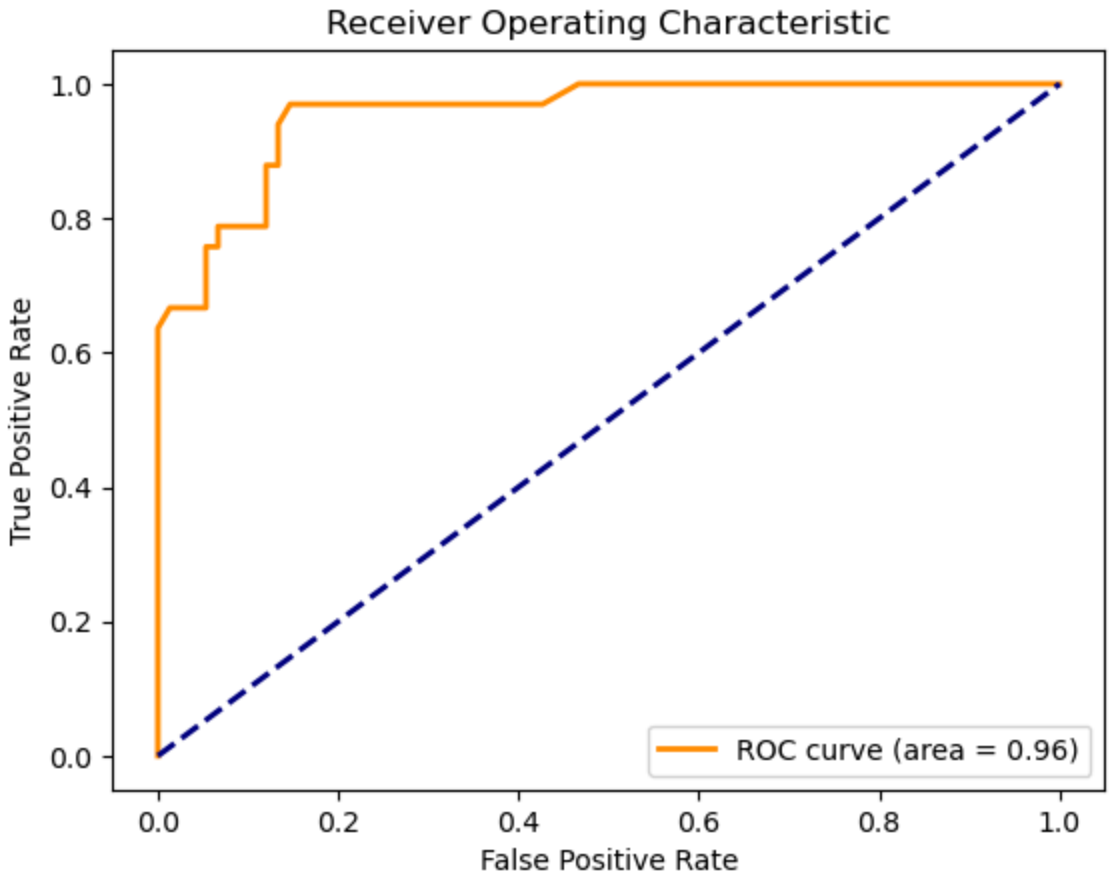
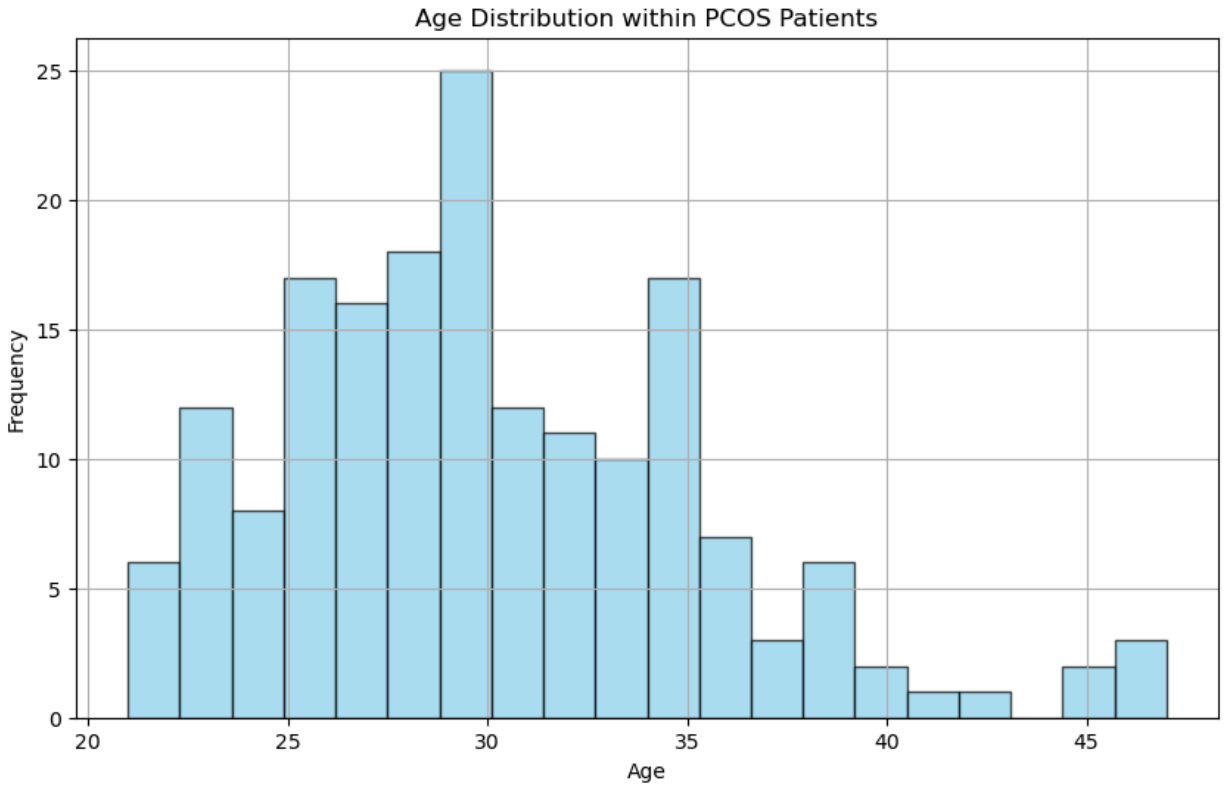
In [3]:
```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import roc_curve, auc, classification_report, confusion_ma
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.impute import SimpleImputer
import pandas as pd


data = pd.read_csv('/Users/mehtap/Downloads/PCOS_data.csv')


imputer = SimpleImputer(strategy='mean')
data_imputed = imputer.fit_transform(data.iloc[:, 3:-1])
X = data_imputed
y = data['PCOS (Y/N)']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, rando

rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)

rf_classifier.fit(X_train, y_train)

y_pred = rf_classifier.predict(X_test)

accuracy = np.mean(y_test == y_pred)

conf_matrix = confusion_matrix(y_test, y_pred)
print("Confusion Matrix:")
print(conf_matrix)

class_report = classification_report(y_test, y_pred)
print("Classification Report:")
print(class_report)

y_scores = rf_classifier.predict_proba(X_test)[:, 1]

roc_auc = roc_auc_score(y_test, y_scores)
print("ROC AUC:", roc_auc)

fpr, tpr, _ = roc_curve(y_test, y_scores)


plt.figure()
plt.plot(fpr, tpr, color='darkorange', lw=2, label='ROC curve (area = %0.2f)' %
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic')
plt.legend(loc="lower right")
plt.show()
```

