# Assignment - 2

## Name - prekshita patil

## Reg.No - 20MAI0073

## Link: https://github.com/prekshita19/DL-Assignments (https://github.com/prekshita19/DL-Assignments)

Que 1 :- Apply all activation Function

In [1]:
```
#Import all the library use in the activation function
import matplotlib.pyplot as plt
import numpy as np
```

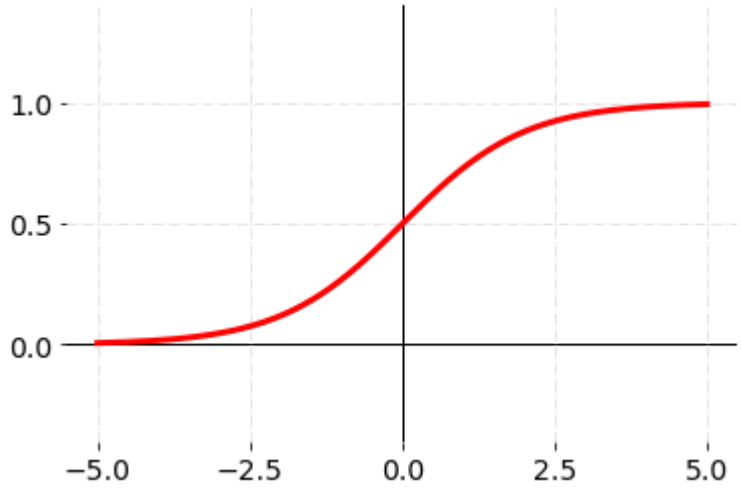
In [2]:
```
#Give the range to the X-axis
x = np.arange(-5, 5, 0.01)
```

In [3]:
```
def plot(func, yaxis=(-1.4, 1.4)):
    plt.ylim(yaxis)
    plt.locator_params(nbins=5)
    plt.xticks(fontsize = 14)
    plt.yticks(fontsize = 14)
    plt.axhline(lw=1, c='black')
    plt.axvline(lw=1, c='black')
    plt.grid(alpha=0.4, ls='-.')
    plt.box(on=None)
    plt.plot(x, func(x), c='r', lw=3)
```

# Sigmoid

In [4]:
```
def sigmoid(x):
    return 1 / (1 + np.exp(-x))
```

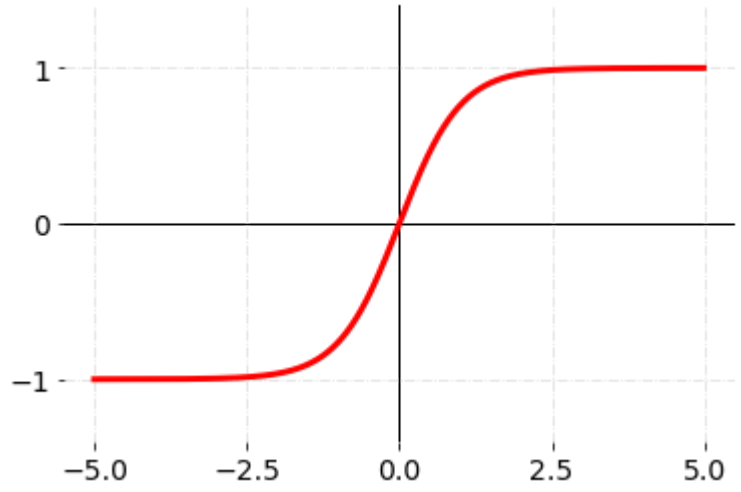
In [5]:
```
plot(sigmoid, yaxis=(-0.4, 1.4))
```




# Tan h

In [6]:
```
def tanh(x):
    return 2 / (1 + np.exp(-2 * x)) -1
```

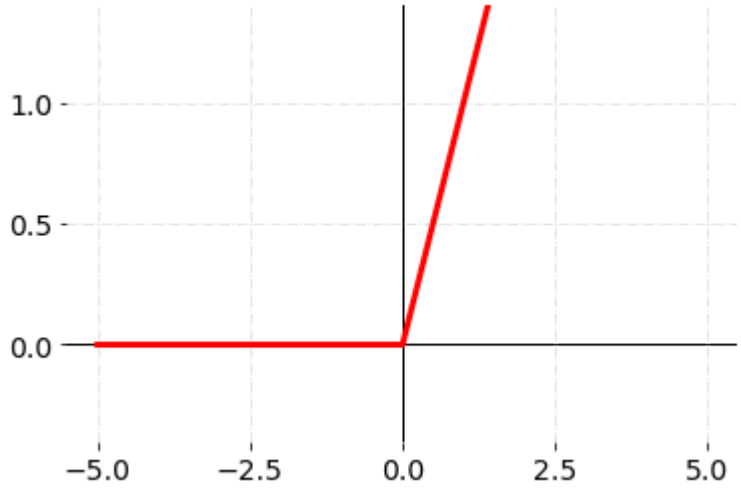
In [7]:
```
plot(tanh)
```




# Rectified Linear Units (ReLU)

In [8]:
```
relu = np.vectorize(lambda x: x if x > 0 else 0, otypes=[np.float])
```

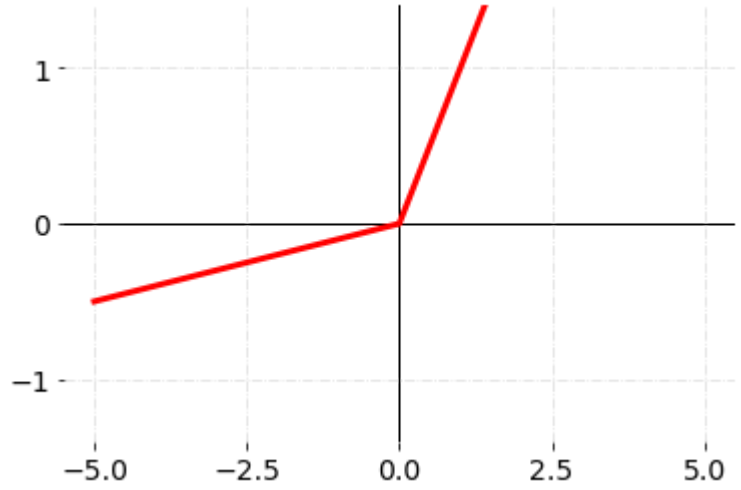
In [9]:
```
plot(relu, yaxis=(-0.4, 1.4))
```




# Leaky Rectified Linear Units (Leaky ReLU)

In [10]:
```
leaky_relu = np.vectorize(lambda x: max(0.1 * x, x), otypes=[np.float])
```

In [11]:
```
plot(leaky_relu)
```




# Exponential Linear Units (ELU)

In [12]:
```
elu = np.vectorize(lambda x: x if x > 0 else 0.5 * (np.exp(x) - 1), otypes=[np.fl

plot(elu)
```