

A21 Part3 DNN

Circulant Matrix (巡回行列) は

Toeplitz Matrix の特殊版

→ Discrete Fourier Transform (DFT) / 離散フーリエ変換

により対角化される

数値計算において 高速フーリエ変換が解くことが便利。

Q7, Prove any circulant matrix (巡回行列)

is commutative with shift matrix

$$(CaS)_{ij} = (SCa)_{ij}$$

これは、畳み込みとシフト演算子と可換であることに直接的に
意味合いがあることに注意してください。

→ DL Book Chapter 9

Convolution は 特殊な線形変換

入力 x (input)
関数 w (kernel)

m が「増加するにつれて」 x の
indexが「増加」kernelのindexは
「減少する意味」で「反転(flipped)」して
いる

2次元画像 I (を入力した場合の置き込み)
2次元カーネル K

$$N(i, j) = (I * K)_{ij} = \sum_m \sum_n I_{m,n} K_{i-m, j-n}$$

input

$$\begin{pmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \end{pmatrix}$$

↑
Image

kernel

$$\begin{pmatrix} w & x \\ y & z \end{pmatrix}$$

↑
Kernel

$$N(0,0) = \sum_m \sum_n I_{m,n} K_{0-m, 0-n}$$

$m \in \{0,1\}$
 $n \in \{0,1\}$

$$= I_{0,0} K_{0,0} (aw) + I_{0,1} K_{-1,0} (ex) + I_{1,0} K_{0,-1} (by) + I_{1,1} K_{-1,-1} (fz)$$

Output

$$\begin{pmatrix} aw + bx + ey + fz & bw + cx + fy + gz & cw + dx + gy + hz \\ ew + fx + iy + jz & fw + gx + jy + kz & gw + hx + ky + lz \end{pmatrix}$$

$K_{-1,-1} = K_{1,1}$ 反転している

flipped の唯一の理由は可換性を獲得するため

↳ 一般の NN においてはあまり重要でない

flipped の代わりに相互相関 (cross-correlation) と呼ばれる量計算と実装が実装される

(← flipped をしないこと)

$$N_{ij} = (I * K)_{ij} = \sum_m \sum_n I_{i+m, j+n} K_{m,n}$$

↑
ML ライブラリは相互相関を実装 (= 4x4 の量計算と実装) している。

input

$$\begin{pmatrix} 0 & 1 & 2 & 3 \\ a & b & c & d \\ 1 & e & f & g \\ 2 & i & j & k \end{pmatrix}$$

↑
Image

kernel

$$\begin{pmatrix} w & x \\ y & z \end{pmatrix}$$

↑
Kernel

$$N_{1,2} = \sum_m \sum_n I_{1+m, 2+n} K_{m,n}$$

$$= I_{1+0, 2+0} K_{0,0} \quad (gw) \\ + I_{2,2} K_{0,0} \quad (fx) \\ + I_{1,3} K_{0,1} \quad (hz) \\ + I_{2,3} K_{1,1} \quad (lz)$$

$$\begin{pmatrix} aw + bx + ey + fz & bw + cx + fy + gz & cw + dx + gy + hz \\ ew + fx + iy + jz & fw + gx + jy + kz & gw + hx + ky + lz \end{pmatrix}$$

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \end{pmatrix}$$

2次元の場合

二重循環行列 (Doubly Block

Circulant Matrix)

 $(CaS)_{ij}$ $Ca \in \mathbb{R}^{n \times n}$, S is circular shift operator

where $Ca = \begin{bmatrix} a_0 & a_{n-1} & \dots & a_1 \\ a_1 & a_0 & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ a_{n-1} & a_{n-2} & \dots & a_0 \end{bmatrix}$

$$S = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & \dots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}$$

$$Sx = S \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{n-1} \end{bmatrix} = \begin{bmatrix} x_{n-1} \\ x_0 \\ \vdots \\ x_{n-2} \end{bmatrix}$$

$$\begin{bmatrix} CaS_{11} & \dots & CaS_{1n} \\ \vdots & \ddots & \vdots \\ CaS_{n1} & \dots & CaS_{nn} \end{bmatrix}$$

Let $S = \{S_0, S_1, \dots, S_{n-1}\}$, $Ca = \{Ca_0, Ca_1, \dots, Ca_{n-1}\}^T$
 where $S_0 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}$, $S_{n-1} = \begin{pmatrix} 1 \\ \vdots \\ 0 \end{pmatrix}$ where $Ca_0 = (a_0, a_{n-1}, \dots, a_1)$
 $Ca_{n-1} = (a_{n-1}, \dots, a_0)$

$$(CaS)_{ij} = Ca_i \cdot S_j = (a_i, a_{i+1 \bmod n}, \dots, a_{i+n-1 \bmod n}) \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}$$

j+1
th row $\rightarrow$

From Youtube

$$x(n) = \{1, 1, -1, 2\}$$

$$h(n) = \{1, 2, 3, 4\}$$

$$\begin{pmatrix} 1 & 2 & -1 & 1 \\ 1 & 1 & 2 & -1 \\ -1 & 1 & 1 & 2 \\ 2 & -1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} = \begin{pmatrix} 6 \\ 5 \\ 12 \\ 7 \end{pmatrix}$$

$$x(n) * h(n) = (6, 5, 12, 7)$$

$$(CaS)_{ij} = (a_{i \bmod n}, a_{i+1 \bmod n}, \dots, a_{i+n-1 \bmod n})$$

$$\times (0 \dots 1 \dots)^T$$

$\uparrow$ j-th row

e.g.,

$$(a \oplus \underset{\substack{\uparrow \\ \text{j-th}}}{c}) \cdot d = c$$

= j-th element of
vector Ca_i

where Ca_j

$$= (a_{i \bmod n}, a_{i+1 \bmod n}, \dots, a_{i+n-1 \bmod n})$$

$$a_{i+j \bmod n}$$

$$(SCa)_{ij} = S_i C_{aj}$$

$$\left(\begin{array}{l} \text{where } S = \{S_0, S_1, \dots, S_{n-1}\}^T \\ S_0 = (0, \dots, 1), S_1 = (1, \dots, 0), S_n = (0, \dots, 1, 0) \\ \text{where } Ca = \{C_{a0}, C_{a1}, \dots, C_{a_{n-1}}\} \\ C_{a0} = (a_0, \dots, a_{n-1})^T, \dots, C_{a_{n-1}} = (a_{n-1}, \dots, a_0)^T \end{array} \right)$$

$$= (0, \dots, \underset{\substack{\uparrow \\ i\text{-th row}}}{1}, \dots, 0) \begin{pmatrix} a_{i \bmod n} \\ a_{j+1 \bmod n} \\ \vdots \\ a_{j+n-1 \bmod n} \end{pmatrix} = \underline{a_{j+i \bmod n}}$$

$$i \in \{0, \dots, n-1\}$$

$$j \in \{0, \dots, n-1\}$$

Transition or shift equivariance

$$f(Sx) = Sf(x)$$

convolution's commutativity with shift implies that it doesn't matter whether we first shift a vector and then convolve it, or first convolve and then shift. - the result will be the same.