In [3]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import mean_squared_error, r2_score,accuracy_score,c
from sklearn.impute import SimpleImputer
```

In [ ]:
```
df =pd.read_csv('train.csv')
```

In [ ]:
```
age_imputer = SimpleImputer(strategy='mean')
df['age'] = age_imputer.fit_transform(df[['age']])
```

In [ ]:
```
df = df.drop(['name','cabin','ticket'], axis = 1)
df['sex'] = df['sex'].map({'male':1, 'female':0})
df['embarked'] = df['embarked'].map({'Q':0, 'S': 1,'C':2})
```

In [ ]:
```
df.dropna(inplace = True)
```

In [ ]:
```
df
```

In [ ]:
```

```

In [ ]:
```
Y = df['survived']
X = df.drop('survived', axis = 1)

x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size = 0.3
```

In [ ]:
```
# Create a logistic regression model
logistic_model = LogisticRegression(max_iter  =1000)

# Train the model on the training data
logistic_model.fit(x_train, y_train)
```

In [ ]:
```
# Predict the 'survived' values for the test set
y_pred = logistic_model.predict(x_test)
```

In [124…
```
# Calculate accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy: {accuracy:.2f}')

# Generate classification report
print(classification_report(y_test, y_pred))

# Generate confusion matrix
conf_matrix = confusion_matrix(y_test, y_pred)
print('Confusion Matrix:')
print(conf_matrix)
```

```
Accuracy: 0.78
              precision    recall  f1-score   support

           0       0.81      0.87      0.84       143
           1       0.71      0.59      0.65        74

    accuracy                           0.78       217
   macro avg       0.76      0.73      0.74       217
weighted avg       0.77      0.78      0.77       217

Confusion Matrix:
[[125  18]
 [ 30  44]]
```

In [ ]:

In [ ]:

In [ ]: