**Use case**

**Step1:#import required datasets**

**from sklearn.datasets import make\_classification**

**import matplotlib.pyplot as plt**

**from sklearn.linear\_model import LogisticRegression**

**from sklearn.model\_selection import train\_test\_split**

**from sklearn.metrics import confusion\_matrix**

**import pandas as pd**

**step2:**

**# Train-80%,Test-20%, Train-70%,Test-30%**

**# Required datasets**

**from sklearn.datasets import make\_classification**

**x,y= make\_classification(**

**n\_samples=100,**

**n\_features=1,**

**n\_classes=2,**

**n\_clusters\_per\_class=1,**

**flip\_y=0.03,**

**n\_informative=1,**

**n\_redundant=0**

**)**

**print(x)**

**print(y)**

**step3:**

**# c-color,cmap-colormap**

**plt.scatter(x,y,c=x,cmap='rainbow')**

**plt.title('scatter plot of logistic regression')**

**plt.show()**

**step4:**

**#c-color,cmap-colormap,Logistic Regression**

**plt.scatter(x,y,c=y,cmap='rainbow')**

**plt.title('scatter plot of logistic regression')**

**plt.show()**

**step5:**

**# machine learning**

**from sklearn.linear\_model import LogisticRegression**

**log\_reg = LogisticRegression()**

**log\_reg.fit(x\_train,y\_train)**

**step6:**

**y\_pred=log\_reg.predict(x\_test)**

**confusion\_matrix(y\_test,y\_pred)**