boruta

January 10, 2023

1 Paquetes a utilizar

[ ]:

```
from boruta import BorutaPy
from sklearn import metrics
import pandas as pd
import numpy as np
import xgboost as xgb
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
```

2 Base de datos

[ ]:

```
df = pd.read_csv("data/wisconsin_breast_cancer_dataset.csv")

print(df.describe().T)
print(df.isnull().sum())
df = df.dropna()
df = df.rename(columns={'diagnosis':'Label'})
print(df.dtypes)
df['Label'].value_counts()
```

```
                   count          mean           std          min  \
id                 569.0  3.037183e+07  1.250206e+08  8670.000000
radius_mean        569.0  1.412729e+01  3.524049e+00     6.981000
texture_mean       569.0  1.928965e+01  4.301036e+00     9.710000
perimeter_mean     569.0  9.196903e+01  2.429898e+01    43.790000
area_mean          569.0  6.548891e+02  3.519141e+02   143.500000
smoothness_mean    569.0  9.636028e-02  1.406413e-02     0.052630
compactness_mean   569.0  1.043410e-01  5.281276e-02     0.019380
concavity_mean     569.0  8.879932e-02  7.971981e-02     0.000000
points_mean        569.0  4.891915e-02  3.880284e-02     0.000000
symmetry_mean      569.0  1.811619e-01  2.741428e-02     0.106000
dimension_mean     569.0  6.279761e-02  7.060363e-03     0.049960
radius_se          569.0  4.051721e-01  2.773127e-01     0.111500
texture_se         569.0  1.216853e+00  5.516484e-01     0.360200
perimeter_se       569.0  2.866059e+00  2.021855e+00     0.757000
area_se            569.0  4.033708e+01  4.549101e+01     6.802000
```

```
smoothness_se      569.0  7.040979e-03  3.002518e-03     0.001713
compactness_se     569.0  2.547814e-02  1.790818e-02     0.002252
concavity_se       569.0  3.189372e-02  3.018606e-02     0.000000
points_se          569.0  1.179614e-02  6.170285e-03     0.000000
symmetry_se        569.0  2.054230e-02  8.266372e-03     0.007882
dimension_se       569.0  3.794904e-03  2.646071e-03     0.000895
radius_worst       569.0  1.626919e+01  4.833242e+00     7.930000
texture_worst      569.0  2.567722e+01  6.146258e+00    12.020000
perimeter_worst    569.0  1.072612e+02  3.360254e+01    50.410000
area_worst         569.0  8.805831e+02  5.693570e+02   185.200000
smoothness_worst   569.0  1.323686e-01  2.283243e-02     0.071170
compactness_worst  569.0  2.542650e-01  1.573365e-01     0.027290
concavity_worst    569.0  2.721885e-01  2.086243e-01     0.000000
points_worst       569.0  1.146062e-01  6.573234e-02     0.000000
symmetry_worst     569.0  2.900756e-01  6.186747e-02     0.156500
dimension_worst    569.0  8.394582e-02  1.806127e-02     0.055040

                            25%            50%           75%           max
id                 869218.000000  906024.000000  8.813129e+06  9.113205e+08
radius_mean            11.700000      13.370000  1.578000e+01  2.811000e+01
texture_mean           16.170000      18.840000  2.180000e+01  3.928000e+01
perimeter_mean         75.170000      86.240000  1.041000e+02  1.885000e+02
area_mean             420.300000     551.100000  7.827000e+02  2.501000e+03
smoothness_mean         0.086370       0.095870  1.053000e-01  1.634000e-01
compactness_mean        0.064920       0.092630  1.304000e-01  3.454000e-01
concavity_mean          0.029560       0.061540  1.307000e-01  4.268000e-01
points_mean             0.020310       0.033500  7.400000e-02  2.012000e-01
