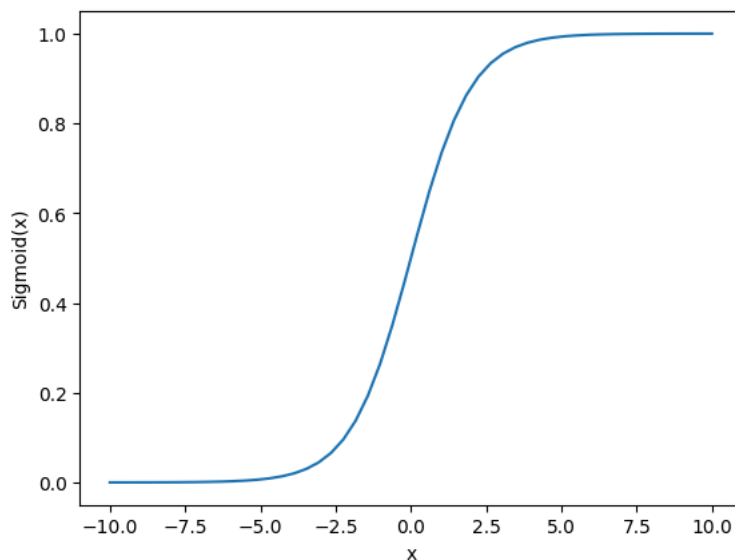


```
#Sigmoid activation

import numpy as np
def sig(x):
    return 1/(1 + np.exp(-x))
x = 1.0
print('Applying Sigmoid Activation on (%.1f) gives %.1f' % (x, sig(x)))
x = -10.0
print('Applying Sigmoid Activation on (%.1f) gives %.1f' % (x, sig(x)))
x = 0.0
print('Applying Sigmoid Activation on (%.1f) gives %.1f' % (x, sig(x)))
x = 15.0
print('Applying Sigmoid Activation on (%.1f) gives %.1f' % (x, sig(x)))
x = -2.0
print('Applying Sigmoid Activation on (%.1f) gives %.1f' % (x, sig(x)))

import matplotlib.pyplot as plt
x = np.linspace(-10, 10, 50)
p = sig(x)
plt.xlabel("x")
plt.ylabel("Sigmoid(x)")
plt.plot(x, p)
plt.show()
```

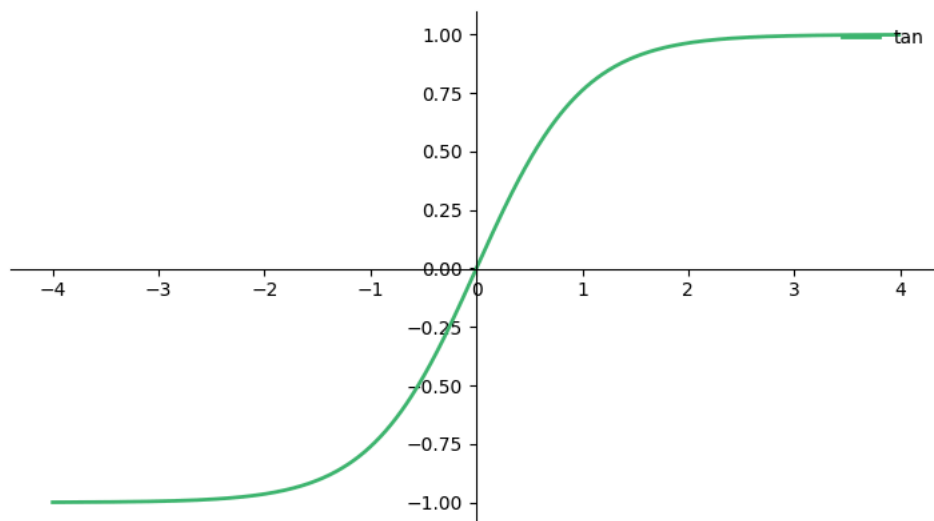
Applying Sigmoid Activation on (1.0) gives 0.7
 Applying Sigmoid Activation on (-10.0) gives 0.0
 Applying Sigmoid Activation on (0.0) gives 0.5
 Applying Sigmoid Activation on (15.0) gives 1.0
 Applying Sigmoid Activation on (-2.0) gives 0.1



```
# tan activation function

import matplotlib.pyplot as plt
import numpy as np
def tanh(x):
    t=(np.exp(x)-np.exp(-x))/(np.exp(x)+np.exp(-x))
    dt=1-t**2
    return t,dt
z=np.arange(-4,4,0.01)
tanh(z)[0].size,tanh(z)[1].size
fig, ax = plt.subplots(figsize=(9, 5))

ax.spines['left'].set_position('center')
ax.spines['bottom'].set_position('center')
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.yaxis.set_ticks_position('left')
ax.plot(z,tanh(z)[0], color="#40b770", linewidth=2, label="tan")
ax.legend(loc="upper right", frameon=False)
fig.show()
```



#tanh or Hyperbolic:

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

```
def tanh(x):
    t=(np.exp(x)-np.exp(-x))/(np.exp(x)+np.exp(-x))
    dt=1-t**2
    return t,dt
z=np.arange(-4,4,0.01)
tanh(z)[0].size,tanh(z)[1].size
```

Setup centered axes

```
fig, ax = plt.subplots(figsize=(9, 5))
ax.spines['left'].set_position('center')
ax.spines['bottom'].set_position('center')
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.yaxis.set_ticks_position('left')
```

Create and show plot

```
ax.plot(z,tanh(z)[0], color="#307EC7", linewidth=3, label="tanh")
ax.plot(z,tanh(z)[1], color="#9621E2", linewidth=3, label="derivative")
ax.legend(loc="upper right", frameon=False)
fig.show()
```

