EX.NO:7

DATE:

Implement accuracy metrics like Receiver Operated Characteristic curves.

Aim:

To implement accuracy metrics like Receiver Operated Characteristic curves.

ALGORITHM:

1. Import the necessary libraries
2. Generate the synthetic data
3. Split the data into training and testing datasets
4. Train a logistic regression model
5. Predict probabilities for the positive class
6. Calculate ROC Curve
7. Calculate the AUC curve for the ROC curve
8. Plot the ROC curve

PROGRAM:

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.datasets import make_classification
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import roc_curve, auc
```

 # Generate synthetic data for binary classification

```
X, y = make_classification(n_samples=1000, n_features=20, n_classes=2, random_state=42)
```

Split the data into training and testing sets

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

 # Train a logistic regression model

```
model = LogisticRegression()
model.fit(X_train, y_train)
```

 # Predict probabilities for the positive class

```
y_probs = model.predict_proba(X_test)[:, 1]
```

 # Calculate the ROC curve

```
fpr, tpr, thresholds = roc_curve(y_test, y_probs)
```

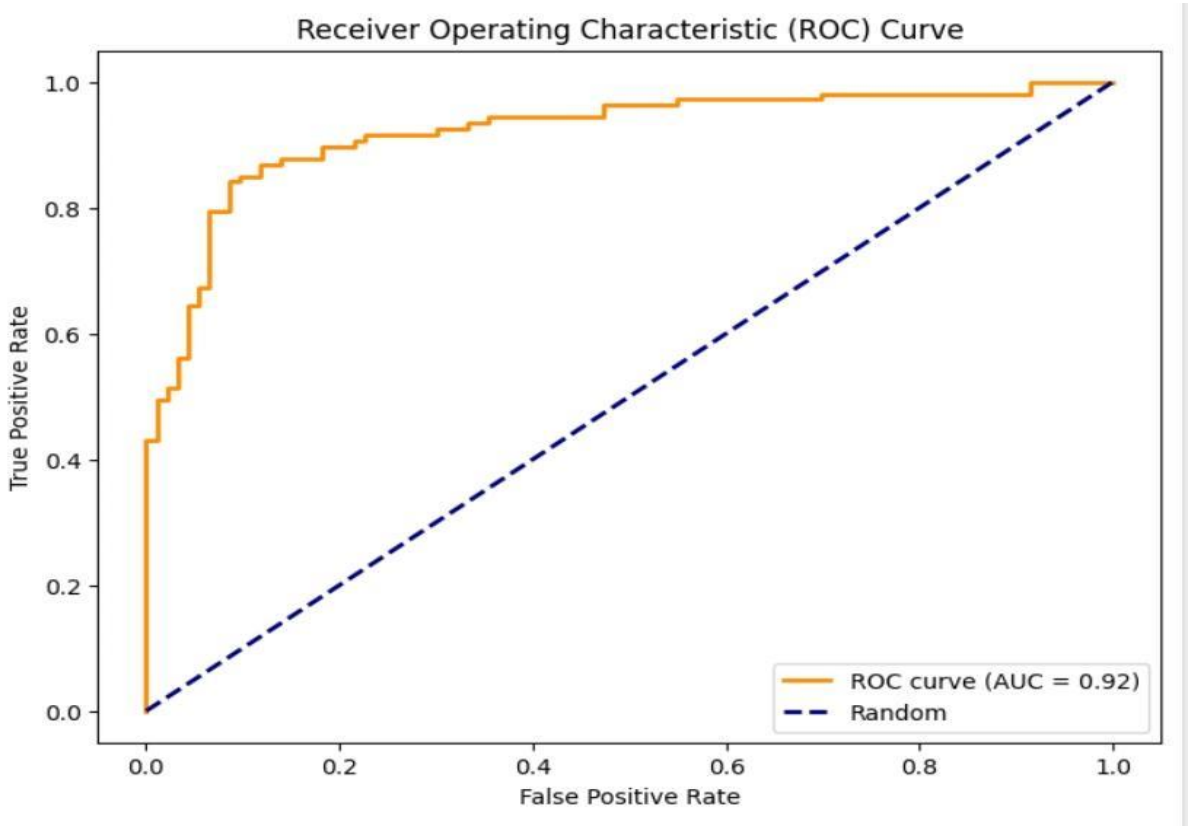
 # Calculate the Area Under the Curve (AUC) for the ROC curve

```
roc_auc = auc(fpr, tpr)
```

 # Plot the ROC curve

```
plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC curve (AUC = {roc_auc:.2f})')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--', label='Random')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve')
plt.legend(loc='lower right')
plt.show()
```

OUTPUT:



Result:

Thus the Hadoop one cluster was installed and simple applications executed successfully.