Mathematical Foundations of Deep Neural Networks, M1407.001200
E. Ryu
Spring 2024

Due 5pm, Monday, May 06, 2024

Problem 1: *Transpose of downsampling.* Consider the downsampling operator $\mathcal{T}\colon \mathbb{R}^{m\times n} \to \mathbb{R}^{(m/2)\times(n/2)}$, defined as the average pool with a 2×2 kernel and stride 2. For the sake of simplicity, assume m and n are even. Describe the action of $\mathcal{T}^\top$. More specifically, describe how to compute $\mathcal{T}^\top(Y)$ for any $Y \in \mathbb{R}^{(m/2)\times(n/2)}$.

Clarification. The downsampling operator $\mathcal{T}$ is a linear operator (why?). Therefore, $\mathcal{T}$ has a matrix representation $A \in \mathbb{R}^{(mn/4)\times(mn)}$ such that

$$\mathcal{T}(X) = (A(X.\text{reshape}(mn))).\text{reshape}(m/2, n/2)$$

for all $X \in \mathbb{R}^{m\times n}$. The adjoint $\mathcal{T}^\top$ has two equivalent definitions. One definition is

$$\mathcal{T}^\top(Y) = (A^\top(Y.\text{reshape}(mn/4))).\text{reshape}(m, n)$$

for all $Y \in \mathbb{R}^{(m/2)\times(n/2)}$. Another is

$$\sum_{i=1}^{m/2}\sum_{j=1}^{n/2} Y_{ij}(\mathcal{T}(X))_{ij} = \sum_{i=1}^{m}\sum_{j=1}^{n}(\mathcal{T}^\top(Y))_{ij}(X)_{ij}$$

for all $X \in \mathbb{R}^{m\times n}$ and $Y \in \mathbb{R}^{(m/2)\times(n/2)}$.

Hint. To spoil the suspence, $\mathcal{T}^\top$ is a constant times the nearest neighbor upsampling. Explain why in your answer.

Solution. First, we will show that the downsampling operator $\mathcal{T}$ is a linear operator. We will abuse the notation of matrix as vector, such as X, Y as

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} = \begin{pmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{m1} \\ x_{12} \\ \vdots \\ x_{m2} \\ x_{13} \\ \vdots \\ \vdots \\ x_{mn} \end{pmatrix}$$

and

$$
Y = \begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1\frac{n}{2}} \\ y_{21} & y_{22} & \cdots & y_{2\frac{n}{2}} \\ \vdots & \vdots & \ddots & \vdots \\ y_{\frac{m}{2}1} & y_{\frac{m}{2}2} & \cdots & y_{\frac{m}{2}\frac{n}{2}} \end{pmatrix} = \begin{pmatrix} y_{11} \\ y_{21} \\ \vdots \\ y_{\frac{m}{2}1} \\ y_{12} \\ \vdots \\ y_{\frac{m}{2}2} \\ y_{13} \\ \vdots \\ \vdots \\ y_{\frac{m}{2}\frac{n}{2}} \end{pmatrix}.
$$

This is equivalent to concatenating all column vectors in one column. Then, we have

$$
\mathcal{T}(X) = \frac{1}{4} \begin{pmatrix} x_{11} + x_{12} + x_{21} + x_{22} \\ x_{31} + x_{32} + x_{41} + x_{42} \\ \vdots \\ x_{(m-1)1} + x_{(m-1)2} + x_{m1} + x_{m2} \\ \vdots \\ \vdots \\ x_{(m-1)(n-1)} + x_{(m-1)n} + x_{m(n-1)} + x_{mn} \end{pmatrix}.
$$

Then for any $\alpha \in \mathbb{R}$,

$$
\mathcal{T}(\alpha X) = \frac{1}{4} \begin{pmatrix} \alpha x_{11} + \alpha x_{12} + \alpha x_{21} + \alpha x_{22} \\ \alpha x_{11} + \alpha x_{12} + \alpha x_{21} + \alpha x_{22} \\ \vdots \\ \alpha x_{(m-1)1} + \alpha x_{(m-1)2} + \alpha x_{m1} + \alpha x_{m2} \\ \vdots \\ \vdots \\ \alpha x_{(m-1)(n-1)} + \alpha x_{(m-1)n} + \alpha x_{m(n-1)} + \alpha x_{mn} \end{pmatrix} = \frac{\alpha}{4}\mathcal{T}(X)
$$

so $\mathcal{T}$ is a linear operator.