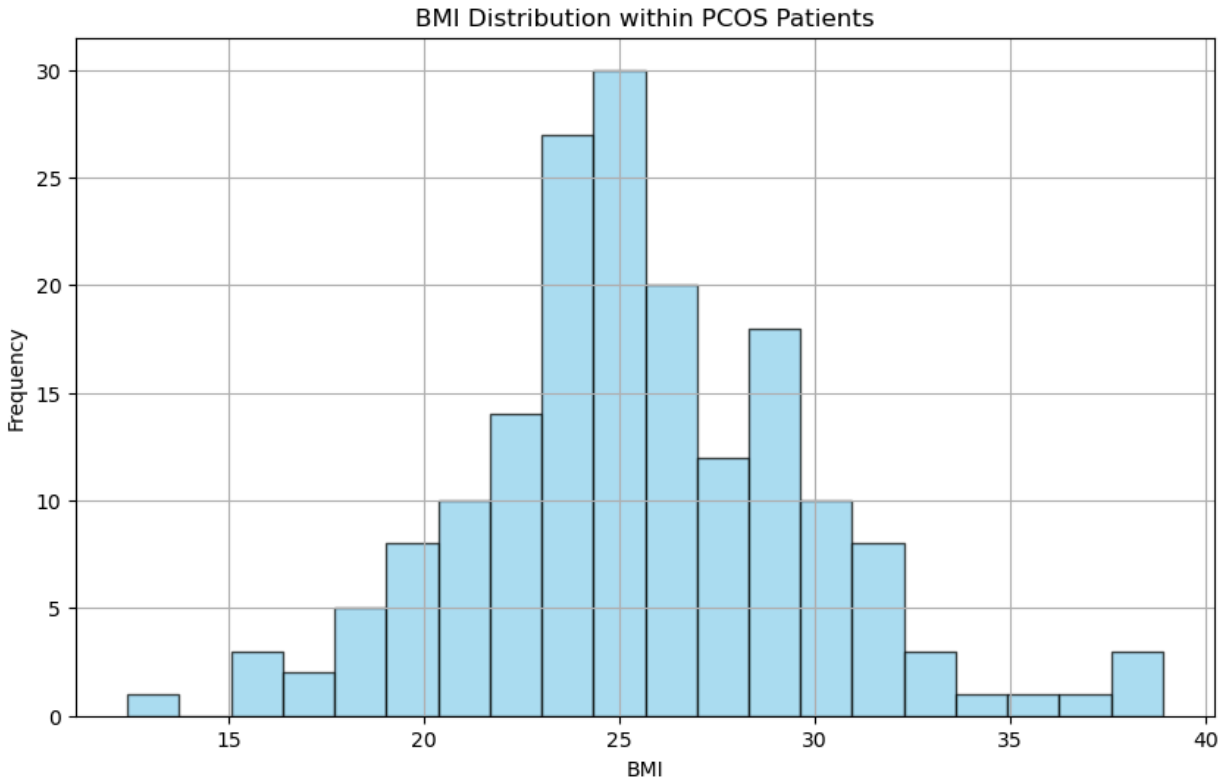
```
Confusion Matrix:
[[71  4]
 [ 8 25]]
Classification Report:
              precision    recall  f1-score   support

           0       0.90      0.95      0.92        75
           1       0.86      0.76      0.81        33

    accuracy                           0.89       108
   macro avg       0.88      0.85      0.86       108
weighted avg       0.89      0.89      0.89       108

ROC AUC: 0.9561616161616162
```



