coursera-cls

June 19, 2024

1 IBM Supervised Machine Learning: Classification

- Pedro Antonio Vázquez González

This analysis is to test the concepts learned during the IBM Coursera Classification course. I chose the Breast Cancer dataset out of curiosity to analyze the results of different classification methods and how they vary by applying transformations to the data or creating new synthetic data in the class with less data.

[ ]:
```
import warnings
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt


from sklearn.datasets import load_breast_cancer
from sklearn.model_selection import train_test_split, GridSearchCV,␣
  ↪RandomizedSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier
from sklearn.metrics import classification_report, confusion_matrix

warnings.filterwarnings('ignore')
```

1.0.1 Information Dataset

1) ID number
2) Diagnosis (M = malignant, B = benign) 3-32)

Ten real-valued features are computed for each cell nucleus:

```
a) radius (mean of distances from center to points on the perimeter)
b) texture (standard deviation of gray-scale values)
c) perimeter
d) area
e) smoothness (local variation in radius lengths)
f) compactness (perimeter^2 / area - 1.0)
g) concavity (severity of concave portions of the contour)
```

```
h) concave points (number of concave portions of the contour)
i) symmetry
j) fractal dimension ("coastline approximation" - 1)
```

[ ]:
```python
cancer_ds = load_breast_cancer()

data = pd.DataFrame(cancer_ds.data, columns=cancer_ds.feature_names)
data['class'] = cancer_ds.target

print(f"Data shape: {data.shape[0]} rows and {data.shape[1]} columns\n")
print(f"Data info: {data.info()}\n")

data.sample(10)
```

```
Data shape: 569 rows and 31 columns

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 31 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   mean radius              569 non-null    float64
 1   mean texture             569 non-null    float64
 2   mean perimeter           569 non-null    float64
 3   mean area                569 non-null    float64
 4   mean smoothness          569 non-null    float64
 5   mean compactness         569 non-null    float64
 6   mean concavity           569 non-null    float64
 7   mean concave points      569 non-null    float64
 8   mean symmetry            569 non-null    float64
 9   mean fractal dimension   569 non-null    float64
 10  radius error             569 non-null    float64
 11  texture error            569 non-null    float64
 12  perimeter error          569 non-null    float64
 13  area error               569 non-null    float64
 14  smoothness error         569 non-null    float64
 15  compactness error        569 non-null    float64
 16  concavity error          569 non-null    float64
 17  concave points error     569 non-null    float64
 18  symmetry error           569 non-null    float64
 19  fractal dimension error  569 non-null    float64
 20  worst radius             569 non-null    float64
 21  worst texture            569 non-null    float64
 22  worst perimeter          569 non-null    float64
 23  worst area               569 non-null    float64
 24  worst smoothness         569 non-null    float64
 25  worst compactness        569 non-null    float64
 26  worst concavity          569 non-null    float64
```

```
 27  worst concave points     569 non-null    float64
 28  worst symmetry           569 non-null    float64
 29  worst fractal dimension  569 non-null    float64
 30  class                    569 non-null    int64
dtypes: float64(30), int64(1)
memory usage: 137.9 KB
Data info: None
```

[ ]:
```
     mean radius  mean texture  mean perimeter  mean area  mean smoothness  \
130        12.19         13.29           79.08      455.8          0.10660
535        20.55         20.86          137.80     1308.0          0.10460
471        12.04         28.14           76.85      449.9          0.08752
39         13.48         20.82           88.40      559.2          0.10160
488        11.68         16.17           75.49      420.5          0.11280
490        12.25         22.44           78.18      466.5          0.08192
448        14.53         19.34           94.25      659.7          0.08388
172        15.46         11.89          102.50      736.9          0.12570
464        13.17         18.22           84.28      537.3          0.07466
462        14.40         26.99           92.25      646.1          0.06995

     mean compactness  mean concavity  mean concave points  mean symmetry  \
130           0.09509         0.02855              0.02882         0.1880
535           0.17390         0.20850              0.13220         0.2127
471           0.06000         0.02367              0.02377         0.1854
39            0.12550         0.10630              0.05439         0.1720
488           0.09263         0.04279              0.03132         0.1853
490           0.05200         0.01714              0.01261         0.1544
448           0.07800         0.08817              0.02925         0.1473
172           0.15550         0.20320              0.10970         0.1966
464           0.05994         0.04859              0.02870         0.1454
462           0.05223         0.03476              0.01737         0.1707

     mean fractal dimension  …  worst texture  worst perimeter  worst area  \
130                 0.06471  …          17.81            91.38       545.2
535                 0.06251  …          25.48           160.20      1809.0
471                 0.05698  …          33.33            87.24       567.6
39                  0.06419  …          26.02           107.30       740.4
488                 0.06401  …          21.59            86.57       549.8
490                 0.05976  …          31.99            92.74       622.9
448                 0.05746  …          28.39           108.10       830.5
172                 0.07069  …          17.04           125.00      1102.0
464                 0.05549  …          23.89            95.10       687.6
462                 0.05433  …          31.98           100.40       734.6

     worst smoothness  worst compactness  worst concavity  \
130            0.1427            0.25850          0.09915
```

```
535         0.1268          0.31350          0.44330
471         0.1041          0.09726          0.05524
39          0.1610          0.42250          0.50300
488         0.1526          0.14770          0.14900
490         0.1256          0.18040          0.12300
448         0.1089          0.26490          0.37790
172         0.1531          0.35830          0.58300
464         0.1282          0.19650          0.18760
462         0.1017          0.14600          0.14720

     worst concave points  worst symmetry  worst fractal dimension  class
130               0.08187          0.3469                  0.09241      1
535               0.21480          0.3077                  0.07569      0
471               0.05547          0.2404                  0.06639      1
39                0.22580          0.2807                  0.10710      0
488               0.09815          0.2804                  0.08024      1
490               0.06335          0.3100                  0.08203      1
448               0.09594          0.2471                  0.07463      1
172               0.18270          0.3216                  0.10100      0
464               0.10450          0.2235                  0.06925      1
462               0.05563          0.2345                  0.06464      1

[10 rows x 31 columns]
```

