right) = 0, stop, otherwise\ x^{k+1} = x^k + a_k p_k$$

$$a_k = \arg\min\left(f\left(x^k + a_k p_k\right)\right)$$

$$p_{k+1} = -\nabla f\left(x^{k+1}\right) + \lambda_k p_k$$

$$\lambda_k = \frac{\left\|\nabla f\left(x^{k+1}\right)\right\|^2}{\left\|\nabla f\left(x^k\right)\right\|^2}$$

$$3.\ k = k+1, return\ to\ step\ 2$$

Newton Method

$$f(x) \approx f\left(x^k\right) + \nabla f\left(x^k\right)\left(x - x^k\right) + \frac{1}{2}\left(x - x^k\right)^T \nabla^2 f\left(x^k\right)\left(x - x^k\right)$$

The first derivative

$$\nabla f\left(x^*\right) = \nabla f\left(x^k\right) + \nabla^2 f\left(x^k\right)\left(x^* - x^k\right)$$

$$\Leftrightarrow x^* = x^k - \frac{\nabla f\left(x^k\right)}{\nabla^2 f\left(x^k\right)}$$

Quasi-Newton Method

$$d_k = H_k \nabla f\left(x^k\right)$$

$$x^{k+1} = x^k - a_k d_k$$

After k+1 iterations, we use Taylor Expansion

$$f(x) \approx f\left(x^{k+1}\right) + \nabla f\left(x^{k+1}\right)\left(x - x^{k+1}\right) + \frac{1}{2}\left(x - x^{k+1}\right)^T \nabla^2 f\left(x^{k+1}\right)\left(x - x^{k+1}\right)$$

Find the derivative of f(x)

$$\nabla f(x) - \nabla f\left(x^{k+1}\right) = \nabla^2 f\left(x^{k+1}\right)\left(x - x^{k+1}\right)$$

Let

$$x = x^k, g_k = \nabla f\left(x^k\right)$$

$$\nabla f\left(x^{k+1}\right) - \nabla f\left(x^k\right) = \nabla^2 f\left(x^{k+1}\right)\left(x^{k+1} - x^k\right)$$

$$\Leftrightarrow g_{k+1} - g_k = \nabla^2 f\left(x^{k+1}\right)\left(x^{k+1} - x^k\right)$$

$$s_k = x^{k+1} - x^k$$

$$y_k = g_{k+1} - g_k$$

$$H_{k+1} = \left(\nabla^2 f\left(x^{k+1}\right)\right)^{-1}$$

$$B_{k+1} = \left(H_{k+1}\right)^{-1} = \nabla^2 f\left(x^{k+1}\right)$$

$$y_k = B_{k+1} s_k$$

$$s_k = H_{k+1} y_k$$

DFP

$$H_{k+1} = H_k + E_k$$

$$E_k = a_k U_k U_k^T + b_k V_k V_k^T$$

$$s_k = \left(H_k + a_k U_k U_k^T + b_k V_k V_k^T\right) y_k$$

$$\Leftrightarrow s_k - H_k y_k = a_k U_k U_k^T y_k + b_k V_k V_k^T y_k$$

$$let$$

$$s_k = a_k U_k U_k^T y_k$$

$$-H_k y_k = b_k V_k V_k^T y_k$$

$$U_k = s_k \Rightarrow a_k = \frac{1}{U_k^T y_k} = \frac{1}{s_k^T y_k}$$

$$V_k = -H_k y_k \Rightarrow b_k = \frac{1}{V_k^T y_k} = \frac{1}{\left(-H_k y_k\right)^T y_k} = \frac{-1}{y_k^T H_k y_k}$$

$$H_{k+1} = H_k + \frac{s_k s_k^T}{s_k^T y_k} - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k}$$

BFGS

$$B_{k+1} = B_k + E_k$$
$$E_k = a_k U_k U_k^T + b_k V_k V_k^T$$
$$y_k = \left(B_k + a_k U_k U_k^T + b_k V_k V_k^T\right) s_k$$
$$\Leftrightarrow y_k - B_k s_k = a_k U_k U_k^T s_k + b_k V_k V_k^T s_k$$
$$let$$
$$y_k = a_k U_k U_k^T s_k$$
$$-B_k s_k = b_k V_k V_k^T s_k$$
$$U_k = y_k \Rightarrow a_k = \frac{1}{U_k^T s_k} = \frac{1}{y_k^T s_k}$$
$$V_k = -B_k s_k \Rightarrow b_k = \frac{1}{V_k^T s_k} = \frac{1}{\left(-B_k s_k\right)^T s_k} = \frac{-1}{s_k^T B_k s_k}$$
$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{B_k y_k y_k^T B_k}{s_k^T B_k s_k}$$
$$H_{k+1} = \left(B_{k+1}\right)^{-1} = \left(B_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{B_k y_k y_k^T B_k}{s_k^T B_k s_k}\right)^{-1}$$
$$H_{k+1} = \left(I - \frac{s_k y_k^T}{y_k^T s_k}\right) H_k \left(I - \frac{y_k s_k^T}{y_k^T s_k}\right) + \frac{s_k s_k^T}{y_k^T s_k}$$

L-BFGS

$$\rho_k = \frac{1}{y_k^T s_k}, V_k = 1 - \rho_k y_k s_k^T$$
$$H_{k+1} = V_k^T H_k V_k + \rho_k s_k s_k^T$$
$$\begin{aligned} H_k = {} & \left(V_{k-1}^T V_{k-2}^T \ldots V_0^T\right) H_0 \left(V_0 V_1 \ldots V_{k-1}\right) + \\ & \left(V_{k-1}^T V_{k-2}^T \ldots V_0^T\right) \rho_0 s_0 s_0^T \left(V_0 V_1 \ldots V_{k-1}\right) + \\ & \qquad + \cdots \\ & \left(V_{k-1}^T\right) \rho_{k-2} s_{k-2} s_{k-2}^T \left(V_{k-1}\right) + \rho_{k-1} s_{k-1} s_{k-1}^T \end{aligned}$$

For m elements,

$$\begin{aligned} H_k = {} & \left(V_{k-1}^T V_{k-2}^T \ldots V_{k-m}^T\right) H_{k-m} \left(V_{k-m} V_{k-m+1} \ldots V_{k-1}\right) + \\ & \left(V_{k-1}^T V_{k-2}^T \ldots V_{k-m+1}^T\right) \rho_{k-m} s_{k-m} s_{k-m}^T \left(V_{k-m+1} V_{k-m} \ldots V_{k-1}\right) + \\ & \left(V_{k-1}^T V_{k-2}^T \ldots V_{k-m+2}^T\right) \rho_{k-m+1} s_{k-m+1} s_{k-m+1}^T \left(V_{k-m+2} V_{k-m} \ldots V_{k-1}\right) + \\ & \qquad + \cdots \\ & \left(V_{k-1}^T\right) \rho_{k-2} s_{k-2} s_{k-2}^T \left(V_{k-1}\right) + \rho_{k-1} s_{k-1} s_{k-1}^T \end{aligned}$$

L-BFGS

$let\ q = \nabla f\left(x^{k+1}\right)$

$for\ i = 1,2,...,m\ do$

$\quad a_i = \rho_{k-i} s_{k-i}^T q$

$\quad q = q - a_i y_{k-i}$

end

$r = H_{k-m} q$

$for\ i = m, m-1,...,1\ do$

$\quad \beta = \rho_{k-i} y_{k-i}^T r$

$\quad r = r + s_{k-i}\left(a_i - \beta\right)$

end