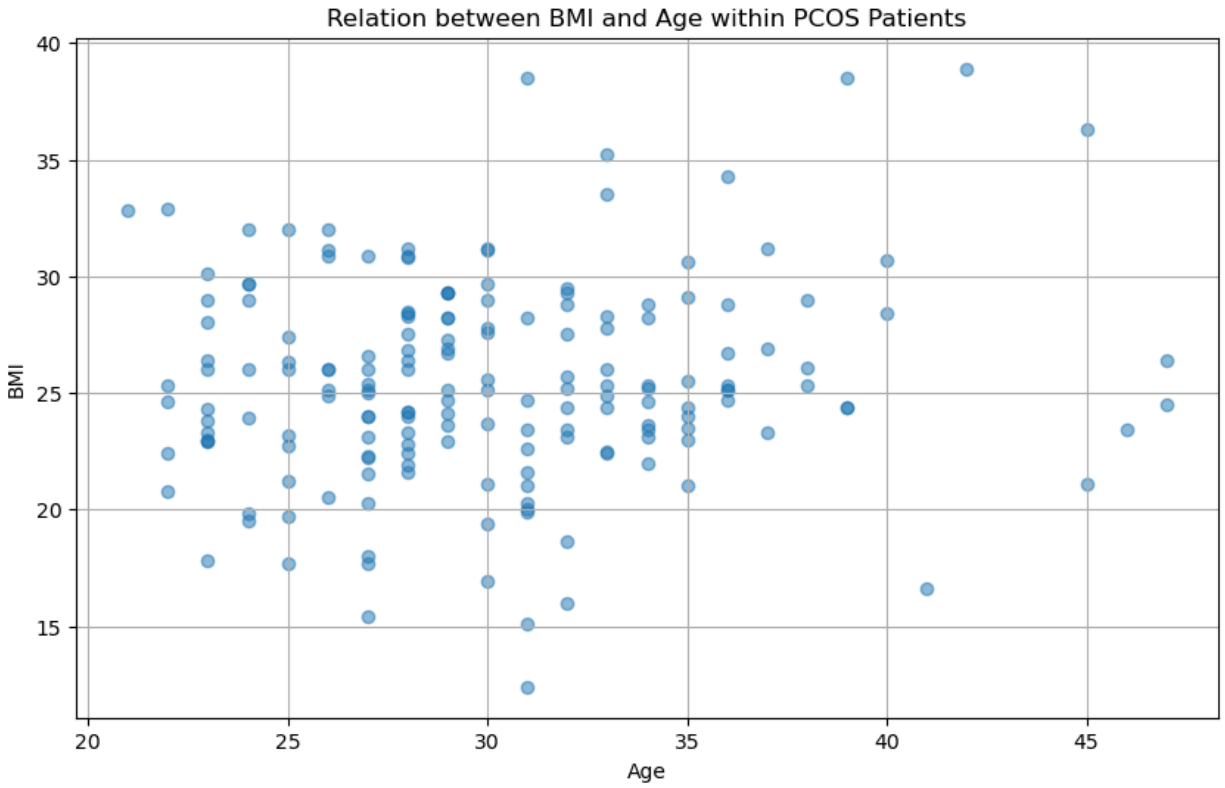
In [6]:
```
pcos_data = data[data['PCOS (Y/N)'] == 1]

plt.figure(figsize=(10, 6))
plt.hist(pcos_data.iloc[:, 3], bins=20, color='skyblue', edgecolor='black', al
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.title('Age Distribution within PCOS Patients')
plt.grid(True)
plt.show()
```



