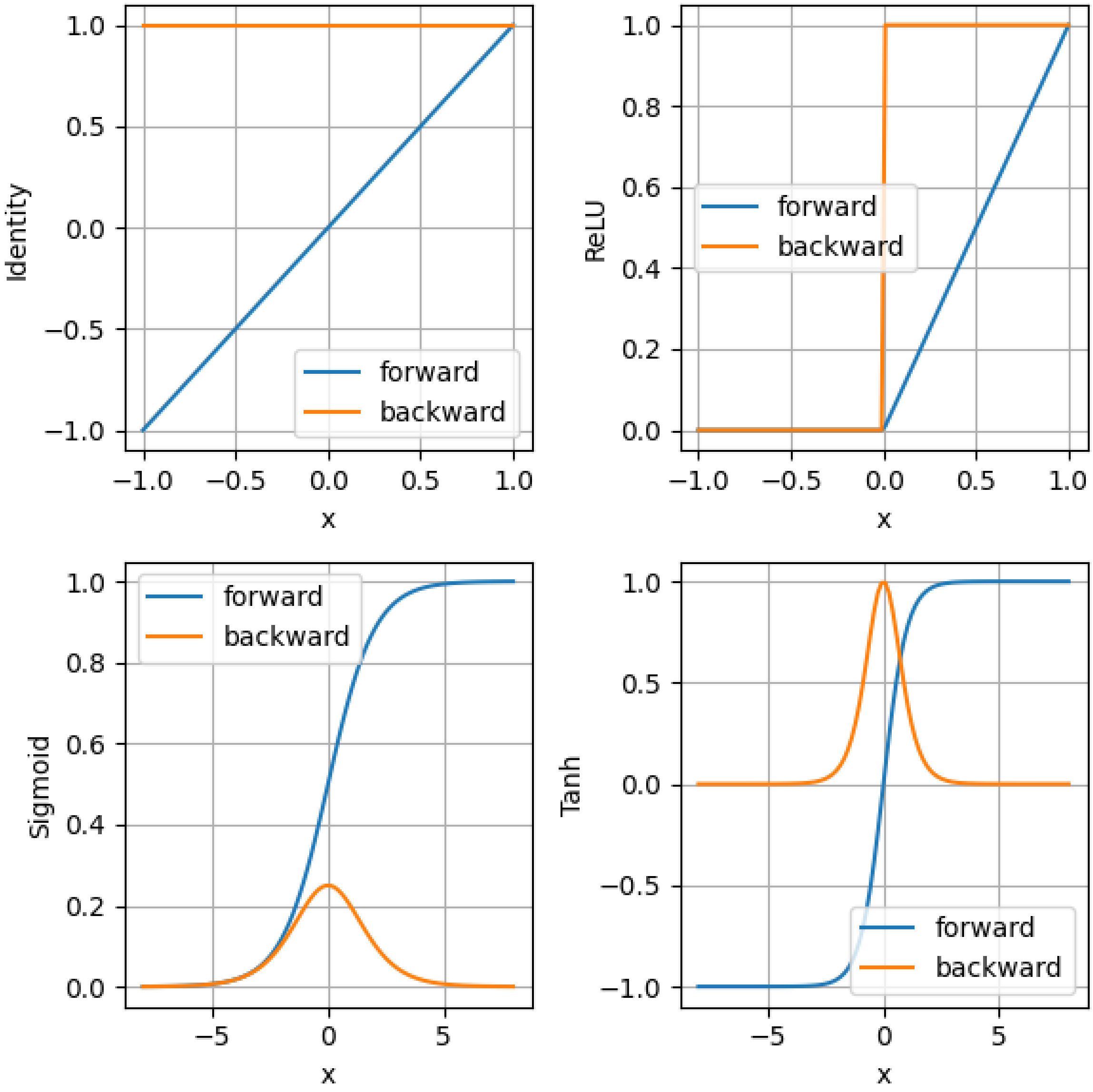
In [ ]:

```
%load_ext autoreload
%autoreload 2

import numpy as np
import matplotlib.pyplot as plt
from torch import nn
import torch
from d2l import torch as d2l

from mytorch import nn as mynn
from mytorch.optim import SGD
```

In [ ]:

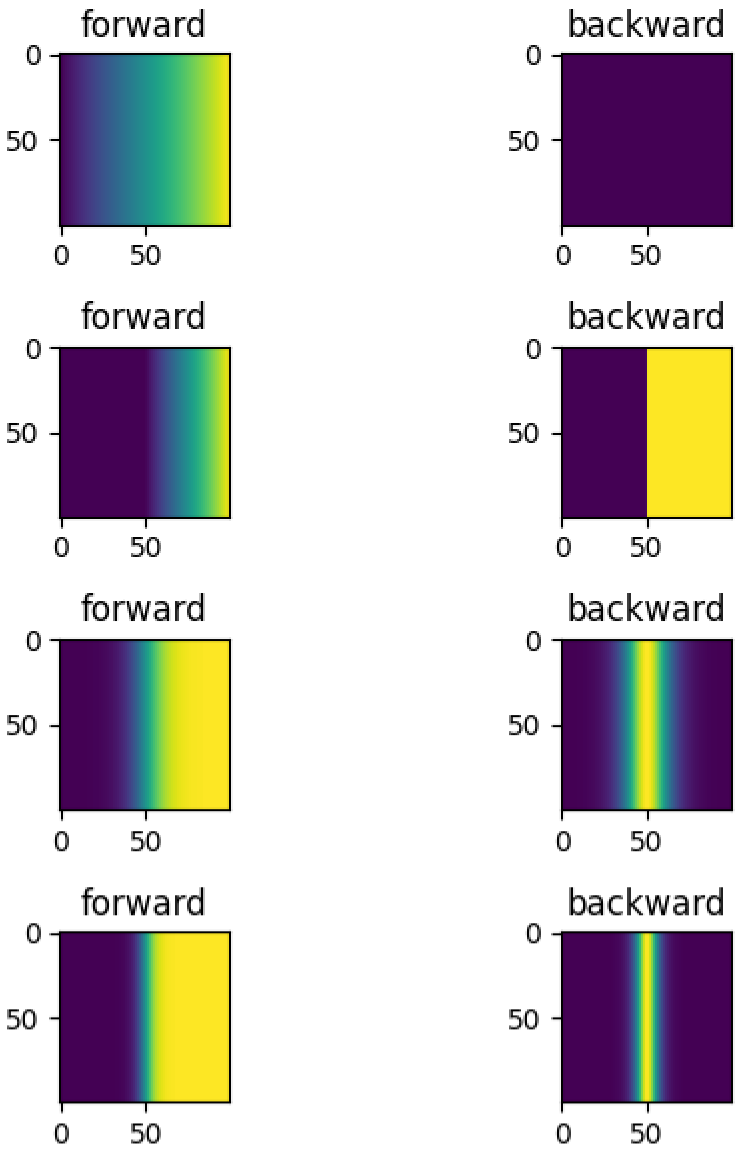
```
# test activation forward/backward in one dimension
fig, ax = plt.subplots(2, 2, figsize=(6, 6))

X = np.linspace(-1, 1, 100)
act = mynn.Identity()
ax[0, 0].plot(X, act.forward(X), label='forward')
ax[0, 0].plot(X, act.backward(), label='backward')
ax[0, 0].grid()
ax[0, 0].set_xlabel('x')
ax[0, 0].set_ylabel('Identity')
ax[0, 0].legend()

act = mynn.ReLU()
ax[0, 1].plot(X, act.forward(X), label='forward')
ax[0, 1].plot(X, act.backward(), label='backward')
ax[0, 1].grid()
ax[0, 1].set_xlabel('x')
ax[0, 1].set_ylabel('ReLU')
ax[0, 1].legend()

X = np.linspace(-8, 8, 100)
act = mynn.Sigmoid()
ax[1, 0].plot(X, act.forward(X), label='forward')
ax[1, 0].plot(X, act.backward(), label='backward')
ax[1, 0].grid()
ax[1, 0].set_xlabel('x')
ax[1, 0].set_ylabel('Sigmoid')
ax[1, 0].legend()

act = mynn.Tanh()
ax[1, 1].plot(X, act.forward(X), label='forward')
ax[1, 1].plot(X, act.backward(), label='backward')
ax[1, 1].grid()
ax[1, 1].set_xlabel('x')
ax[1, 1].set_ylabel('Tanh')
ax[1, 1].legend()

fig.tight_layout()
```



In [ ]:

```python
# test activation forward/backward in two dimensions
fig, ax = plt.subplots(4, 2, figsize=(6, 6))

X = np.outer(np.ones(100), np.linspace(-1, 1, 100))
act = mynn.Identity()
ax[0, 0].imshow(act.forward(X))
ax[0, 1].imshow(act.backward())
ax[0, 0].set_title('forward')
ax[0, 1].set_title('backward')

act = mynn.ReLU()
ax[1, 0].imshow(act.forward(X))
ax[1, 1].imshow(act.backward())
ax[1, 0].set_title('forward')
ax[1, 1].set_title('backward')

X = np.outer(np.ones(100), np.linspace(-8, 8, 100))
act = mynn.Sigmoid()
ax[2, 0].imshow(act.forward(X))
ax[2, 1].imshow(act.backward())
```

```
ax[2, 0].set_title('forward')
ax[2, 1].set_title('backward')

act = mynn.Tanh()
ax[3, 0].imshow(act.forward(X))
ax[3, 1].imshow(act.backward())
ax[3, 0].set_title('forward')
ax[3, 1].set_title('backward')

fig.tight_layout()
```



```
In [ ]:
```