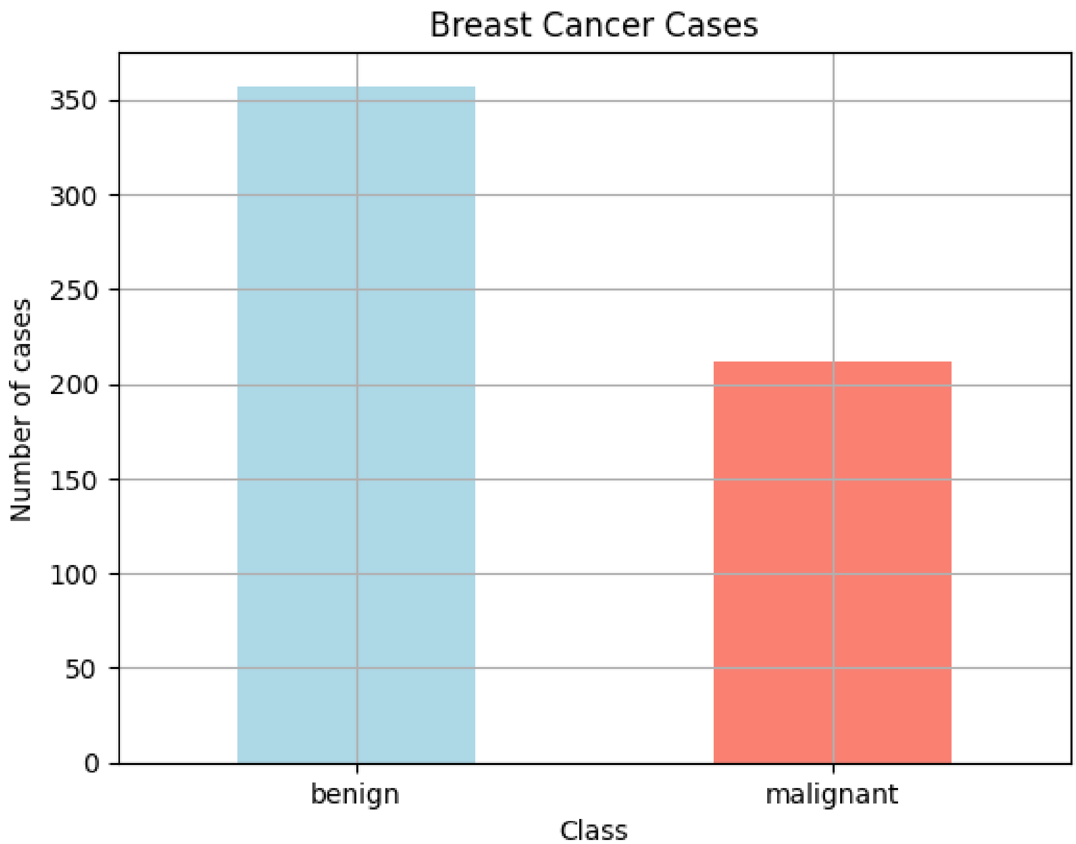
[ ]:

```python
print(
    f"Values of every class: \n{pd.Series(cancer_ds.target).map(lambda x: cancer_ds.target_names[x]).value_counts()}\n"
)

data["class"].map(lambda x: cancer_ds.target_names[x]).value_counts().plot(
    kind="bar",
    color=["lightblue", "salmon"],
    title="Breast Cancer Cases",
    xlabel="Class",
    ylabel="Number of cases",
    grid=True,
    rot=("horizontal")
)
```

```
Values of every class:
benign       357
malignant    212
Name: count, dtype: int64
```

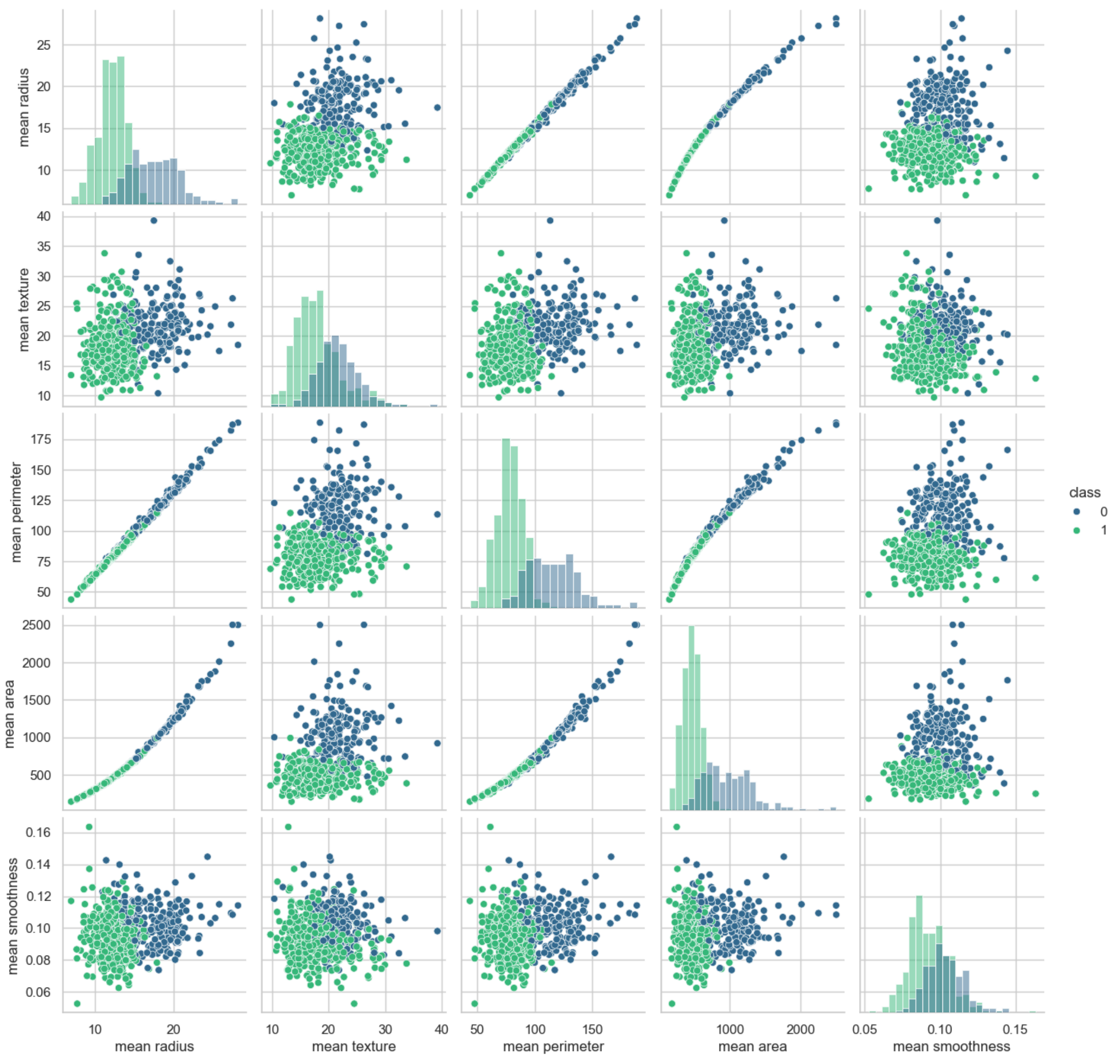
[ ]:

```
<Axes: title={'center': 'Breast Cancer Cases'}, xlabel='Class', ylabel='Number of cases'>
```



[ ]:
```
sns.set_theme(style="whitegrid")

pair_plot = sns.pairplot(data, hue='class', vars=cancer_ds.feature_names[:5],
    palette='viridis', diag_kind='hist')
pair_plot.figure.suptitle('Pairplot of Breast Cancer Features', y=1.02)
plt.show()
```

Pairplot of Breast Cancer Features



[ ]:
```
X = data.drop(columns=['class'])
y = data['class']

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

[ ]:
```
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,␣
↪random_state=123)
```

At this point we are going to evaluate different models seen in this course. We have the normal SVC, Random Forest and Gradiant Boosting in order to test different classification levels and classification methods, all while using Cross Validation with the Grid method.

[ ]:
```python
classifiers = {
    "SVC": GridSearchCV(
        SVC(),
        param_grid={
            "C": [0.1, 1, 10, 100],
            "gamma": [1, 0.1, 0.01, 0.001],
            "kernel": ["rbf"],
        },
        refit=True,
        verbose=2,
        n_jobs=-1,
        error_score=0,
    ),
    "RandomForest": GridSearchCV(
        RandomForestClassifier(),
        param_grid={
            "n_estimators": [50, 100, 200],
            "max_features": ["auto", "sqrt", "log2"],
            "max_depth": [4, 6, 8, 10],
        },
        refit=True,
        verbose=2,
        n_jobs=-1,
        error_score=0,
    ),
    "GradientBoosting": GridSearchCV(
        GradientBoostingClassifier(),
        param_grid={
            "n_estimators": [50, 100, 200],
            "learning_rate": [0.01, 0.1, 0.2],
            "max_depth": [3, 4, 5],
        },
        refit=True,
        verbose=2,
        n_jobs=-1,
        error_score=0,
    ),
}
```

[ ]:
```python
predictions = {}

for name, clf in classifiers.items():
    print(f"Training {name}...")
    clf.fit(X_train, y_train)
    print(f"Best parameters for {name}: {clf.best_params_}\n")
    predictions[name] = clf.predict(X_test)
```

```
Training SVC…
```

```
Fitting 5 folds for each of 16 candidates, totalling 80 fits
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
```

```
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
Best parameters for SVC: {'C': 10, 'gamma': 0.01, 'kernel': 'rbf'}

Training RandomForest…
Fitting 5 folds for each of 36 candidates, totalling 180 fits
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=
0.0s[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=
0.0s

[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
```

```
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.2s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.2s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.2s
```

```
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.2s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.0s
```

```
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.2s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.2s
```

```
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.2s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.1s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.2s
Best parameters for RandomForest: {'max_depth': 6, 'max_features': 'sqrt',
'n_estimators': 200}

Training GradientBoosting…
Fitting 5 folds for each of 27 candidates, totalling 135 fits
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.3s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   0.7s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   0.8s
```

```
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.2s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.2s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.2s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.5s
```

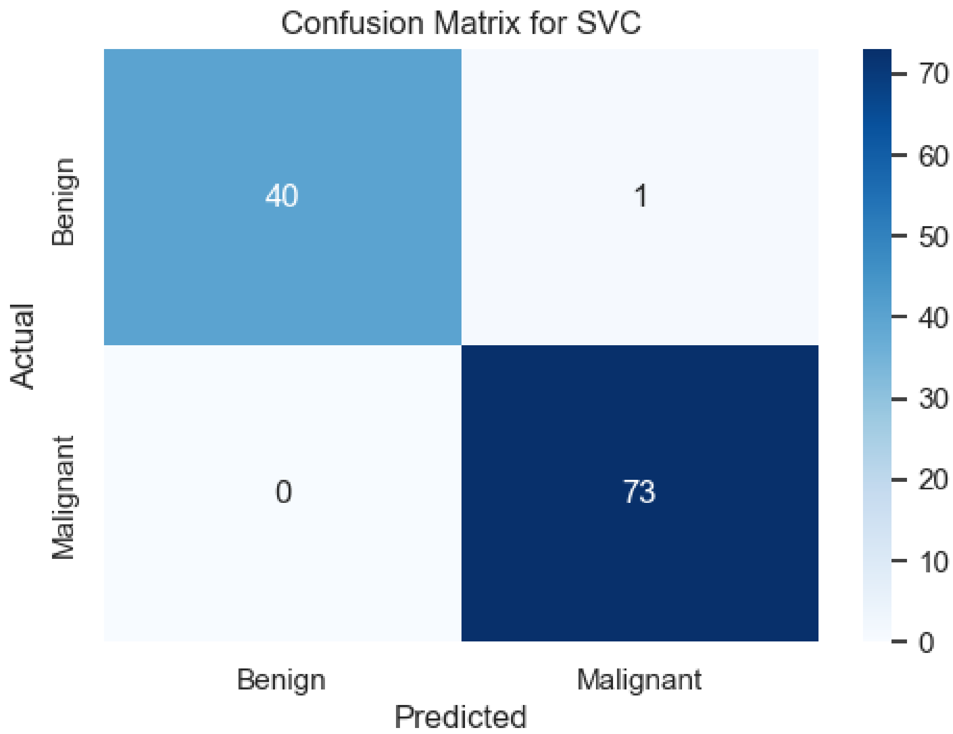
```
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   0.7s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   1.1s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   0.9s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   0.9s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.1s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   0.9s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   0.8s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.5s
```

```
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   0.8s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.4s
Best parameters for GradientBoosting: {'learning_rate': 0.2, 'max_depth': 3,
'n_estimators': 200}
```

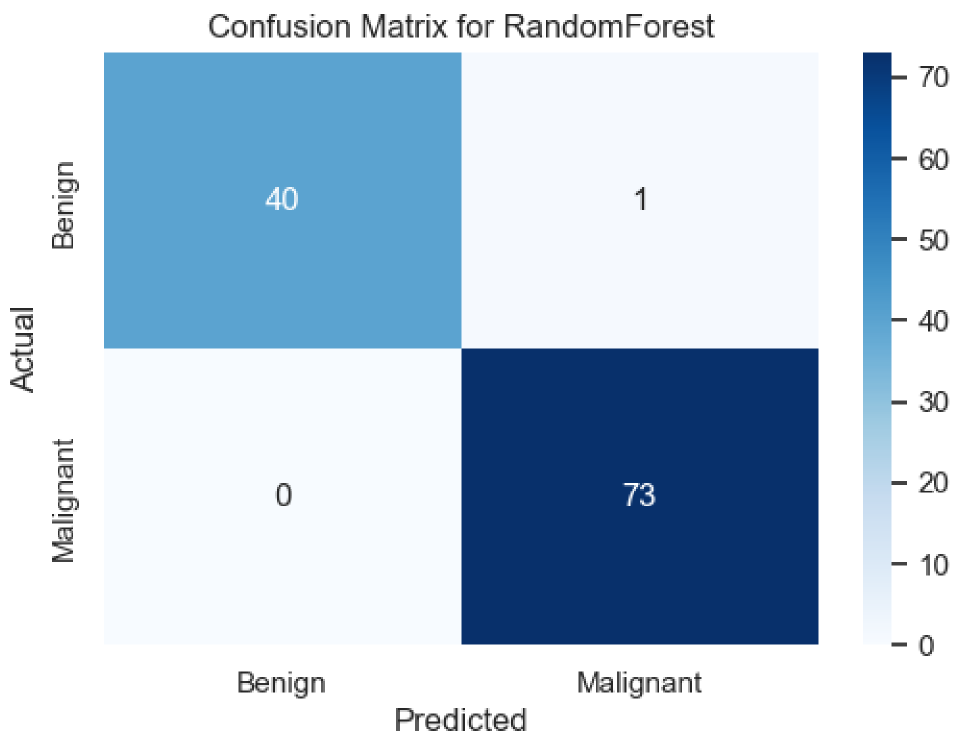
[ ]:
```python
for name, y_pred in predictions.items():
    print(f"Classification report for {name}:")
    print(classification_report(y_test, y_pred))

    cm = confusion_matrix(y_test, y_pred)
    plt.figure(figsize=(6, 4))
    sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['Benign',␣
↪'Malignant'], yticklabels=['Benign', 'Malignant'])
    plt.title(f'Confusion Matrix for {name}')
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.show()
```

```
Classification report for SVC:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99        41
           1       0.99      1.00      0.99        73

    accuracy                           0.99       114
   macro avg       0.99      0.99      0.99       114
```

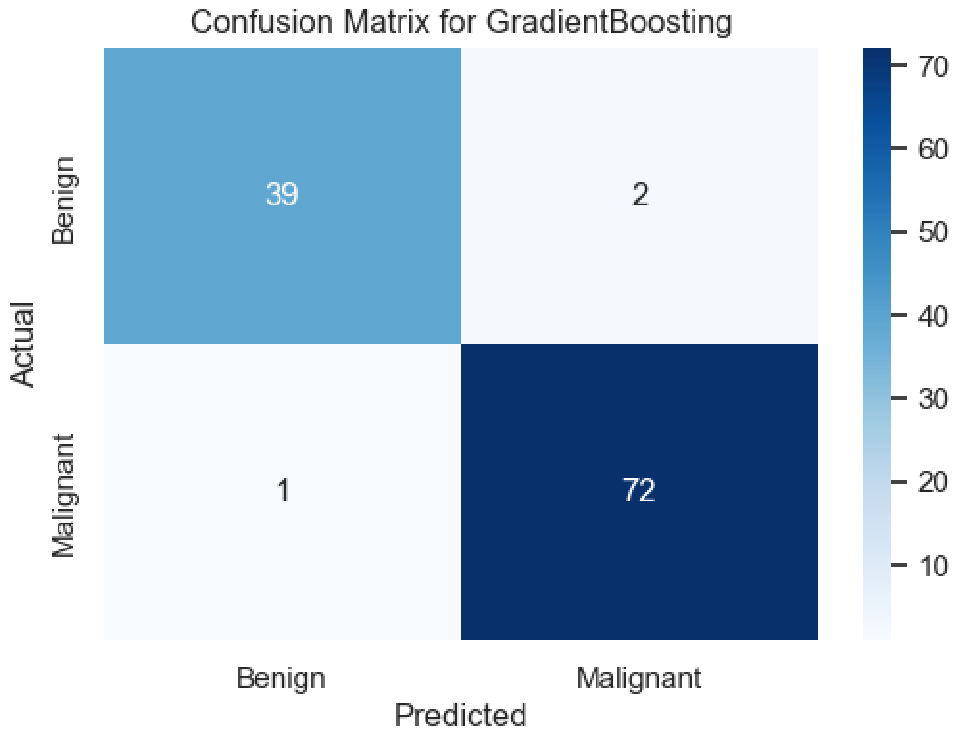
```
weighted avg       0.99      0.99      0.99       114
```




```
Classification report for RandomForest:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99        41
           1       0.99      1.00      0.99        73

    accuracy                           0.99       114
   macro avg       0.99      0.99      0.99       114
weighted avg       0.99      0.99      0.99       114
```




```
Classification report for GradientBoosting:
              precision    recall  f1-score   support

           0       0.97      0.95      0.96        41
           1       0.97      0.99      0.98        73

    accuracy                           0.97       114
   macro avg       0.97      0.97      0.97       114
weighted avg       0.97      0.97      0.97       114
```



The results show very similar results for the classifications due to the limited dataset we have. Demonstrating that mainly SVC and Random Forest have better results when classifying. We can say that **with well-regulated use of SVC it is sufficient in this case** to classify breast cancer cases.

1.1 Use SMOTE to make Oversampling

This is the most interesting test, as we will test if it makes any difference (hard to beat) when creating synthetic data for the lowest class using SMOTE.

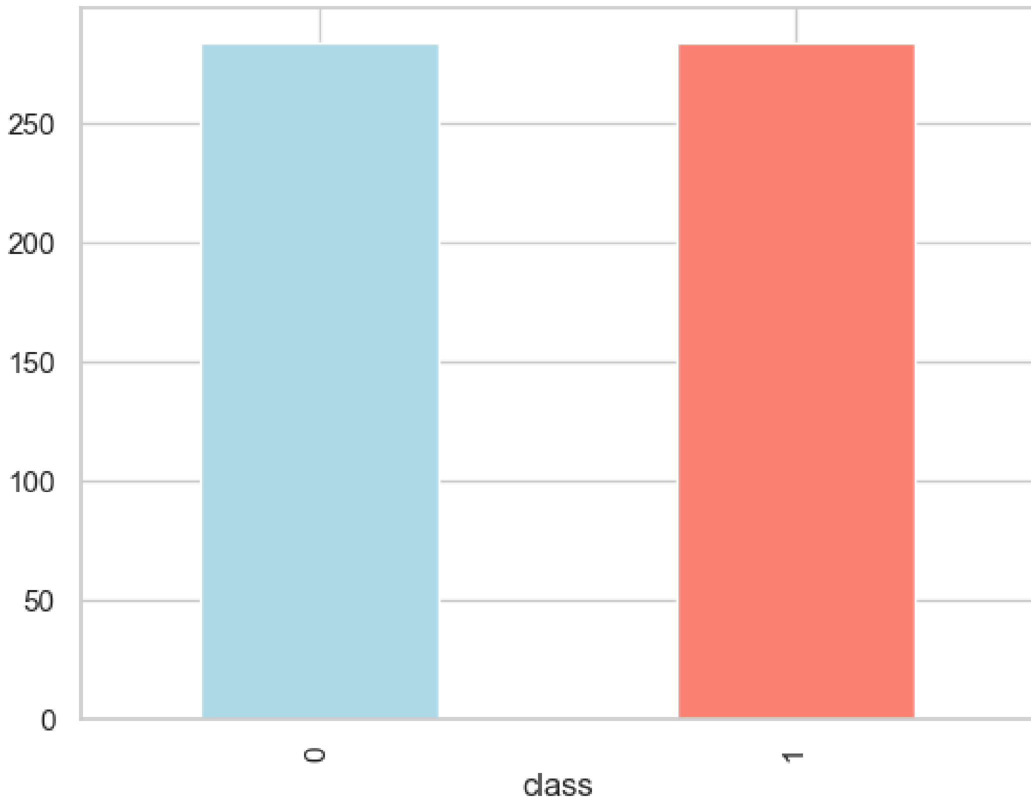
[ ]:
```
from imblearn.over_sampling import RandomOverSampler, SMOTE
```

[ ]:
```
smote_sampler = SMOTE(random_state = 123)
```

[ ]:
```
X_smo, y_smo = smote_sampler.fit_resample(X_train, y_train)
```

[ ]:
```
y_smo.value_counts().plot.bar(color=['lightblue', 'salmon'])
```

[ ]:
```
<Axes: xlabel='class'>
```



[ ]:
```
predictions_smo = {}

for name, clf in classifiers.items():
    print(f"Training {name}...")
    clf.fit(X_smo, y_smo)
    print(f"Best parameters for {name}: {clf.best_params_}\n")
    predictions_smo[name] = clf.predict(X_test)
```

```
Training SVC…
Fitting 5 folds for each of 16 candidates, totalling 80 fits
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
```

```
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=0.1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=1, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.001, kernel=rbf; total time=   0.0s
```

```
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=10, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.001, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.01, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=0.1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
[CV] END …C=100, gamma=1, kernel=rbf; total time=   0.0s
Best parameters for SVC: {'C': 10, 'gamma': 0.01, 'kernel': 'rbf'}

Training RandomForest…
Fitting 5 folds for each of 36 candidates, totalling 180 fits
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.0s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.1s
```

```
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=4, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=50; total time=   0.1s
```

```
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=4, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=6, max_features=log2, n_estimators=100; total time=   0.2s
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.3s
```

```
[CV] END …max_depth=6, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=100; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END ..max_depth=10, max_features=auto, n_estimators=200; total time=   0.0s
[CV] END …max_depth=6, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.2s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=6, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=10, max_features=sqrt, n_estimators=50; total time=   0.1s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.0s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.2s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.2s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.2s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END …max_depth=8, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END …max_depth=10, max_features=auto, n_estimators=50; total time=   0.0s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.2s
```

```
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END …max_depth=10, max_features=log2, n_estimators=50; total time=   0.1s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END …max_depth=8, max_features=sqrt, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=log2, n_estimators=100; total time=   0.1s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.3s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=log2, n_estimators=200; total time=   0.2s
[CV] END ..max_depth=10, max_features=sqrt, n_estimators=200; total time=   0.2s
Best parameters for RandomForest: {'max_depth': 10, 'max_features': 'sqrt',
'n_estimators': 50}

Training GradientBoosting…
Fitting 5 folds for each of 27 candidates, totalling 135 fits
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.01, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.6s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.4s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   1.1s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   1.1s
[CV] END …learning_rate=0.01, max_depth=4, n_estimators=50; total time=   0.4s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   1.3s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   1.2s
[CV] END ..learning_rate=0.01, max_depth=3, n_estimators=200; total time=   1.1s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.7s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.7s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.8s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.7s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=100; total time=   0.9s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.3s
```

```
[CV] END …learning_rate=0.01, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.5s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.5s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.4s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.9s
[CV] END ..learning_rate=0.01, max_depth=4, n_estimators=200; total time=   1.5s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.9s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.9s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   0.9s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=100; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.7s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.5s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.3s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.5s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.4s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END ..learning_rate=0.01, max_depth=5, n_estimators=200; total time=   1.5s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   0.9s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   0.9s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   1.1s
[CV] END …learning_rate=0.1, max_depth=3, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.7s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   1.2s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   1.2s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   1.3s
```

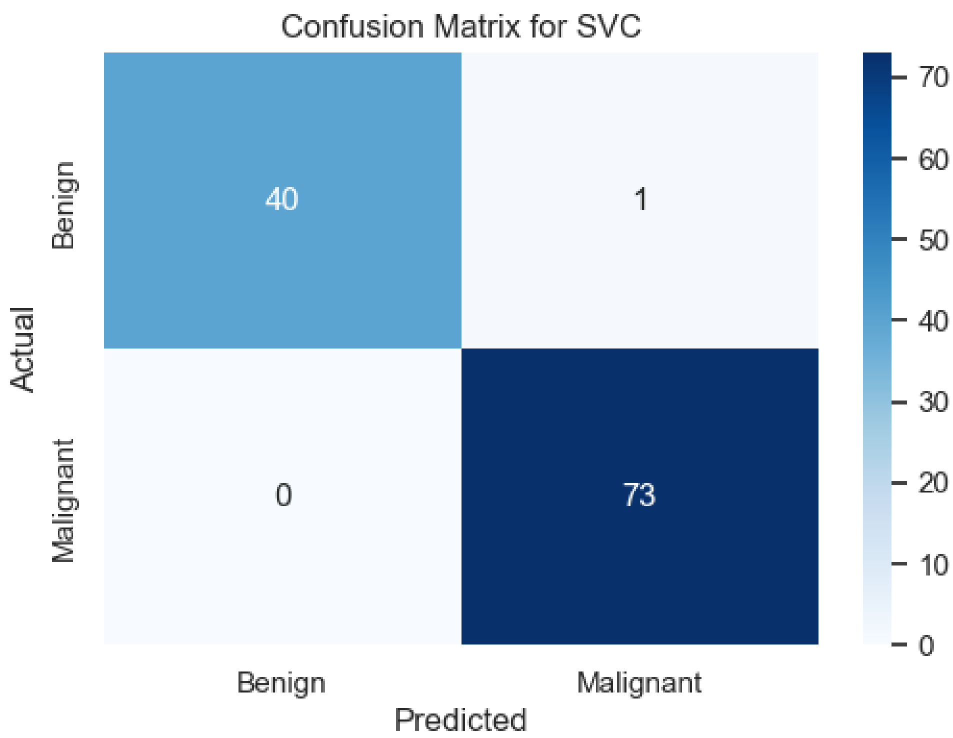
```
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   1.3s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.8s
[CV] END …learning_rate=0.1, max_depth=4, n_estimators=200; total time=   1.2s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.8s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=100; total time=   0.8s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.2s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.1s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.4s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.5s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.5s
[CV] END …learning_rate=0.1, max_depth=5, n_estimators=200; total time=   1.5s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   0.9s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   1.1s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   1.1s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.2, max_depth=3, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.8s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   0.9s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.3s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   0.9s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=50; total time=   0.4s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   1.0s
[CV] END …learning_rate=0.2, max_depth=4, n_estimators=200; total time=   1.1s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.7s
```

```
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.8s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=100; total time=   0.7s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.5s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.6s
[CV] END …learning_rate=0.2, max_depth=5, n_estimators=200; total time=   0.7s
Best parameters for GradientBoosting: {'learning_rate': 0.1, 'max_depth': 3,
'n_estimators': 200}
```

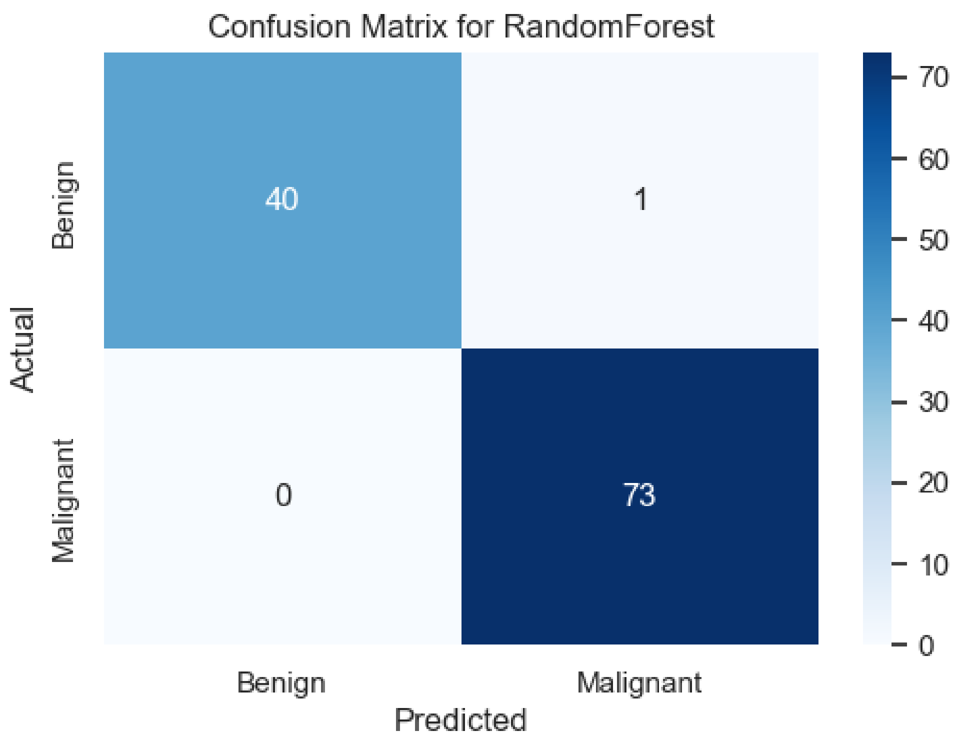
[ ]:

```python
for name, y_pred in predictions_smo.items():
    print(f"Classification report for {name}:")
    print(classification_report(y_test, y_pred))

    cm = confusion_matrix(y_test, y_pred)
    plt.figure(figsize=(6, 4))
    sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['Benign',␣
 ↪'Malignant'], yticklabels=['Benign', 'Malignant'])
    plt.title(f'Confusion Matrix for {name}')
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.show()
```

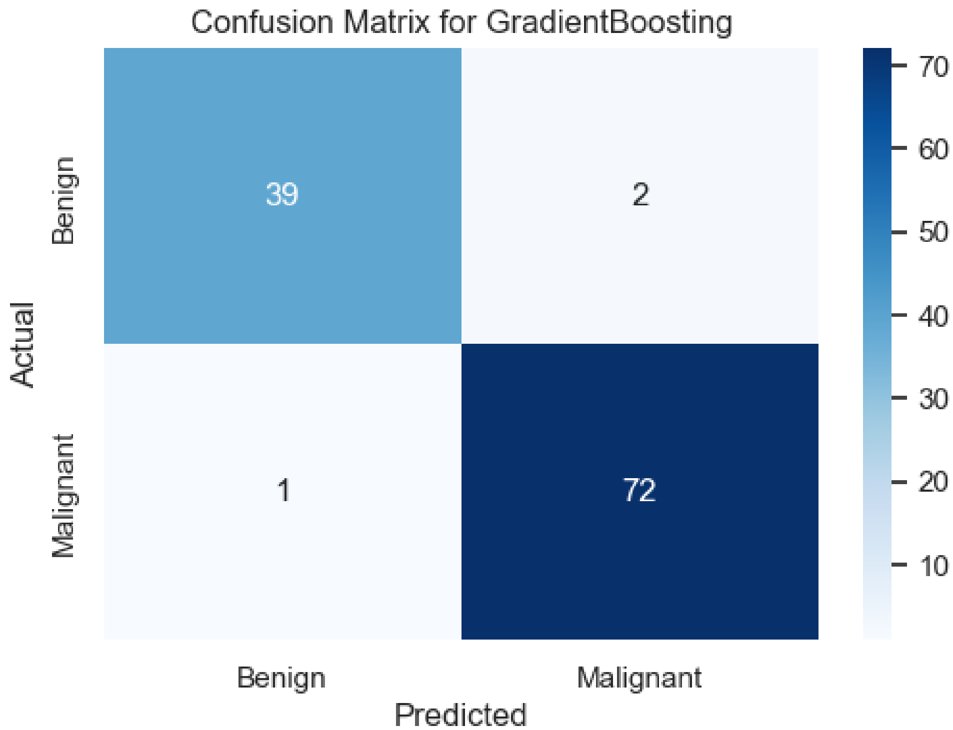
```
Classification report for SVC:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99        41
           1       0.99      1.00      0.99        73

    accuracy                           0.99       114
   macro avg       0.99      0.99      0.99       114
weighted avg       0.99      0.99      0.99       114
```




```
Classification report for RandomForest:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99        41
           1       0.99      1.00      0.99        73

    accuracy                           0.99       114
   macro avg       0.99      0.99      0.99       114
weighted avg       0.99      0.99      0.99       114
```




```
Classification report for GradientBoosting:
              precision    recall  f1-score   support

           0       0.97      0.95      0.96        41
           1       0.97      0.99      0.98        73

    accuracy                           0.97       114
   macro avg       0.97      0.97      0.97       114
weighted avg       0.97      0.97      0.97       114
```



1.2 Conclusions

It appears that the results are completely similar regardless of whether we create more synthetic data for the lowest class or leave it with the original distribution of the dataset. This is due to the size of the lowest class, which despite having less data, is a representative sample of the cases we have. And that is what we are looking for, to make the sample representative for the case study.

1.2.1 Some solutions or improvements

To study what we have learned, we can implement different datasets that allow us to have more biased data that test the algorithms presented. However, this also has a very large filter and that is that the person in charge of collecting the data is often the one in charge of taking representative samples. But when confronted in real life, we may come across information that we have to process so that the classifiers work much better.