Sigmoid Function: Transforming Data into Predictions

In the vast landscape of data science, understanding the intricacies of non-linear relationships is paramount for building accurate predictive models and can uncover hidden insights within data. One such function that plays a pivotal role, particularly in classification tasks, is the sigmoid function.

What is the sigmoid function?

The sigmoid function, also known as logistic function, is a mathematical function that maps any real number between -∞ and +∞ to a value between 0 and 1. Mathematically it is computed with the formula,

$$f(z) = \frac{1}{1+e^{-z}} \tag{1}$$

where,

- z is the input to the sigmoid function)
- e - Euler's number (Mathematical constant)- 2.781

Properties:

- Range: The output of the sigmoid function always lies between 0 and 1, making it suitable for modeling probabilities.
- Symmetry: The sigmoid function is symmetric around its midpoint (0.5), meaning that f(z)=0.5 when 'z' is 0.
- Differentiability: The sigmoid function is smooth and continuously differentiable, which is crucial for gradient-based optimization algorithms used in machine learning.
- Monotonicity: The sigmoid function is strictly increasing, ensuring that as 'z' increases f(z) moves closer to 1 and as 'z' decreases f(z) moves closer to o.

Python Implementation:

Lets create a function to mathematically calculate sigmoid.

In [9]:

```
def sigmoid(x):
    return 1 / (1 + np.exp(-x))
```

Creating 100 data point between -10 and +10 and reviewing the inputs with the corresponding Sigmoid values

In [27]:

```
import numpy as np
import pandas as pd

#create data
x = np.linspace(-10, 10, 100)
```

```python
#Computing (function call)
y = sigmoid(x)

#Dataframe for the data to make it easily viewable in a table:
df = pd.DataFrame({"x": x, "sigmoid(x)": y})
df
```

Out[27]:

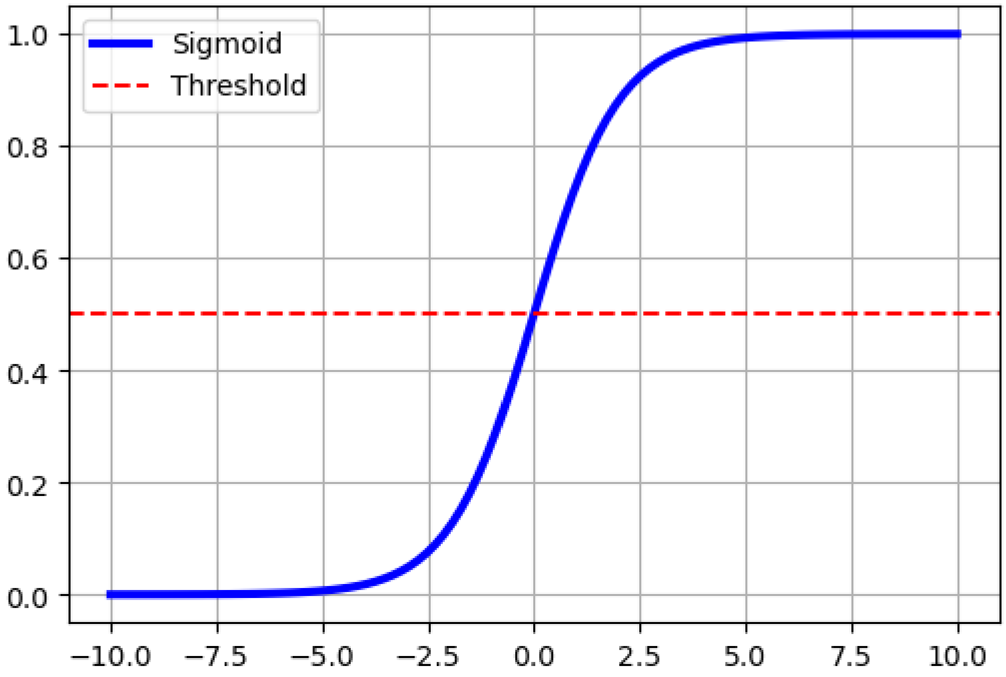
	x	sigmoid(x)
0	-10.000000	0.000045
1	-9.797980	0.000056
2	-9.595960	0.000068
3	-9.393939	0.000083
4	-9.191919	0.000102
...	...	...
95	9.191919	0.999898
96	9.393939	0.999917
97	9.595960	0.999932
98	9.797980	0.999944
99	10.000000	0.999955

100 rows × 2 columns

Let's visualize this data. By plotting the sigmoid function, we get the familiar S-curve:

In [30]:

```python
import matplotlib.pyplot as plt
plt.figure(figsize=(6,4))
plt.plot(x, y, c="blue", linewidth=3.0, label="Sigmoid")
plt.axhline(y=0.5, color='r', linestyle='--', label='Threshold')
plt.legend()
plt.grid(True)
plt.show()
```



Applications:

- In logistic regression, the sigmoid function is used to model the probability that a given input belongs to a particular class.
- In artificial neural networks, sigmoid function is used as an activation function in hidden layers to introduce non-linearity into the model.
- Sigmoid function is used in probabilistic models to represent conditional probabilities or to model cumulative distribution functions.

Conclusion:

- The sigmoid function is characterized by its S-shaped curve, which smoothly transitions from 0 to 1 as the input variable changes. This shape allows the function to capture non-linear relationships.

- The sigmoid function can be interpreted as representing probabilities. This makes it useful in binary classification tasks, where the output of the model needs to indicate the likelihood of a given input belonging to a particular class.

Thank You!