symmetry_mean           0.161900       0.179200  1.957000e-01  3.040000e-01
dimension_mean          0.057700       0.061540  6.612000e-02  9.744000e-02
radius_se               0.232400       0.324200  4.789000e-01  2.873000e+00
texture_se              0.833900       1.108000  1.474000e+00  4.885000e+00
perimeter_se            1.606000       2.287000  3.357000e+00  2.198000e+01
area_se                17.850000      24.530000  4.519000e+01  5.422000e+02
smoothness_se           0.005169       0.006380  8.146000e-03  3.113000e-02
compactness_se          0.013080       0.020450  3.245000e-02  1.354000e-01
concavity_se            0.015090       0.025890  4.205000e-02  3.960000e-01
points_se               0.007638       0.010930  1.471000e-02  5.279000e-02
symmetry_se             0.015160       0.018730  2.348000e-02  7.895000e-02
dimension_se            0.002248       0.003187  4.558000e-03  2.984000e-02
radius_worst           13.010000      14.970000  1.879000e+01  3.604000e+01
texture_worst          21.080000      25.410000  2.972000e+01  4.954000e+01
perimeter_worst        84.110000      97.660000  1.254000e+02  2.512000e+02
area_worst            515.300000     686.500000  1.084000e+03  4.254000e+03
smoothness_worst        0.116600       0.131300  1.460000e-01  2.226000e-01
compactness_worst       0.147200       0.211900  3.391000e-01  1.058000e+00
concavity_worst         0.114500       0.226700  3.829000e-01  1.252000e+00
points_worst            0.064930       0.099930  1.614000e-01  2.910000e-01
symmetry_worst          0.250400       0.282200  3.179000e-01  6.638000e-01
```

```
dimension_worst         0.071460       0.080040  9.208000e-02  2.075000e-01
id                   0
diagnosis            0
radius_mean          0
texture_mean         0
perimeter_mean       0
area_mean            0
smoothness_mean      0
compactness_mean     0
concavity_mean       0
points_mean          0
symmetry_mean        0
dimension_mean       0
radius_se            0
texture_se           0
perimeter_se         0
area_se              0
smoothness_se        0
compactness_se       0
concavity_se         0
points_se            0
symmetry_se          0
dimension_se         0
radius_worst         0
texture_worst        0
perimeter_worst      0
area_worst           0
smoothness_worst     0
compactness_worst    0
concavity_worst      0
points_worst         0
symmetry_worst       0
dimension_worst      0
dtype: int64
id                     int64
Label                 object
radius_mean          float64
texture_mean         float64
perimeter_mean       float64
area_mean            float64
smoothness_mean      float64
compactness_mean     float64
concavity_mean       float64
points_mean          float64
symmetry_mean        float64
dimension_mean       float64
radius_se            float64
texture_se           float64
```

```
perimeter_se          float64
area_se               float64
smoothness_se         float64
compactness_se        float64
concavity_se          float64
points_se             float64
symmetry_se           float64
dimension_se          float64
radius_worst          float64
texture_worst         float64
perimeter_worst       float64
area_worst            float64
smoothness_worst      float64
compactness_worst     float64
concavity_worst       float64
points_worst          float64
symmetry_worst        float64
dimension_worst       float64
dtype: object
```

[ ]:

```
B    357
M    212
Name: Label, dtype: int64
```

3 Variable dependiente que debe predecirse

[ ]:

```
y = df["Label"].values

# Codificación de datos categóricos
labelencoder = LabelEncoder()
Y = labelencoder.fit_transform(y)
```

4 Definir x, normalizar valores y definir variables independientes

[ ]:

```
X = df.drop(labels = ["Label", "id"], axis=1)

feature_names = np.array(X.columns)

scaler = StandardScaler()
scaler.fit(X)
X = scaler.transform(X)
```

5 Train and test para verificar la precisión después de ajustar el modelo

[ ]:

```
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.25,␣
  ↪random_state=42)
```

6 XGBOOST para ser utilizado por Boruta

[ ]:

```
model = xgb.XGBClassifier()
```

- Crear funciones de sombra: funciones aleatorias y valores aleatorios en columnas
- Entrenar Random Forest / XGBoost y calcular la importancia de la característica a través de la disminución media de la impureza
- Comprobar si las características reales tienen mayor importancia en comparación con las características de sombra
- Repetir esto para cada iteración
- Si la función original funcionó mejor, marcarla como importante

[ ]:

```
# definir el método de selección de características de Boruta
feat_selector = BorutaPy(model, n_estimators='auto', verbose=2, random_state=1)

# encontrar todas las características relevantes
feat_selector.fit(X_train, y_train)

# llamar a transform() en X para filtrarlo a las características seleccionadas
X_filtered = feat_selector.transform(X_train)  # Aplicar selección de␣
  ↪características y devolver datos transformados
```

```
Iteration:      1 / 100
Confirmed:      0
Tentative:      30
Rejected:       0
Iteration:      2 / 100
Confirmed:      0
Tentative:      30
Rejected:       0
Iteration:      3 / 100
Confirmed:      0
Tentative:      30
Rejected:       0
Iteration:      4 / 100
Confirmed:      0
Tentative:      30
Rejected:       0
Iteration:      5 / 100
Confirmed:      0
Tentative:      30
```

```
Rejected:       0
Iteration:      6 / 100
Confirmed:      0
Tentative:      30
Rejected:       0
Iteration:      7 / 100
Confirmed:      0
Tentative:      30
Rejected:       0
Iteration:      8 / 100
Confirmed:      5
Tentative:      13
Rejected:       12
Iteration:      9 / 100
Confirmed:      5
Tentative:      13
Rejected:       12
Iteration:      10 / 100
Confirmed:      5
Tentative:      13
Rejected:       12
Iteration:      11 / 100
Confirmed:      5
Tentative:      13
Rejected:       12
Iteration:      12 / 100
Confirmed:      5
Tentative:      13
Rejected:       12
Iteration:      13 / 100
Confirmed:      5
Tentative:      12
Rejected:       13
Iteration:      14 / 100
Confirmed:      5
Tentative:      12
Rejected:       13
Iteration:      15 / 100
Confirmed:      5
Tentative:      12
Rejected:       13
Iteration:      16 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      17 / 100
Confirmed:      5
Tentative:      9
```

```
Rejected:       16
Iteration:      18 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      19 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      20 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      21 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      22 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      23 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      24 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      25 / 100
Confirmed:      5
Tentative:      9
Rejected:       16
Iteration:      26 / 100
Confirmed:      5
Tentative:      8
Rejected:       17
Iteration:      27 / 100
Confirmed:      5
Tentative:      8
Rejected:       17
Iteration:      28 / 100
Confirmed:      5
Tentative:      8
Rejected:       17
Iteration:      29 / 100
Confirmed:      5
Tentative:      8
```

```
Rejected:       17
Iteration:      30 / 100
Confirmed:      5
Tentative:      8
Rejected:       17
Iteration:      31 / 100
Confirmed:      5
Tentative:      8
Rejected:       17
Iteration:      32 / 100
Confirmed:      6
Tentative:      7
Rejected:       17
Iteration:      33 / 100
Confirmed:      6
Tentative:      7
Rejected:       17
Iteration:      34 / 100
Confirmed:      6
Tentative:      7
Rejected:       17
Iteration:      35 / 100
Confirmed:      6
Tentative:      7
Rejected:       17
Iteration:      36 / 100
Confirmed:      6
Tentative:      7
Rejected:       17
Iteration:      37 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      38 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      39 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      40 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      41 / 100
Confirmed:      7
Tentative:      6
```

```
Rejected:       17
Iteration:      42 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      43 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      44 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      45 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      46 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      47 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      48 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      49 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      50 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      51 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      52 / 100
Confirmed:      7
Tentative:      6
Rejected:       17
Iteration:      53 / 100
Confirmed:      7
Tentative:      6
```

```
Rejected:       17
Iteration:      54 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      55 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      56 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      57 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      58 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      59 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      60 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      61 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      62 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      63 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      64 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      65 / 100