Solution 1. We use the equation

$$
\sum_{i=1}^{m/2} \sum_{j=1}^{n/2} Y_{ij} (\mathcal{T}(X))_{ij} = \sum_{i=1}^{m} \sum_{j=1}^{n} (\mathcal{T}^{\top}(Y))_{ij} (X)_{ij}
$$

for any $X \in \mathbb{R}^{m \times n}$ and $Y \in \mathbb{R}^{(m/2)\times(n/2)}$ to describe $\mathcal{T}^{\intercal}$. To avoid confusion, we refer to the index of Y by $\bar{i}$ and $\bar{j}$. Due to the definition of our downsampling operator $\mathcal{T}$ as an average pool,

$$
(\mathcal{T}(X))_{\bar{i}\bar{j}} = \frac{1}{4} \sum_{k=1}^{2} \sum_{l=1}^{2} x_{2\bar{i}-2+k,\, 2\bar{j}-2+k}.
$$

Then we get

$$
\begin{aligned}
\sum_{\bar{i}=1}^{m/2}\sum_{\bar{j}=1}^{n/2} Y_{\bar{i}\bar{j}}(\mathcal{T}(X))_{\bar{i}\bar{j}} &= \sum_{i=1}^{m}\sum_{j=1}^{n}(\mathcal{T}^\top(Y))_{ij}(X)_{ij} \\
&= \sum_{\bar{i}=1}^{m/2}\sum_{\bar{j}=1}^{n/2} Y_{\bar{i}\bar{j}} \cdot \frac{1}{4}\left(\sum_{k=1}^{2}\sum_{l=1}^{2} x_{2\bar{i}-2+k,\, 2\bar{j}-2+k}\right) \\
&= \sum_{\bar{i}=1}^{m/2}\sum_{\bar{j}=1}^{n/2}\sum_{k=1}^{2}\sum_{l=1}^{2} \frac{1}{4} Y_{\bar{i}\bar{j}} x_{2\bar{i}-2+k,\, 2\bar{j}-2+k}
\end{aligned}
$$

From the index identification

$$
\begin{cases} i = 2\bar{i} - 2 + k & (\bar{i} = 1, \dots, \frac{m}{2},\ k = 1, 2) \\ j = 2\bar{j} - 2 + l & (\bar{j} = 1, \dots, \frac{n}{2},\ l = 1, 2) \end{cases} \quad \Longleftrightarrow \quad \begin{cases} \bar{i} = \lfloor \frac{i+1}{2} \rfloor & (i = 1, \dots, m) \\ \bar{j} = \lfloor \frac{j+1}{2} \rfloor & (j = 1, \dots, n) \end{cases}
$$

where $t \mapsto \lfloor t \rfloor$ is a floor function, we can rephrase the above summation as

$$
\begin{aligned}
\sum_{\bar{i}=1}^{m/2}\sum_{\bar{j}=1}^{n/2} Y_{\bar{i}\bar{j}}(\mathcal{T}(X))_{\bar{i}\bar{j}} &= \sum_{i=1}^{m}\sum_{j=1}^{n}(\mathcal{T}^\top(Y))_{ij}(X)_{ij} \\
&= \sum_{i=1}^{m}\sum_{j=1}^{n} \frac{1}{4} y_{\lfloor \frac{i+1}{2} \rfloor, \lfloor \frac{j+1}{2} \rfloor} x_{ij}
\end{aligned}
$$

Therefore, $(\mathcal{T}^\intercal(Y))_{ij} = y_{\lfloor \frac{i+1}{2} \rfloor, \lfloor \frac{j+1}{2} \rfloor}$ and converting this to $m \times n$ matrix, we get

$$
\mathcal{T}^\top(Y) = \frac{1}{4}\begin{pmatrix} y_{11} & y_{11} & \cdots & y_{1\frac{n}{2}} \\ y_{11} & y_{11} & \cdots & y_{1\frac{n}{2}} \\ \vdots & \vdots & \ddots & \vdots \\ y_{\frac{m}{2}1} & y_{\frac{m}{2}1} & \cdots & y_{\frac{m}{2}\frac{n}{2}} \end{pmatrix}
$$

Solution 2. Since $\mathcal{T}$ is a linear operator, we can consider $\mathcal{T}(X)$ as multiplying matrix T in the left side of X, where T is defined as

$$
T = \frac{1}{4}\begin{pmatrix} T_{11} & T_{12} & \cdots & T_{1n} \\ T_{21} & T_{22} & \cdots & T_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ T_{\frac{n}{2}1} & T_{\frac{n}{2}2} & \cdots & T_{\frac{n}{2}n} \end{pmatrix}
$$

that $T_{i,j}$ is $\frac{m}{2} \times m$ matrix, $T_{i,j} = 0$ except $j = 2 * i - 1$ or $j = 2i$, and

$$
T_{i,2i} = T_{i,2i-1} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 1 \end{pmatrix}.
$$

Therefore, $\mathcal{T}$ can be represented as matrix, and it is linear operator. Now, we calculate $\mathcal{T}^\top$ with $T^\intercal$. We have

$$T^\intercal = \frac{1}{4}\begin{pmatrix} T_{11}^\intercal & T_{21}^\intercal & \cdots & T_{\frac{n}{2}1}^\intercal \\ T_{12}^\intercal & T_{22}^\intercal & \cdots & T_{\frac{n}{2}2}^\intercal \\ \vdots & \vdots & \ddots & \vdots \\ T_{1n}^\intercal & T_{2n}^\intercal & \cdots & T_{\frac{n}{2}n}^\intercal \end{pmatrix},$$

which indicates

$$\mathcal{T}^\top(Y) = T^\intercal \begin{pmatrix} y_{11} \\ y_{21} \\ \vdots \\ y_{\frac{m}{2}1} \\ y_{12} \\ \vdots \\ y_{\frac{m}{2}2} \\ y_{13} \\ \vdots \\ \vdots \\ y_{\frac{m}{2}\frac{n}{2}} \end{pmatrix} = \frac{1}{4}\begin{pmatrix} y_{11} \\ y_{11} \\ y_{21} \\ y_{21} \\ \vdots \\ y_{\frac{m}{2}1} \\ y_{\frac{m}{2}1} \\ y_{11} \\ y_{11} \\ y_{21} \\ y_{21} \\ \vdots \\ y_{\frac{m}{2}1} \\ y_{\frac{m}{2}1} \\ \vdots \\ y_{12} \\ y_{12} \\ y_{22} \\ y_{22} \\ \vdots \\ y_{\frac{m}{2}2} \\ y_{\frac{m}{2}2} \\ \vdots \\ \vdots \\ y_{1\frac{n}{2}} \\ y_{1\frac{n}{2}} \\ y_{2\frac{n}{2}} \\ y_{2\frac{n}{2}} \\ \vdots \\ y_{\frac{m}{2}\frac{n}{2}} \\ y_{\frac{m}{2}\frac{n}{2}} \end{pmatrix}.$$

Converting this to $m \times n$ matrix, we have

$$\mathcal{T}^\top(Y) = \frac{1}{4}\begin{pmatrix} y_{11} & y_{11} & \cdots & y_{1\frac{n}{2}} \\ y_{11} & y_{11} & \cdots & y_{1\frac{n}{2}} \\ \vdots & \vdots & \ddots & \vdots \\ y_{\frac{m}{2}1} & y_{\frac{m}{2}1} & \cdots & y_{\frac{m}{2}\frac{n}{2}} \end{pmatrix}$$

and this is also the nearest neighbor upsampling. ■

Problem 2: *Nearest neighbor upsampling.* How is the nearest neighbor upsampling operator an instance of transpose convolution? Specifically, describe how

```
layer = nn.Upsample(scale_factor=r, mode='nearest')
```

where r is a positive integer, can be equivalently represented by

```
layer = nn.ConvTranspose2d(...)
layer.weight.data = ...
```

with `...` appropriately filled in.

Solution.

```
layer = nn.ConvTranspose2d(kernel_size=r, stride=r, bias=False)
layer.weight.data = torch.ones(r,r)
```

We can easily check that this perform nearest mode of Upsampling. ■

Problem 3: *f-divergence.* Let X and Y be two continuous random variables with densities p_X and p_Y. The f-divergence of X from Y is defined as

$$D_f(X\|Y) = \int f\left(\frac{p_X(x)}{p_Y(x)}\right) p_Y(x)\, dx,$$

where f is a convex function such that $f(1) = 0$.

(a) Show that $D_f(X\|Y) \geq 0$.

(b) Show that $f = -\log t$ and $f = t \log t$ correspond to the KL divergence.

