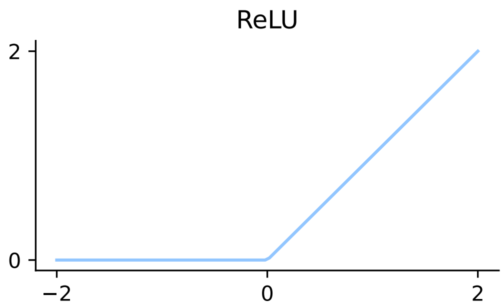
Activation Functions

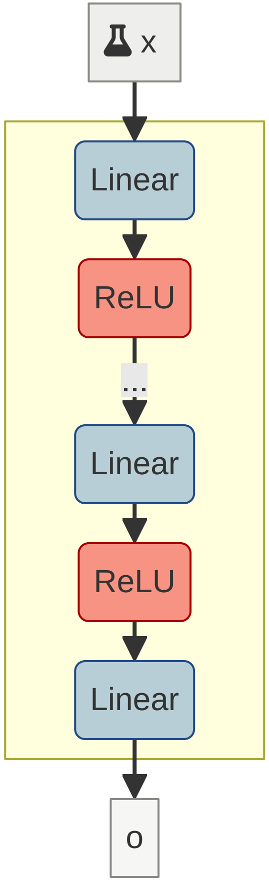
Recap: Non-Linearities

Rectified Linear Unit (ReLU)

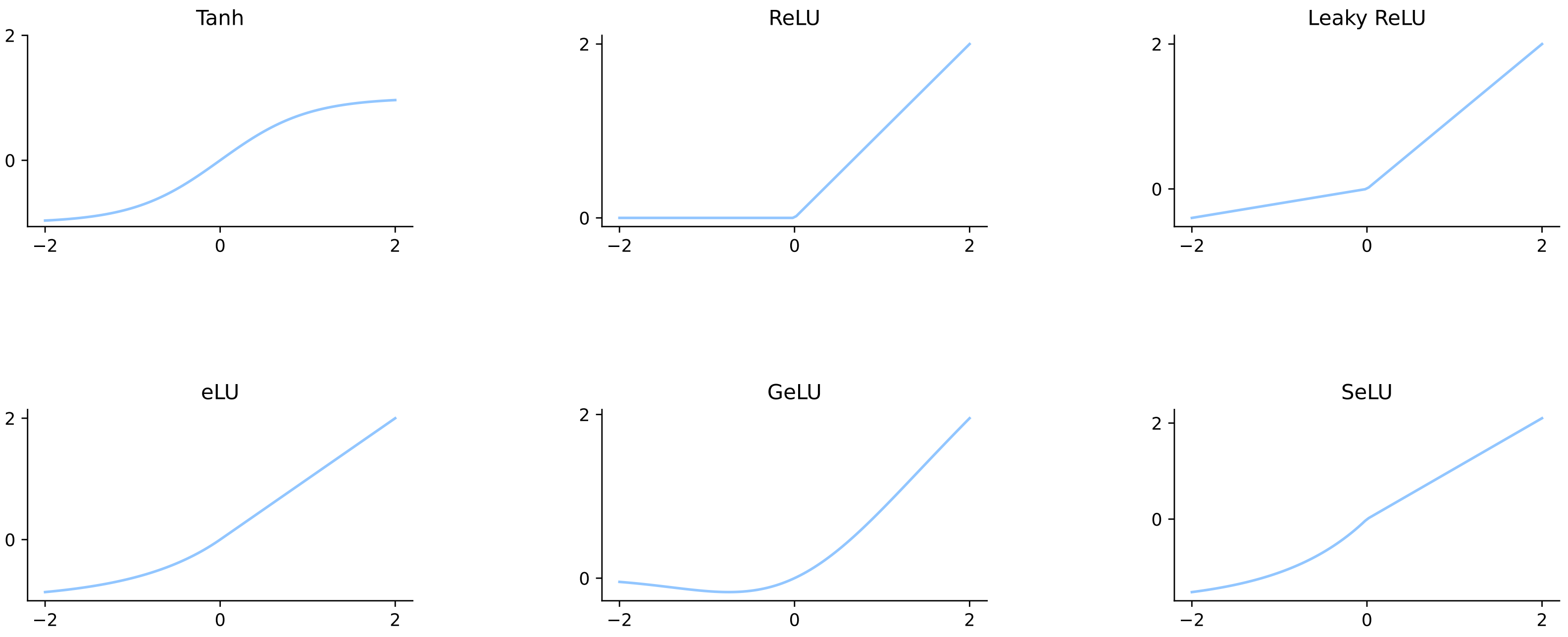
$$\mathrm{ReLU}(x) = \max(x, 0)$$

Allows deep networks to model arbitrary differentiable functions



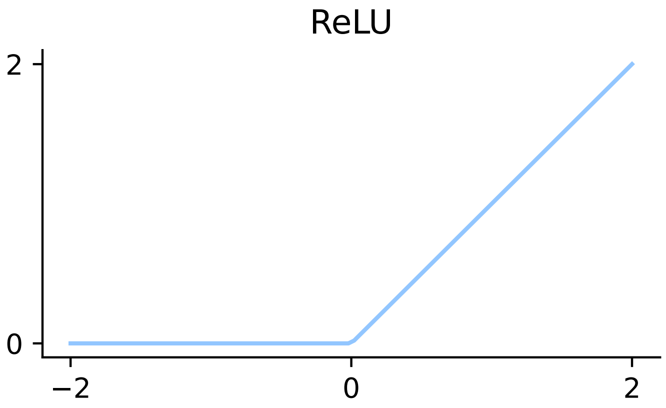


Zoo of Activation Functions



ReLU

$$\mathrm{ReLU}(x) = \max(x, 0)$$



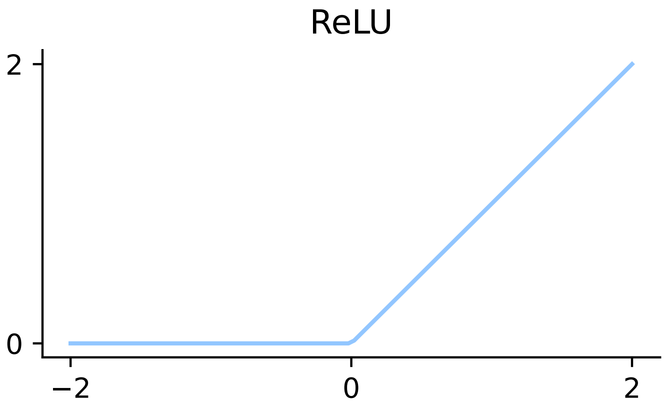
✓ Simple

✗ ReLU units can be fragile during training and "die"

Dead ReLUs

How can we prevent dead ReLUs?

- Initialize network carefully
- Decrease the learning rate



Leaky ReLU

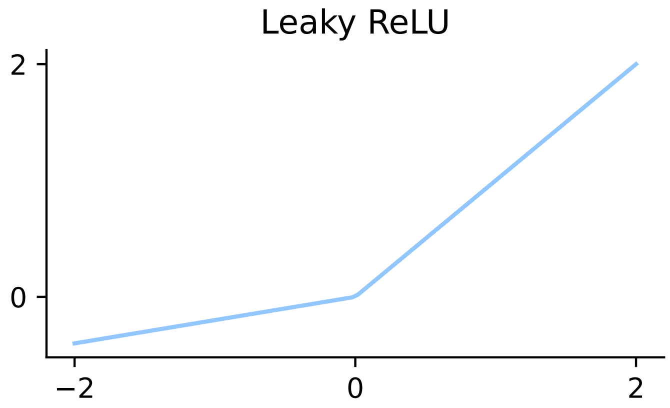
$$\text{LeakyReLU}(x) = \max(x, \alpha x)$$

- Where $0 < \alpha < 1$
- Called **PReLU** if α is learned

✓ Non-negative gradient for negative inputs

✗ The slope α needs to be tuned

✗ Cannot wipe the negative signal out



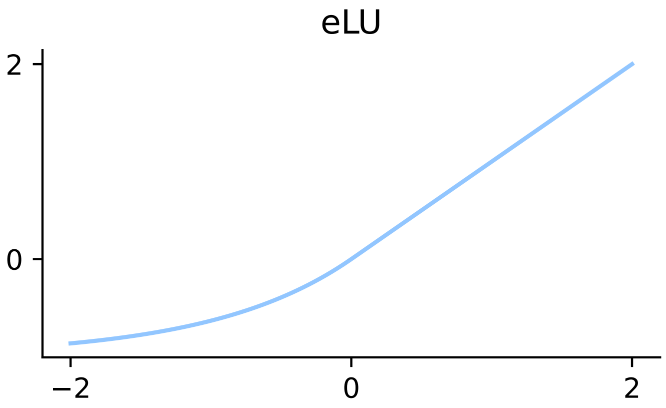
Elu

$$\mathrm{ELU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha(e^x - 1) & \text{if } x < 0 \end{cases}$$

✓ Non-negative gradient for negative inputs

✗ α needs to be tuned

✗ Exponential is computationally expensive



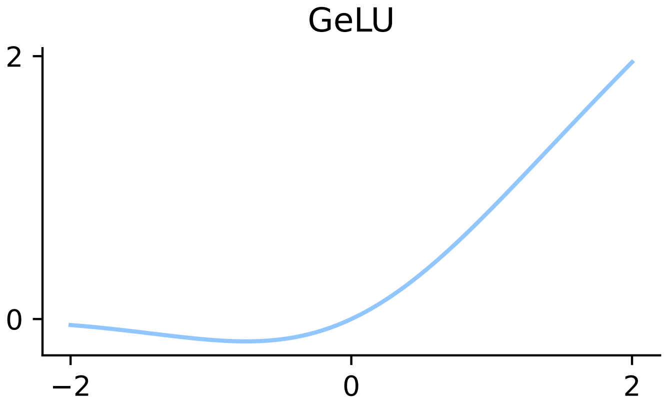
GeLU

$$\mathrm{GeLU}(x) = x \times \Phi(x)$$

- Where $\Phi(x)$ is the CDF of the standard Gaussian
- $\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{t^2}{2}} dt$