Confirmed:      8
Tentative:      5
```

```
Rejected:       17
Iteration:      66 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      67 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      68 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      69 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      70 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      71 / 100
Confirmed:      8
Tentative:      5
Rejected:       17
Iteration:      72 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      73 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      74 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      75 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      76 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      77 / 100
Confirmed:      8
Tentative:      4
```

```
Rejected:       18
Iteration:      78 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      79 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      80 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      81 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      82 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      83 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      84 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      85 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      86 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      87 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      88 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      89 / 100
Confirmed:      8
Tentative:      4
```

```
Rejected:       18
Iteration:      90 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      91 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      92 / 100
Confirmed:      8
Tentative:      4
Rejected:       18
Iteration:      93 / 100
Confirmed:      9
Tentative:      3
Rejected:       18
Iteration:      94 / 100
Confirmed:      9
Tentative:      3
Rejected:       18
Iteration:      95 / 100
Confirmed:      9
Tentative:      3
Rejected:       18
Iteration:      96 / 100
Confirmed:      9
Tentative:      3
Rejected:       18
Iteration:      97 / 100
Confirmed:      9
Tentative:      3
Rejected:       18
Iteration:      98 / 100
Confirmed:      9
Tentative:      3
Rejected:       18
Iteration:      99 / 100
Confirmed:      9
Tentative:      3
Rejected:       18


BorutaPy finished running.

Iteration:      100 / 100
Confirmed:      9
Tentative:      1
```

```
Rejected:       18
```

```
[ ]: """
     Revisar las características
     """
     # zip nombres de características, rangos y decisiones
     feature_ranks = list(zip(feature_names,
                              feat_selector.ranking_,
                              feat_selector.support_))

     # imprimir los resultados
     for feat in feature_ranks:
         print('Feature: {:<30} Rank: {},  Keep: {}'.format(feat[0], feat[1],␣
       ↪feat[2]))

     # Ahora usar el subconjunto de funciones para ajustar el modelo XGBoost en los␣
       ↪datos de entrenamiento
     xgb_model = xgb.XGBClassifier()
     xgb_model.fit(X_filtered, y_train)

     # Ahora predecir con datos de prueba usando el modelo entrenado
     # Primero aplicar la transformación del selector de funciones para asegurarse␣
       ↪de que se seleccionen las mismas funciones de los datos de prueba
     X_test_filtered = feat_selector.transform(X_test)
     prediction_xgb = xgb_model.predict(X_test_filtered)

     # Imprimir precisión
     print("Precisión = ", metrics.accuracy_score(y_test, prediction_xgb))
```

```
Feature: radius_mean                    Rank: 8,  Keep: False
Feature: texture_mean                   Rank: 1,  Keep: True
Feature: perimeter_mean                 Rank: 2,  Keep: False
Feature: area_mean                      Rank: 1,  Keep: True
Feature: smoothness_mean                Rank: 13,  Keep: False
Feature: compactness_mean               Rank: 17,  Keep: False
Feature: concavity_mean                 Rank: 4,  Keep: False
Feature: points_mean                    Rank: 1,  Keep: True
Feature: symmetry_mean                  Rank: 15,  Keep: False
Feature: dimension_mean                 Rank: 21,  Keep: False
Feature: radius_se                      Rank: 5,  Keep: False
Feature: texture_se                     Rank: 9,  Keep: False
Feature: perimeter_se                   Rank: 3,  Keep: False
Feature: area_se                        Rank: 10,  Keep: False
Feature: smoothness_se                  Rank: 22,  Keep: False
Feature: compactness_se                 Rank: 14,  Keep: False
Feature: concavity_se                   Rank: 6,  Keep: False
Feature: points_se                      Rank: 11,  Keep: False
```

```
Feature: symmetry_se                    Rank: 20,  Keep: False
Feature: dimension_se                   Rank: 6,  Keep: False
Feature: radius_worst                   Rank: 1,  Keep: True
Feature: texture_worst                  Rank: 1,  Keep: True
Feature: perimeter_worst                Rank: 1,  Keep: True
Feature: area_worst                     Rank: 1,  Keep: True
Feature: smoothness_worst               Rank: 12,  Keep: False
Feature: compactness_worst              Rank: 18,  Keep: False
Feature: concavity_worst                Rank: 1,  Keep: True
Feature: points_worst                   Rank: 1,  Keep: True
Feature: symmetry_worst                 Rank: 15,  Keep: False
Feature: dimension_worst                Rank: 18,  Keep: False
Precisión =  0.9790209790209791
```