

Logistic regression is a statistical model used primarily for binary classification tasks, where the outcome variable is categorical and has two possible outcomes. Unlike linear regression, which predicts continuous values, logistic regression predicts the probability of an observation belonging to a particular class. The model achieves this by applying a logistic function to the linear combination of the predictor variables, transforming the output into a probability score between 0 and 1. This probability score represents the likelihood of the observation belonging to the positive class. Logistic regression estimates the parameters of the linear equation during training, maximizing the likelihood of the observed data. During prediction, a decision boundary is applied to classify observations into one of the two classes based on their predicted probabilities. Logistic regression is widely used in various fields, including medicine, finance, and marketing, owing to its simplicity, interpretability, and effectiveness in binary classification tasks.

Study Hours	Pass (1) / Fail (0)
2	0
3	0
4	0
5	0
6	0
7	0
8	1
9	1
10	1
11	1
12	1
13	1
14	1
15	1
16	1

```
from sklearn.linear_model import LogisticRegression
```

```
X = [[2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16]]
```

```
y = [0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1]
```

```
model = LogisticRegression()
```

```
model.fit(X, y)
```

```
predicted_probability = model.predict_proba([[4]])[:, 1]
```

```
print("Probability of passing:", predicted_probability[0])
```