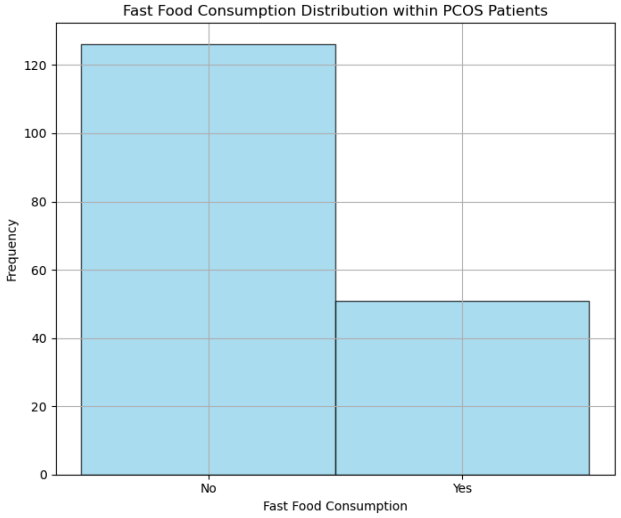
In [10]:
```
pcos_data = data[data['PCOS (Y/N)'] == 1]

plt.figure(figsize=(10, 6))
plt.hist(pcos_data.iloc[:, 6], bins=20, color='skyblue', edgecolor='black', al
plt.xlabel('BMI')
plt.ylabel('Frequency')
plt.title('BMI Distribution within PCOS Patients')
plt.grid(True)
plt.show()
```