✓ Non-zero gradient for negative inputs

✗ Requires more computation



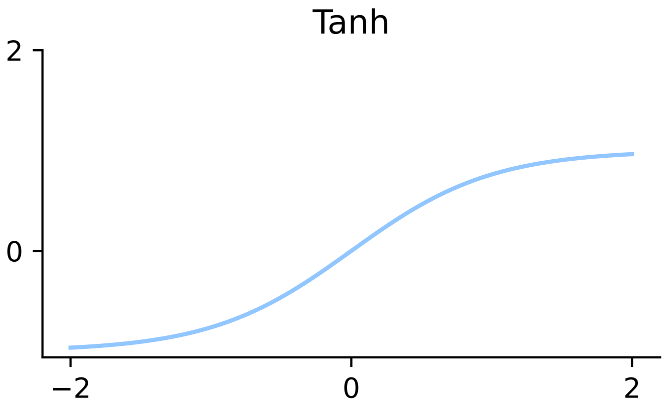
Sigmoid

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

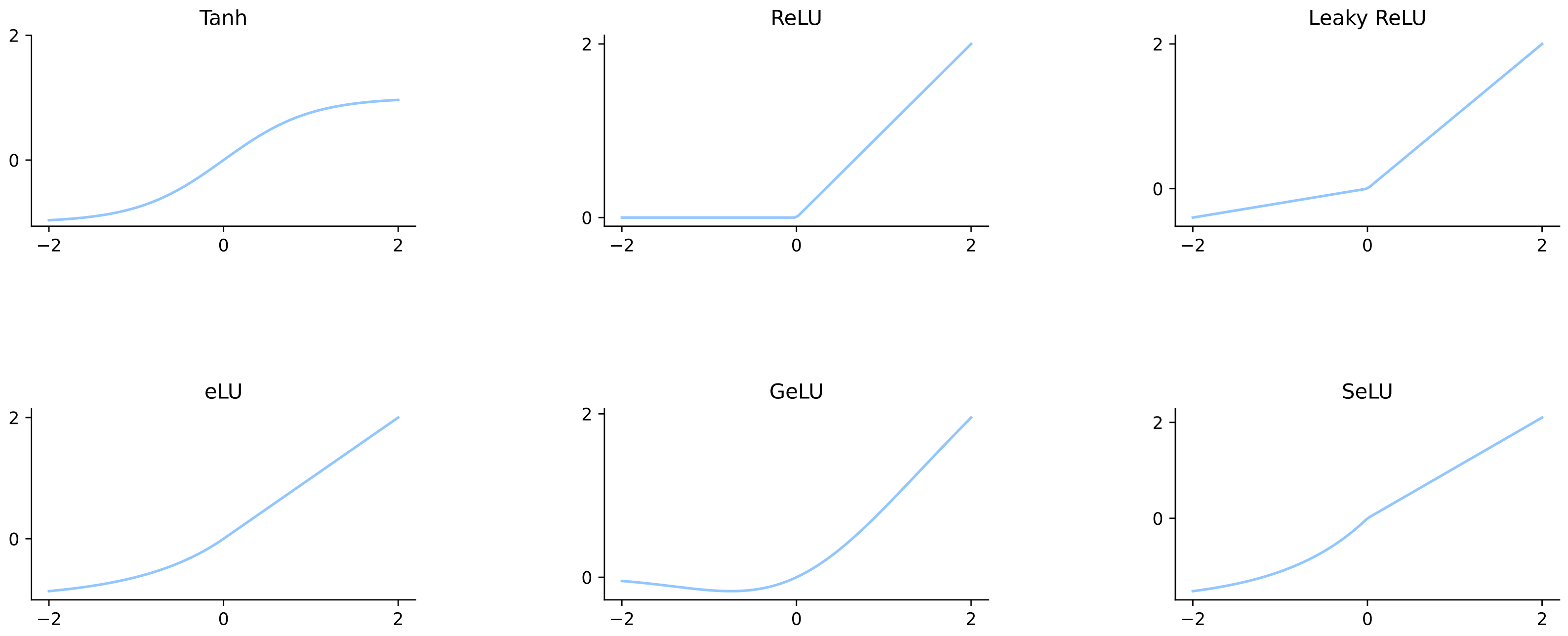
- Same as $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$

✗ Saturates on both ends

✗ Do **not** use sigmoid/tanh



Which Activation to Choose?



Activation Functions - TL;DR

Use ReLU with careful initialization and small learning rate

If ReLU fails, try Leaky ReLU or PReLU

Avoid Sigmoid and Tanh

Use GeLU for sophisticated models