In [15]:

```
import pandas as pd
import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.metrics import roc_auc_score
import numpy as np
```

In [12]:

```
# Data of Anjani, Bhargavi and Surada for training the model
df_features = pd.read_csv("H:/mastersProject/activity_analyzer/LogisticRegress
ion/Data/featuresfile.csv")
X_train_data = df_features.values[:, 1:44]
y_train_data = df_features.values[:, 44]
usersList = set(df_features.values[:,0])
print("Users")
print(usersList)
# Data of 12 people for testing the model
df_features_test = pd.read_csv("H:/mastersProject/activity_analyzer/LogisticRe
gression/Data/featuresfile_10.csv")
X_test = df_features_test.values[:, 2:45]
y_test = df_features_test.values[:, 45]
usersList = set(df_features_test.values[:,0])
print("Users")
print(usersList)
```

```
Users
{'Anjani', 'Surada', 'Bhargavi'}
Users
{'ahsu6', 'achan', 'jding11', 'mbaybay', 'rpshah2', 'pgarg', 'Chai', 'bbalasu
bramanian', 'npbandal', 'sbgavade', 'ptiwari3', 'cjain'}
```

In [17]:

```
# Fitting the logistic regression model
lr = LogisticRegression(C=100.0, random_state=1)
lr.fit(X_train_data, y_train_data)
predict = lr.predict(X_test)
logisticRegScore = lr.score(X_test, y_test)
print("Importance of features")
print(lr.coef_)
print("Logistic regression Score")
print(logisticRegScore)
```

```
Importance of features
[[-0.04428713 -0.10980405 -0.09281945 -0.02975576 -0.07048953 -0.04791663
   0.19319413  0.06625453 -0.28487191 -0.14588925 -0.05611415 -0.3902119
  -0.53819686 -0.44442756 -0.01878854 -0.14648825  0.32435065  0.38147093
   0.15674629  0.01363013 -0.05527147 -0.04656041  0.41339789  0.14532732
   0.16133763 -0.29539818 -0.43668888 -0.14776003 -0.07336433 -0.18017453
   0.09129413  0.48563653  0.35681367 -4.40927205 -3.82667241 -1.13283029
   1.60704672  1.12371103 -3.96590168 -4.86073903 -4.23544345 -0.60632588
   0.36074301]]
Logistic regression Score
0.868725868726
```

In [18]:

```
# Confusion matrix analysis
cm = confusion_matrix(y_test, predict, labels=["walking", "running"])
print(cm)
```

```
[[213  47]
 [ 21 237]]
```