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

Input signal

```
x = np.array([1, 2, 3, 4], dtype=float)
```

Filter: First-order difference

```
h = np.array([1, -1])
```

Convolution

```
y = np.convolve(x, h, mode='full')
```

Time axes

```
n_x = np.arange(len(x))
n_y = np.arange(len(y))
```

Plot

```
plt.stem(n_x, x, basefmt=" ", linefmt='b-', markerfmt='bo', label='Input x[n]')
plt.stem(n_y, y, basefmt=" ", linefmt='r-', markerfmt='ro', label='Output y[n] = x[n] * h[n]')
plt.title("Convolution with [1, -1] (Difference Filter)")
plt.xlabel("n")
plt.ylabel("Amplitude")
plt.legend()
plt.grid(True)
```

plt.show()

print("Output y[n] =", y)
y = [1. 1. 1. 1. -4.]

`n`	Calculation	Result
0	`x[0]*1`	**1**
1	`x[1]*1 + x[0]*(-1)` = 2 − 1	**1**
2	`x[2]*1 + x[1]*(-1)` = 3 − 2	**1**
3	`x[3]*1 + x[2]*(-1)` = 4 − 3	**1**
4	`x[3]*(-1)` = −4	**−4**

What Does It Mean Physically?
The output gives the difference between consecutive values:

2 − 1 = 1

3 − 2 = 1

4 − 3 = 1

It highlights how fast the signal changes.

The last sample (−4) is just the tail effect due to filter length — it has no more valid points to compare against.

Use	Why This Filter?
Change detection	Spots rising/falling edges in a signal
Activity detection	Detects start of transmission (burst IQ)
Basic modulation	Helps isolate transitions in BPSK/QPSK
Packet detection	When preambles have sharp changes