```python
import numpy as np
import matplotlib.pyplot as plt

# Define activation functions and their derivatives
def sigmoid(x):
    return 1 / (1 + np.exp(-x))

def sigmoid_derivative(x):
    return sigmoid(x) * (1 - sigmoid(x))

def tanh(x):
    return np.tanh(x)

def tanh_derivative(x):
    return 1 - np.tanh(x)**2

def relu(x):
    return np.maximum(0, x)

def relu_derivative(x):
    return np.where(x > 0, 1, 0)

def leaky_relu(x, alpha=0.01):
    return np.where(x > 0, x, alpha * x)

def leaky_relu_derivative(x, alpha=0.01):
    return np.where(x > 0, 1, alpha)

def gelu(x):
    return 0.5 * x * (1 + np.tanh(np.sqrt(2 / np.pi) * (x + 0.044715 * x**3)))

def gelu_derivative(x):
    cdf = 0.5 * (1 + np.tanh(np.sqrt(2 / np.pi) * (x + 0.044715 * x**3)))
    pdf = np.exp(-0.5 * x**2) / np.sqrt(2 * np.pi)
    return cdf + x * pdf

def silu(x):
    return x * sigmoid(x)

def silu_derivative(x):
    sig = sigmoid(x)
    return sig * (1 + x * (1 - sig))

# Plot settings
x = np.linspace(-10, 10, 400)

# Plot activation functions
plt.figure(figsize=(12, 12))

plt.subplot(3, 2, 1)
plt.plot(x, sigmoid(x), label='Sigmoid')
plt.plot(x, sigmoid_derivative(x), label='Sigmoid Derivative')
plt.title('Sigmoid and its Derivative')
plt.legend()

plt.subplot(3, 2, 2)
plt.plot(x, tanh(x), label='Tanh')
plt.plot(x, tanh_derivative(x), label='Tanh Derivative')
plt.title('Tanh and its Derivative')
plt.legend()

plt.subplot(3, 2, 3)
plt.plot(x, relu(x), label='ReLU')
plt.plot(x, relu_derivative(x), label='ReLU Derivative')
plt.title('ReLU and its Derivative')
plt.legend()

plt.subplot(3, 2, 4)
plt.plot(x, leaky_relu(x), label='Leaky ReLU')
plt.plot(x, leaky_relu_derivative(x), label='Leaky ReLU Derivative')
plt.title('Leaky ReLU and its Derivative')
plt.legend()

plt.subplot(3, 2, 5)
plt.plot(x, gelu(x), label='GELU')
plt.plot(x, gelu_derivative(x), label='GELU Derivative')
plt.title('GELU and its Derivative')
plt.legend()

plt.subplot(3, 2, 6)
plt.plot(x, silu(x), label='SiLU')
```

```
plt.plot(x, silu_derivative(x), label='SiLU Derivative')
plt.title('SiLU and its Derivative')
plt.legend()

plt.tight_layout()
plt.show()
```