Solution. (a) Use Jensen's Inequality since f is convex. We have

$$D_f(X\|Y) = \int f\left(\frac{p_X(x)}{p_Y(x)}\right) p_Y(x)\, dx \geq f\left(\int \frac{p_X(x)}{p_Y(x)} p_Y(x)\right)\, dx = f(1) = 0.$$

(c) Put $f = -\log t$, then

$$D_f(X\|Y) = \int -\log\left(\frac{p_X(x)}{p_Y(x)}\right) p_Y(x)\, dx = D_{KL}(Y\|X),$$

which is exactly same as KL divergence's definition.
Put $f = t \log t$, then

$$D_f(X\|Y) = \int \log\left(\frac{p_X(x)}{p_Y(x)}\right) p_X(x)\, dx = D_{KL}(X\|Y),$$

which is exactly same as KL divergence's definition. ■

Problem 4: *Generalized inverse transform sampling.* Let $F\colon \mathbb{R} \to [0,1]$ be the CDF of a random variable and let $U \sim \text{Uniform}([0,1])$. If F is continuous and strictly increasing and therefore invertible, then $F^{-1}(U)$ is a random variable with CDF F, because

$$\mathbb{P}(F^{-1}(U) \le t) = \mathbb{P}(U \le F(t)) = F(t).$$

When F is not necessarily invertible, the *generalized inverse* of F is $G\colon (0,1) \to \mathbb{R}$ with

$$G(u) = \inf\{x \in \mathbb{R} \mid u \le F(x)\}.$$

Show that $G(U)$ is a random variable with CDF F.

Hint. Use the fact that F is right-continuous, i.e., $\lim_{h\to 0^+} F(x+h) = F(x)$ for all $x \in \mathbb{R}$, and that $\lim_{x\to -\infty} F(x) = 0$.

Solution. We first show $G(u) \le x \iff u \le F(x)$ for any $u \in (0,1)$ and $x \in \mathbb{R}$.
($\Leftarrow$) If $u \le F(x)$, then $G(u) = \inf\{w \mid u \le F(w)\} \le x$.
($\Rightarrow$) The infimum of $G(u) = \inf\{x \in \mathbb{R} \mid u \le F(x)\}$ is attained since F right-continuous and nondecreasing and since $\lim_{x\to-\infty} F(x) = 0$. Therefore, $u \le F(G(u)) \le F(x)$, since F is nondecreasing.

We now complete the proof:

$$\mathbb{P}\left(G(U) \le x\right) = \mathbb{P}\left(U \le F(x)\right) = F(x).$$

■

Problem 5: *Change of variables formula for Gaussians.* If $\varphi\colon \mathbb{R}^n \to \mathbb{R}^n$ is a one-to-one differentiable function, $Y = \varphi(X)$, and Y is a continuous random variable with density function p_Y, then X is a continuous random variable with density function

$$p_X(x) = p_Y(\varphi(x)) \left|\det \frac{\partial \varphi}{\partial x}(x)\right|.$$

Let $Y \in \mathbb{R}^n$ be a continuous random vector with density

$$p_Y(y) = \frac{1}{(2\pi)^{n/2}} e^{-\frac{1}{2}\|y\|^2},$$

i.e., $Y \sim \mathcal{N}(0, I)$. Let $X = AY + b$ with an invertible matrix $A \in \mathbb{R}^{n\times n}$ and a vector $b \in \mathbb{R}^n$. Define $\Sigma = AA^\intercal$. Show that X is a continuous random vector with density

$$p_X(x) = \frac{1}{\sqrt{(2\pi)^n \det \Sigma}} e^{-\frac{1}{2}(x-b)^\intercal \Sigma^{-1}(x-b)}.$$

Solution. $\left|\det A^{-1}\right| = |\det A|^{-1} = |\det A \det A^\intercal|^{-\frac{1}{2}} = |\det \Sigma|^{-\frac{1}{2}}$, so

$$\begin{aligned}
&p_X(x) \\
&= p_Y(A^{-1}(x-b)) \left|\det \frac{\partial A^{-1}(x-b)}{\partial x}\right| \\
&= p_Y(A^{-1}(x-b)) \left|\det A^{-1}\right| \\
&= \frac{1}{(2\pi)^{n/2}} e^{-\frac{1}{2}\|A^{-1}(x-b)\|^2} |\det \Sigma|^{-\frac{1}{2}} \\
&= \frac{1}{\sqrt{(2\pi)^n \det \Sigma}} e^{-\frac{1}{2}(x-b)^\intercal A^{-\intercal} A^{-1}(x-b)} \\
&= \frac{1}{\sqrt{(2\pi)^n \det \Sigma}} e^{-\frac{1}{2}(x-b)^\intercal \Sigma^{-1}(x-b)}.
\end{aligned}$$