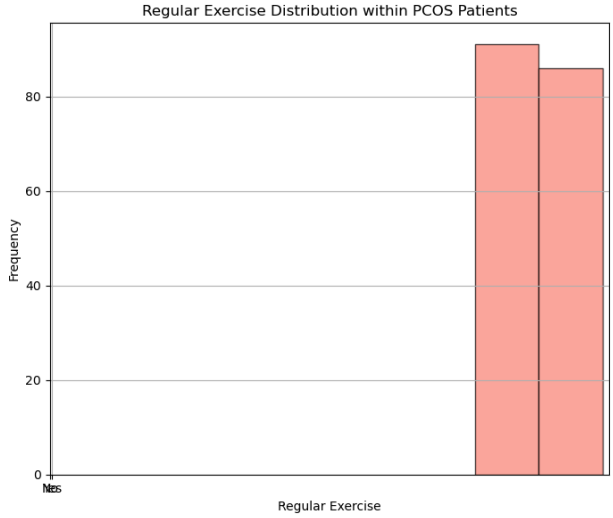


In [11]:
```
pcos_data = data[data['PCOS (Y/N)'] == 1]

plt.figure(figsize=(10, 6))
plt.scatter(pcos_data.iloc[:, 3], pcos_data.iloc[:, 6], alpha=0.5)
plt.xlabel('Age')
plt.ylabel('BMI')
plt.title('Relation between BMI and Age within PCOS Patients')
plt.grid(True)
plt.show()
```



In [12]:
```
pcos_data = data[data['PCOS (Y/N)'] == 1]

plt.figure(figsize=(14, 6))

plt.subplot(1, 2, 1)
plt.hist(pcos_data.iloc[:, 36], bins=2, color='skyblue', edgecolor='black', alp
plt.xlabel('Fast Food Consumption')
plt.ylabel('Frequency')
plt.title('Fast Food Consumption Distribution within PCOS Patients')
plt.xticks([0.25, 0.75], ['No', 'Yes'])
plt.grid(True)

plt.subplot(1, 2, 2)
plt.hist(pcos_data.iloc[:, 37], bins=2, color='salmon', edgecolor='black', alph
plt.xlabel('Regular Exercise')
plt.ylabel('Frequency')
plt.title('Regular Exercise Distribution within PCOS Patients')
plt.xticks([0.25, 0.75], ['No', 'Yes'])
plt.grid(True)

plt.tight_layout()
plt.show()
```



