MLE Exercise 6

1. Convolutions of Continuous and Discrete Variables

1(a) Adding two Gaussians

Let

$$X \sim \mathcal{N}(\mu_1, \sigma_1^2), \qquad Y \sim \mathcal{N}(\mu_2, \sigma_2^2), \qquad Z = X + Y,$$

with X and Y *independent*.

Because of independence we can write the pdf of Z as a convolution of pdfs of X and Y:

$$f_Z(z) = (f_X * f_Y)(z) = \int_{-\infty}^{\infty} f_X(x)\, f_Y(z-x)\, dx.$$

This comes from the definition of convolution.

Both f_X and f_Y have the familiar bell-curve form

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\Big[-\frac{(x-\mu_1)^2}{2\sigma_1^2}\Big], \quad f_Y(y) = \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp\Big[-\frac{(y-\mu_2)^2}{2\sigma_2^2}\Big].$$

Substituting $y = z - x$ for the second one gives

$$f_Z(z) \;=\; \frac{1}{2\pi\sigma_1\sigma_2} \int_{-\infty}^{\infty} \exp\Big[-\frac{(x-\mu_1)^2}{2\sigma_1^2} - \frac{(z-x-\mu_2)^2}{2\sigma_2^2}\Big] dx.$$

Completing the square: The exponent is a quadratic in x plus some terms that do *not* depend on x. We collect the quadratic terms and rewrite them as a single perfect square:

$$-\frac{1}{2}\Big[\frac{(x-\mu_1)^2}{\sigma_1^2} + \frac{(z-x-\mu_2)^2}{\sigma_2^2}\Big] \;=\; -\frac{(x-\mu_\star)^2}{2\Sigma^2} \;-\; \frac{\big(z-(\mu_1+\mu_2)\big)^2}{2(\sigma_1^2+\sigma_2^2)},$$

where

$$\Sigma^2 = \frac{\sigma_1^2\sigma_2^2}{\sigma_1^2+\sigma_2^2}, \qquad \mu_\star = \Sigma^2\Big(\frac{\mu_1}{\sigma_1^2} + \frac{z-\mu_2}{\sigma_2^2}\Big).$$

The factor $e^{-(x-\mu_\star)^2/(2\Sigma^2)}$ integrates to $\sqrt{2\pi\Sigma^2}$ (the standard Gaussian integral), so we are left with

$$f_Z(z) = \frac{1}{\sqrt{2\pi(\sigma_1^2+\sigma_2^2)}} \exp\Big[-\frac{(z-(\mu_1+\mu_2))^2}{2(\sigma_1^2+\sigma_2^2)}\Big].$$

This *is* a Gaussian pdf, with mean $\mu_1+\mu_2$ and variance $\sigma_1^2+\sigma_2^2$. Therefore the final result:

$$\boxed{Z \sim \mathcal{N}\big(\mu_1+\mu_2,\ \sigma_1^2+\sigma_2^2\big)}.$$

1(b) Output size of a 2-D convolution layer

Problem: Given

- input image size (H, W),
- kernel / filter size (K_H, K_W),
- *zero-padding* of p pixels on *each* edge, and
- stride s (same in both spatial dimensions),

derive the height H_{out} and width W_{out} of the resulting feature map.

Answer: Padding inflates the useful image region to $(H+2p)\times(W+2p)$. With stride s, the kernel's top-left corner "jumps" by s pixels each step.

One dimension first. Along, the vertical axis, the kernel can land at positions.

$$0,\ s,\ 2s,\ \ldots,\ \big(H_{\text{eff}}-K_H\big), \quad \text{where } H_{\text{eff}} = H+2p.$$

Counting how many multiples of s fit into that range gives

$$\Big\lfloor \frac{H_{\text{eff}}-K_H}{s} \Big\rfloor + 1.$$

Exactly the same argument applies horizontally.

Final formulae:

$$\boxed{H_{\text{out}} = \Big\lfloor \frac{H+2p-K_H}{s} \Big\rfloor + 1}, \qquad \boxed{W_{\text{out}} = \Big\lfloor \frac{W+2p-K_W}{s} \Big\rfloor + 1}.$$

Special-cases:

- *"Valid" convolution* ($p=0,\ s=1$): $H_{\text{out}} = H-K_H+1$, $W_{\text{out}} = W-K_W+1$.
- *"Same" convolution* (choose $p=\frac{K_H-1}{2}$, K odd, $s=1$): $H_{\text{out}} = H$ etc. Nice consistency check!
- *Down-sampling* ($s>1$) visibly reduces the size.