■

Problem 6: *Inverse permutation.* Let S_n denote the group of length-n permutations. Note that the map $i \mapsto \sigma(i)$ is a bijection. Define $\sigma^{-1} \in S_n$ as the permutation representing the inverse of this map, i.e, $\sigma^{-1}(\sigma(i)) = i$ for $i = 1, \ldots, n$. Describe an algorithm for computing σ^{-1} given σ.

Clarification. In this class, we defined σ as a list of length n containing the elements of $\{1, \ldots, n\}$ exactly once. The output of the algorithm, σ^{-1}, should also be provided as a list.

Clarification. For this problem, it is sufficient to describe the algorithm in equations or pseudocode. There is no need to submit a Python script for this problem.

Solution. When the input permutation $\sigma \in S_n$ is in form of list as $\sigma[i] = \sigma(i)$, we can calculate the inverse of given permutation σ in the following steps.

1. Define a list π with a length n.
2. For $i = 1, ..., n$,
   repeat $\pi[\sigma[i]] \leftarrow i$
3. return π

π is σ^{-1}, an inverse of σ since $\pi(\sigma(i)) = \pi[\sigma[i]] = i$ for $i = 1, ..., n$. ■

Problem 7: *Permutation matrix.* Given a permutation $\sigma \in S_n$, the *permutation matrix* of σ is defined as

$$P_\sigma = \begin{bmatrix} e_{\sigma(1)}^\intercal \\ e_{\sigma(2)}^\intercal \\ \vdots \\ e_{\sigma(n)}^\intercal \end{bmatrix} \in \mathbb{R}^{n \times n},$$

where $e_1, \ldots, e_n \in \mathbb{R}^n$ are the standard unit vectors. Show

(a) $(P_\sigma x)_i = x_{\sigma(i)}$ for all $x \in \mathbb{R}^n$ and $i = 1, \ldots, n$,

(b) $P_\sigma^\intercal = P_\sigma^{-1} = P_{\sigma^{-1}}$ and

(c) $|\det P_\sigma| = 1$.

Hint. If the rows of $U \in \mathbb{R}^{n \times n}$ are orthonormal, we say U is an orthogonal matrix. Orthogonal matrices satisfy $UU^\intercal = U^\intercal U = I$.

Solution.

(a) This result comes from $(P_\sigma x)_i = e_{\sigma(i)}^\intercal x = \left\langle e_{\sigma(i)}, \sum_{i=1}^n x_i e_i \right\rangle = x_{\sigma(i)}$.

(b) First, P_σ is an orthogonal matrix since $\left\langle e_{\sigma(i)}, e_{\sigma(j)} \right\rangle = \delta_{\sigma(i)\sigma(j)} = \delta_{ij}$ due to the bijectivity of σ. Hence, $P_\sigma^\intercal = P_\sigma^{-1}$.

Second equality comes from $P_\pi P_\sigma = P_{\sigma \circ \pi}$. Since $P_\sigma P_{\sigma^{-1}} = P_{\sigma^{-1} \circ \sigma} = P_{id} = I$ where $id \in S_n$ is an identity permutation, it follows that $P_{\sigma^{-1}} = P_\sigma^{-1}$.

$$P_\pi P_\sigma = \begin{bmatrix} e_{\pi(1)}^\intercal \\ e_{\pi(2)}^\intercal \\ \vdots \\ e_{\pi(n)}^\intercal \end{bmatrix} \begin{bmatrix} e_{\sigma(1)}^\intercal \\ e_{\sigma(2)}^\intercal \\ \vdots \\ e_{\sigma(n)}^\intercal \end{bmatrix} = \begin{bmatrix} e_{\sigma(\pi(1))}^\intercal \\ e_{\sigma((\pi 2))}^\intercal \\ \vdots \\ e_{\sigma(\pi(n))}^\intercal \end{bmatrix} = P_{\sigma \circ \pi}$$

(c) Since P_σ is an orthogonal matrix, $1 = \det(P_\sigma^\intercal P_\sigma) = \det(P_\sigma)^2$. Thus, $|\det(P_\sigma)| = 1$. ■