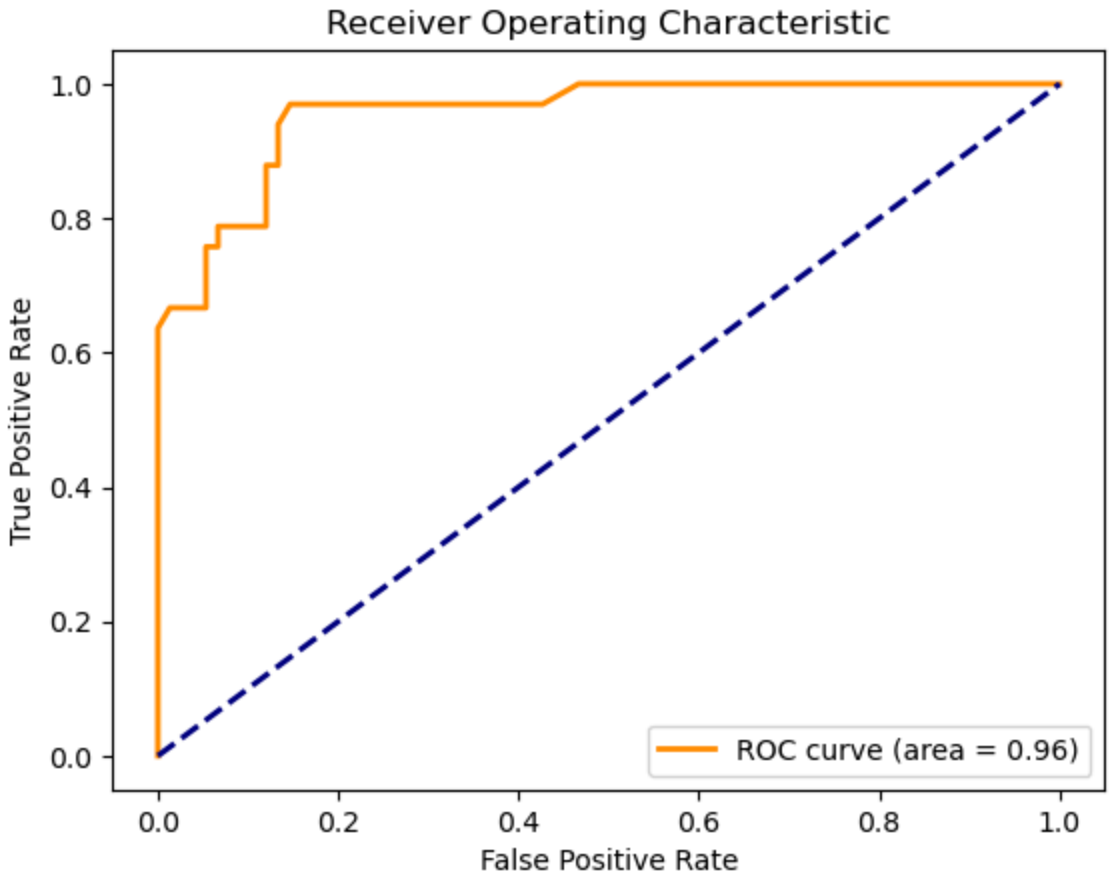


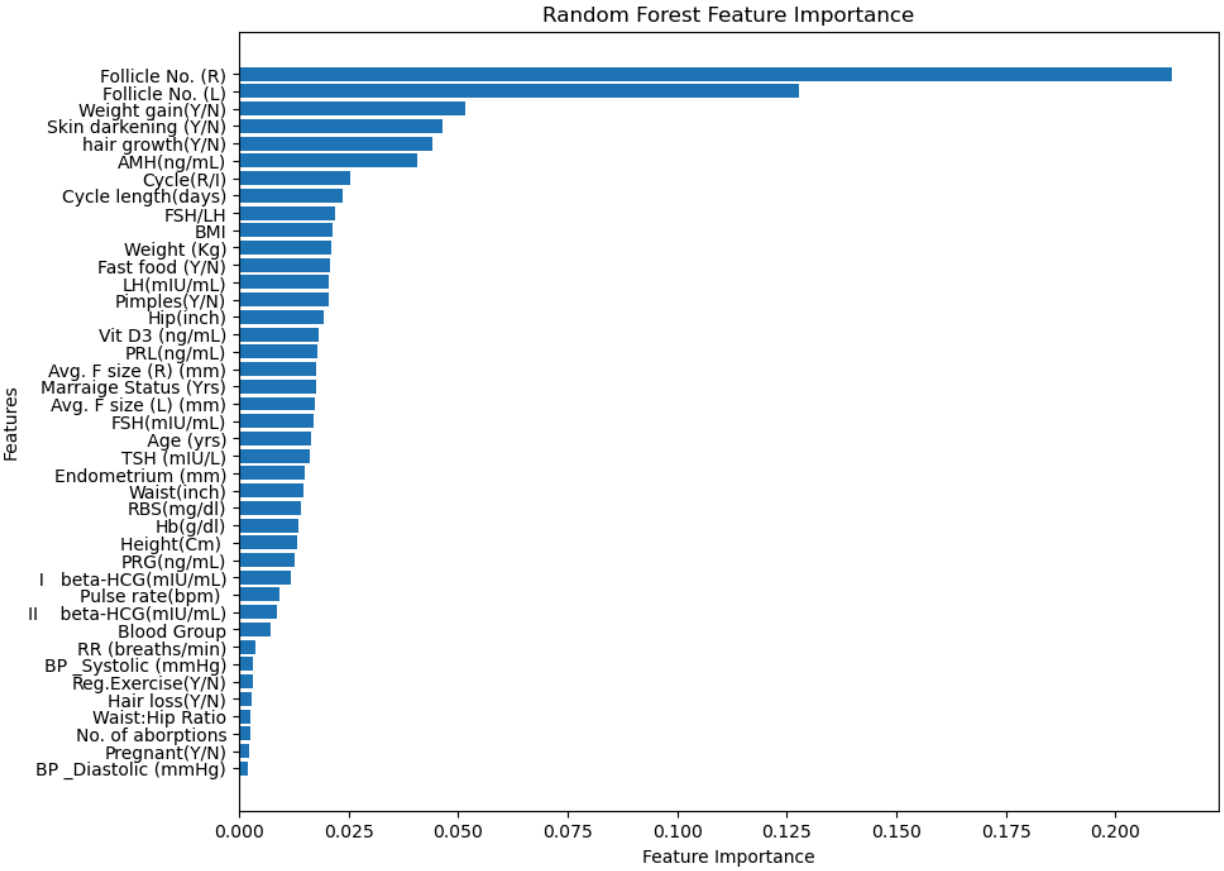
In [1]:

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import roc_curve, auc, classification_report, confusion_ma
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.impute import SimpleImputer
import pandas as pd


data = pd.read_csv('/Users/mehtap/Downloads/PCOS_data.csv')

imputer = SimpleImputer(strategy='mean')
data_imputed = imputer.fit_transform(data.iloc[:, 3:-1])  # Impute missing valu

X = data_imputed
y = data['PCOS (Y/N)']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, rando

rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)

rf_classifier.fit(X_train, y_train)

y_pred = rf_classifier.predict(X_test)

accuracy = np.mean(y_test == y_pred)

conf_matrix = confusion_matrix(y_test, y_pred)
print("Confusion Matrix:")
print(conf_matrix)

class_report = classification_report(y_test, y_pred)
print("Classification Report:")
print(class_report)

y_scores = rf_classifier.predict_proba(X_test)[:, 1]

roc_auc = roc_auc_score(y_test, y_scores)
print("ROC AUC:", roc_auc)

fpr, tpr, _ = roc_curve(y_test, y_scores)
```

```
plt.figure()
plt.plot(fpr, tpr, color='darkorange', lw=2, label='ROC curve (area = %0.2f)' %
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic')
plt.legend(loc="lower right")
plt.show()

feature_importance = rf_classifier.feature_importances_
feature_names = data.columns[3:-1]  # Assuming the features start from the 4th
sorted_idx = np.argsort(feature_importance)

plt.figure(figsize=(10, 8))
plt.barh(range(len(sorted_idx)), feature_importance[sorted_idx], align='center
plt.yticks(range(len(sorted_idx)), feature_names[sorted_idx])
plt.xlabel('Feature Importance')
plt.ylabel('Features')
plt.title('Random Forest Feature Importance')
plt.show()
```

```
Confusion Matrix:
[[71  4]
 [ 8 25]]
Classification Report:
              precision    recall  f1-score   support

           0       0.90      0.95      0.92        75
           1       0.86      0.76      0.81        33

    accuracy                           0.89       108
   macro avg       0.88      0.85      0.86       108
weighted avg       0.89      0.89      0.89       108

ROC AUC: 0.9561616161616162
```


Receiver Operating Characteristic
True Positive Rate
False Positive Rate
ROC curve (area = 0.96)
0.0
0.2
0.4
0.6
0.8
1.0


Random Forest Feature Importance
Features
Feature Importance
Follicle No. (R)
Follicle No. (L)
Weight gain(Y/N)
Skin darkening (Y/N)
hair growth(Y/N)
AMH(ng/mL)
Cycle(R/I)
Cycle length(days)
FSH/LH
BMI
Weight (Kg)
Fast food (Y/N)
LH(mIU/mL)
Pimples(Y/N)
Hip(inch)
Vit D3 (ng/mL)
PRL(ng/mL)
Avg. F size (R) (mm)
Marraige Status (Yrs)
Avg. F size (L) (mm)
FSH(mIU/mL)
Age (yrs)
TSH (mIU/L)
Endometrium (mm)
Waist(inch)
RBS(mg/dl)
Hb(g/dl)
Height(Cm)
PRG(ng/mL)
I beta-HCG(mIU/mL)
Pulse rate(bpm)
II beta-HCG(mIU/mL)
Blood Group
RR (breaths/min)
BP _Systolic (mmHg)
Reg.Exercise(Y/N)
Hair loss(Y/N)
Waist:Hip Ratio
No. of aborptions
Pregnant(Y/N)
BP _Diastolic (mmHg)
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200

In [ ]: