- H-K 算法

H-K 算法是求解 Xw=b，式中 $b=(b_1, b_2, \ldots, b_n)^T$，b 的所有分量都是正值。这里要同时计算 w 和 b，我们已知 X 不是 N*N 的方阵，通常是行多于列的 N*(n+1)阶的长方阵，属于超定方程，因此一般情况下，Xw=b 没有唯一确定解，但可求其线性最小二乘解。

设 Xw=b 的线性最小二乘解为 w*，即使||Xw*-b||=极小

采用梯度法，定义准则函数：

$$J(w,x,b)=\frac{1}{2}\sum_{i=1}^{n}(w^T x_i-b_i)^2=\frac{1}{2}\|Xw-b\|^2=\frac{1}{2}(Xw-b)^T(Xw-b)$$

当 Xw=b 的条件满足时，J 达到最小值。由于上式中包括的 $\sum_{i=1}^{n}(w^T x_i-b_i)^2$ 项为两个数量方差的和，且我们将使其最小化，因此也称之为最小均方误差算法。

使函数 J 同时对变量 w 和 b 求最小。对于 w 的梯度为：

$$\frac{\partial J}{\partial w}=X^T(Xw-b)$$

使 $\frac{\partial J}{\partial w}=0$，得 $X^T(Xw-b)=0$，从而 $X^TXw=X^Tb$。因为 X^TX 为(n+1)*(n+1)阶方阵，因此可求得解：

$$w=(X^TX)^{-1}X^Tb=X^{\#}b$$

这里 $X^{\#}=(X^TX)^{-1}X^T$ 称为 X 的伪逆，X 是 N*(n+1)阶的长方阵。

由上式可知，只要求出 b 即可求得 w。利用梯度法可求得 b 的迭代公式为：

$$b(k+1) = b(k) - C \cdot \left(\frac{\partial J}{\partial b} \right)_{b=b(k)}$$

根据上述约束条件，在每次迭代中，b(k)的全部分量只能是正值。由 J 的准则函数式，J 也是正值，因此，当取校正增量 C 为正值时，为保证每次迭代中的 b(k)都是正值，应使$\left(\frac{\partial J}{\partial b} \right)_{b=b(k)}$为非正值。在此条件下，准则函数 J 的微分为：

$$-2\left(\frac{\partial J}{\partial b} \right)_{b=b(k)} = (Xw - b) + |Xw - b|$$

该式满足以下条件：

$$\text{若}[Xw(k) - b(k)] > 0\text{，则} -\left(\frac{\partial J}{\partial b} \right)_{b=b(k)} = Xw(k) - b(k)$$

$$\text{若}[Xw(k) - b(k)] < 0\text{，则} -\left(\frac{\partial J}{\partial b} \right)_{b=b(k)} = 0$$

由 b 的迭代式和微分，有：

$$b(k+1) = b(k) + \delta\, b(k)$$

$$\delta\, b(k) = C[Xw(k) - b(k) + |Xw(k) - b(k)|]$$

将此式代入 $w=X^{\#}b$，有：

$$w(k+1) = X^{\#}b(k+1) = X^{\#}[b(k) + \delta\, b(k)] = w(k) + X^{\#}\delta\, b(k)$$

为简化起见，令 $e(k) = Xw(k) - b(k)$，可得 H-K 算法的迭代式。

设初值为 b(1)，其每一分量均为正值，则：

$$w(1) = X^{\#}b(1)$$

$$e(k) = Xw(k) - b(k)$$

$$\begin{aligned} w(k+1) &= w(k) + X^{\#}\{C[Xw(k) - b(k) + |Xw(k) - b(k)|]\} \\ &= w(k) + CX^{\#}[e(k) + |e(k)|] \end{aligned}$$

由于

$$X^{\#}e(k) = X^{\#}[Xw(k) - b(k)] = (X^{T}X)^{-1}X^{T}[Xw(k) - b(k)]$$

$$= w(k) - X^{\#}b(k) = 0$$

因此

$$w(k+1) = w(k) + CX^{\#}|e(k)|$$

$$b(k+1) = b(k) + C[Xw(k) - b(k) + |Xw(k) - b(k)|]$$

$$= b(k) + C[e(k) + |e(k)|]$$