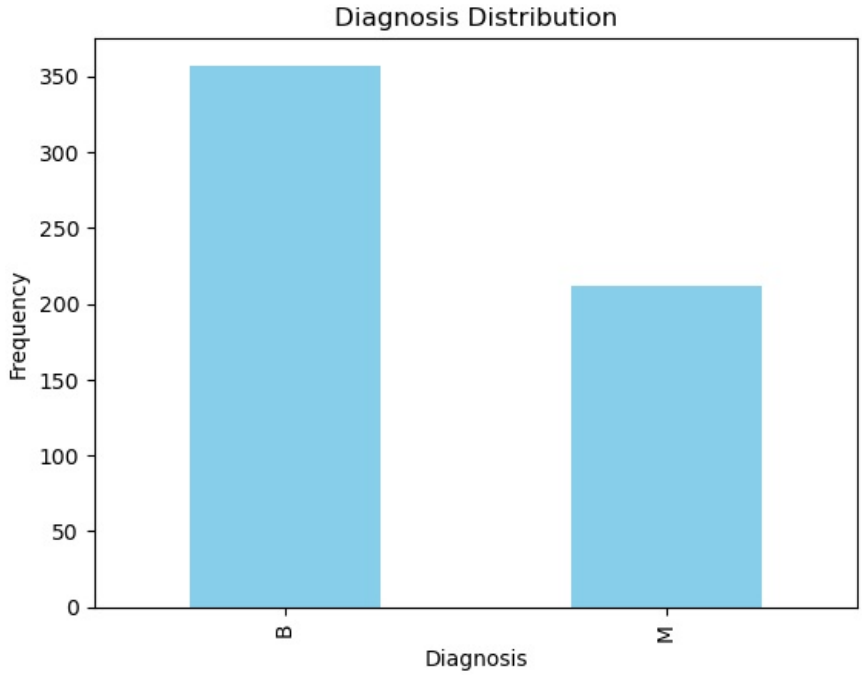
In [ ]:
```
data.columns
```

In [17]:
```
import pandas as pd
import matplotlib.pyplot as plt
import statsmodels.api as sm
from sklearn.ensemble import RandomForestClassifier

# Load the dataset
data = pd.read_csv("C:/Users/USER/Desktop/Kisii University 2023 students projects/Jefferson Mutinda/data.csv")

# Display the first few rows of the dataset
print(data.head())

# Display column names
print(data.columns)

# Bar plot for diagnosis
data['diagnosis'].value_counts().plot(kind='bar', color='skyblue')
plt.title('Diagnosis Distribution')
plt.xlabel('Diagnosis')
plt.ylabel('Frequency')
plt.show()

# Objective 1
# Recode "diagnosis" variable to 0 (benign) and 1 (malignant)
data['diagnosis'] = data['diagnosis'].apply(lambda x: 1 if x == 'M' else 0)

# Fit logistic regression model
model1 = sm.Logit(data['diagnosis'], sm.add_constant(data[['radius_mean', 'texture_mean', 'perimeter_mean', 'ar
result = model1.fit()
print(result.summary())

# Objective 2
model2 = RandomForestClassifier(n_estimators=500, random_state=123)
model2.fit(data[['radius_mean', 'texture_mean', 'perimeter_mean', 'area_mean']], data['diagnosis'])

# Objective 3
model3 = RandomForestClassifier(n_estimators=100, random_state=123)
model3.fit(data[['radius_mean', 'texture_mean', 'perimeter_mean', 'area_mean']], data['diagnosis'])
print(model3.feature_importances_)
```

```
         id diagnosis  radius_mean  texture_mean  perimeter_mean  area_mean  \
0    842302         M        17.99         10.38          122.80     1001.0
1    842517         M        20.57         17.77          132.90     1326.0
2  84300903         M        19.69         21.25          130.00     1203.0
3  84348301         M        11.42         20.38           77.58      386.1
4  84358402         M        20.29         14.34          135.10     1297.0

   smoothness_mean  compactness_mean  concavity_mean  concave points_mean  \
0          0.11840           0.27760          0.3001              0.14710
1          0.08474           0.07864          0.0869              0.07017
2          0.10960           0.15990          0.1974              0.12790
3          0.14250           0.28390          0.2414              0.10520
4          0.10030           0.13280          0.1980              0.10430

   ...  texture_worst  perimeter_worst  area_worst  smoothness_worst  \
0  ...          17.33           184.60      2019.0            0.1622
1  ...          23.41           158.80      1956.0            0.1238
2  ...          25.53           152.50      1709.0            0.1444
3  ...          26.50            98.87       567.7            0.2098
4  ...          16.67           152.20      1575.0            0.1374

   compactness_worst  concavity_worst  concave points_worst  symmetry_worst  \
0             0.6656           0.7119                0.2654          0.4601
1             0.1866           0.2416                0.1860          0.2750
2             0.4245           0.4504                0.2430          0.3613
3             0.8663           0.6869                0.2575          0.6638
4             0.2050           0.4000                0.1625          0.2364

   fractal_dimension_worst  Unnamed: 32
0                  0.11890          NaN
1                  0.08902          NaN
2                  0.08758          NaN
3                  0.17300          NaN
4                  0.07678          NaN

[5 rows x 33 columns]
Index(['id', 'diagnosis', 'radius_mean', 'texture_mean', 'perimeter_mean',
       'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean',
       'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
       'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
       'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
       'fractal_dimension_se', 'radius_worst', 'texture_worst',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst',
       'symmetry_worst', 'fractal_dimension_worst', 'Unnamed: 32'],
      dtype='object')
```




```
Optimization terminated successfully.
         Current function value: 0.185735
         Iterations 10
                           Logit Regression Results
==============================================================================
Dep. Variable:              diagnosis   No. Observations:                  569
Model:                          Logit   Df Residuals:                      564
Method:                           MLE   Df Model:                            4
Date:                Thu, 07 Dec 2023   Pseudo R-squ.:                  0.7187
Time:                        07:52:18   Log-Likelihood:                -105.68
converged:                       True   LL-Null:                       -375.72
Covariance Type:            nonrobust   LLR p-value:                1.437e-115
==============================================================================
                     coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
const              1.7729      6.870      0.258      0.796     -11.692      15.238
radius_mean       -9.4287      1.640     -5.751      0.000     -12.642      -6.215
texture_mean       0.2376      0.046      5.162      0.000       0.147       0.328
perimeter_mean     1.1507      0.164      7.001      0.000       0.829       1.473
area_mean          0.0328      0.012      2.771      0.006       0.010       0.056
==============================================================================

Possibly complete quasi-separation: A fraction 0.11 of observations can be
perfectly predicted. This might indicate that there is complete
quasi-separation. In this case some parameters will not be identified.
[0.1954285  0.12564608 0.38503995 0.29388547]
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: