

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

```
pip install ucimlrepo
```

```
Collecting ucimlrepo
  Downloading ucimlrepo-0.0.7-py3-none-any.whl.metadata (5.5 kB)
Requirement already satisfied: pandas>=1.0.0 in /usr/local/lib/python3.11/dist-packages (from ucimlrepo) (2.2.2)
Requirement already satisfied: certifi>=2020.12.5 in /usr/local/lib/python3.11/dist-packages (from ucimlrepo) (2025.8.3)
Requirement already satisfied: numpy>=1.23.2 in /usr/local/lib/python3.11/dist-packages (from pandas>=1.0.0->ucimlrepo) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas>=1.0.0->ucimlrepo) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas>=1.0.0->ucimlrepo) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas>=1.0.0->ucimlrepo) (2025.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas>=1.0.0->ucimlrepo) (1.17.0)
Downloading ucimlrepo-0.0.7-py3-none-any.whl (8.0 kB)
Installing collected packages: ucimlrepo
Successfully installed ucimlrepo-0.0.7
```

```
from ucimlrepo import fetch_ucirepo
```

```
# fetch dataset
```

```
breast_cancer_wisconsin_diagnostic = fetch_ucirepo(id=17)
```

```
# data (as pandas dataframes)
```

```
X = breast_cancer_wisconsin_diagnostic.data.features
```

```
y = breast_cancer_wisconsin_diagnostic.data.targets
```

```
# metadata
```

```
print(breast_cancer_wisconsin_diagnostic.metadata)
```

```
# variable information
```

```
print(breast_cancer_wisconsin_diagnostic.variables)
```

```

9      concave_points1  Feature  Continuous  None  None  None
10      symmetry1      Feature  Continuous  None  None  None
11  fractal_dimension1  Feature  Continuous  None  None  None
12      radius2        Feature  Continuous  None  None  None
13      texture2        Feature  Continuous  None  None  None
14      perimeter2      Feature  Continuous  None  None  None
15      area2           Feature  Continuous  None  None  None
16      smoothness2     Feature  Continuous  None  None  None
17      compactness2    Feature  Continuous  None  None  None
18      concavity2      Feature  Continuous  None  None  None
19      concave_points2  Feature  Continuous  None  None  None
20      symmetry2       Feature  Continuous  None  None  None
21  fractal_dimension2  Feature  Continuous  None  None  None
22      radius3         Feature  Continuous  None  None  None

```

```

0      no
1      no
2      no
3      no
4      no
5      no
6      no
7      no
8      no
9      no
10     no
11     no
12     no
13     no
14     no
15     no
16     no
17     no
18     no
19     no
20     no
21     no
22     no
23     no
24     no
25     no
26     no
27     no
28     no
29     no
30     no
31     no

```

```
import pandas as pd
```

```
# Define column names based on dataset documentation
```

```
columns = [
    "id", "clump_thickness", "uniformity_cell_size", "uniformity_cell_shape",
    "marginal_adhesion", "single_epithelial_cell_size", "bare_nuclei",
    "bland_chromatin", "normal_nucleoli", "mitoses", "class"
]
```

```
# Load dataset directly from UCI
```

```
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/breast-cancer-wisconsin.data"
```

```
df = pd.read_csv(url, names=columns)
```

```
# Preview
```

```
print(df.head())
```

```

↵      id  clump_thickness  uniformity_cell_size  uniformity_cell_shape \
0  1000025             5             1             1
1  1002945             5             4             4
2  1015425             3             1             1
3  1016277             6             8             8
4  1017023             4             1             1

      marginal_adhesion  single_epithelial_cell_size  bare_nuclei \
0             1             2             1
1             5             7            10
2             1             2             2
3             1             3             4
4             3             2             1

```

	bland_chromatin	normal_nucleoli	mitoses	class
0	3	1	1	2
1	3	2	1	2
2	3	1	1	2
3	3	7	1	2
4	3	1	1	2

```
# Step 1: Load dataset from UCI
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/breast-cancer-wisconsin.data"
columns = [
    "id", "clump_thickness", "uniformity_cell_size", "uniformity_cell_shape",
    "marginal_adhesion", "single_epithelial_cell_size", "bare_nuclei",
    "bland_chromatin", "normal_nucleoli", "mitoses", "class"
]
df = pd.read_csv(url, names=columns)
```

```
import joblib
from xgboost import XGBClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
```


```
# Step 1: Load preprocessed data
X_train, X_test, y_train, y_test = joblib.load("preprocessed_data.pkl")
```

```
# Convert labels: 2 → 0 (benign), 4 → 1 (malignant)
y_train = y_train.map({2: 0, 4: 1})
y_test = y_test.map({2: 0, 4: 1})
```

```
# Step 1: Load preprocessed data
X_train, X_test, y_train, y_test = joblib.load("preprocessed_data.pkl")
```

```
# Step 1.5: Remap labels for XGBoost
y_train = y_train.map({2: 0, 4: 1})
y_test = y_test.map({2: 0, 4: 1})
```


```
# Step 2: Initialize and train the model
model = XGBClassifier(use_label_encoder=False, eval_metric='logloss', random_state=42)
model.fit(X_train, y_train)
```

```
 /usr/local/lib/python3.11/dist-packages/xgboost/training.py:183: UserWarning: [03:28:45] WARNING: /workspace/src/learner.cc:738:
.....

# Step 3: Make predictions
y_pred = model.predict(X_test)

# Step 4: Evaluate performance
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)

# Step 5: Display results
print("⚡ XGBoost Results")
print(f"Accuracy: {accuracy:.4f}")
print("Confusion Matrix:")
print(conf_matrix)
```

```
 ⚡ XGBoost Results
Accuracy: 0.9474
Confusion Matrix:
[[102  1]
 [ 8 60]]
```

Start coding or [generate](#) with AI.