```
actual=[1,0,0,1,0,0,1,0,0,1,1,0,0,0,0,1,0,1,0,1]
print("actual values=", actual)
prediction_prob=[0.886,0.375,0.174,0.817,0.574,0.319,0.812,0.314,0.098,0.741,0.847,0.202,0.31,0.073,0.179,0.917,0.64,0.388,0.116,0.72]

# (i) Write a function from scratch called predict() that computes the final predictions to be output by the model and returns them as a list
def predict(prediction_prob, threshold):
    predicted=list()
    for probability in prediction_prob:
        if probability<=threshold: predicted.append(0)
        else: predicted.append(1)
    print("predicted values=",predicted)
    return predicted


# (ii) Write a function from scratch called acc_score() that calculates the model accuracy score using the true labels as compared to the prediction
def acc_score(actual, predicted):
    correct=0
    for i in range(len(actual)):
        if actual[i]==predicted[i]: correct+=1
    return correct/len(actual)


predicted=predict(prediction_prob, 0.5)
accuracy0=acc_score(actual, predicted)

# (iii) Get the accuracy from sklearn then compare accuracy with output from the function acc_score()
from sklearn.metrics import accuracy_score
accuracy1=accuracy_score(actual, predicted)

print('Accuracy calculated by acc_score= ', accuracy0)
print('Accuracy calculated by sklearn library function accuracy_scoree= ', accuracy1)
```

```
actual values= [1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1]
predicted values= [1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1]
Accuracy calculated by acc_score=  0.85
Accuracy calculated by sklearn library function accuracy_scoree=  0.85
```

+ Code + Text

LOGISTIC REGRESSION

Colab paid products - Cancel contracts here

✓ 0s completed at 9:15 PM