✅ ReLU

$$f(x) = \max(0, x)$$

```
def relu(x):
    return max(0, x)
```

✅ Leaky ReLU

$$f(x) = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases}$$

```
def leaky_relu(x, alpha=0.01):
    return x if x > 0 else alpha * x
```

✅ Sigmoid

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

```
import math
def sigmoid(x):
    return 1 / (1 + math.exp(-x))
```

✅ Tanh

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

```
import math
def tanh(x):
    e_pos = math.exp(x)
    e_neg = math.exp(-x)
    return (e_pos - e_neg) / (e_pos + e_neg)
```

✅ GELU（近似）

$$f(x) \approx 0.5x\left(1 + \tanh\left(\sqrt{\frac{2}{\pi}}(x + 0.044715x^3)\right)\right)$$

```python
import math
def gelu(x):
    return 0.5 * x * (1 + math.tanh(math.sqrt(2 / math.pi) * (x + 0.044715 * x**3)))
```

✅ SiLU / Swish

$$f(x) = x \cdot \mathrm{sigmoid}(x) = \frac{x}{1 + e^{-x}}$$

```python
import math
def silu(x):
    return x * (1 / (1 + math.exp(-x)))
```

✅ ELU

$$f(x) = \begin{cases} x, & x > 0 \\ \alpha(e^x - 1), & x \le 0 \end{cases}$$

```python
import math
def elu(x, alpha=1.0):
    return x if x > 0 else alpha * (math.exp(x) - 1)
```

✅ Softplus

$$f(x) = \log(1 + e^x)$$

```python
import math
def softplus(x):
    return math.log(1 + math.exp(x))
```

✅ Softmax（向量输入）

$$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

```python
import math
def softmax(xs):
    exps = [math.exp(x) for x in xs]
    sum_exps = sum(exps)
    return [e / sum_exps for e in exps]
```