

Pumpkin Classification with Lazypredict Method

```
In [1]: !pip install openpyxl
Collecting openpyxl
  Downloading openpyxl-3.1.2-py2.py3-none-any.whl (249 kB)
    |████████| 249 kB 6.4 MB/s
Collecting et-xmlfile
  Downloading et_xmlfile-1.1.0-py3-none-any.whl (4.7 kB)
Installing collected packages: et-xmlfile, openpyxl
Successfully installed et-xmlfile-1.1.0 openpyxl-3.1.2
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv

In [2]: import pandas as pd
train=pd.read_excel('../input/Pumpkin_Seeds_Dataset/Pumpkin_Seeds_Dataset.xlsx')
train.head()

Out[2]:   Area Perimeter Major_Axis_Length Minor_Axis_Length Convex_Area Equiv_Diameter Eccentricity Solidity Extent Roundness Aspect_Ration
0  56276      888.242        326.1485       220.2388      56831     267.6805    0.7376  0.9902  0.7453  0.8963
1  76631      1068.146       417.1932       234.2289      77280     312.3614    0.8275  0.9916  0.7151  0.8440
2  71623      1082.987       435.8328       211.0457      72663     301.9822    0.8749  0.9857  0.7400  0.7674
3  66458      992.051       381.5638       222.5322      67118     290.8899    0.8123  0.9902  0.7396  0.8486
4  66107      998.146       383.8883       220.4545      67117     290.1207    0.8187  0.9850  0.6752  0.8338

In [3]: train.shape
Out[3]: (2500, 13)

In [4]: train.drop_duplicates(keep=False, inplace=False)
train.shape
Out[4]: (2500, 13)

In [5]: train.nunique().sort_values(ascending=False)
Out[5]: Major_Axis_Length    2499
Minor_Axis_Length    2497
Perimeter          2490
Convex_Area         2445
Area               2424
Equiv_Diameter     2424
Aspect_Ration       2237
Roundness           1480
Compactness          1405
Extent              1392
Eccentricity         1295
Solidity             166
Class                  2
dtype: int64

In [6]: train.shape
Out[6]: (2500, 13)

In [7]: round(train.isnull().sum()*100/len(train),2).sort_values(ascending=False)
Out[7]: Area          0.0
Perimeter      0.0
Major_Axis_Length 0.0
Minor_Axis_Length 0.0
Convex_Area     0.0
Equiv_Diameter  0.0
Eccentricity    0.0
Solidity        0.0
Extent          0.0
Roundness        0.0
Aspect_Ration   0.0
Compactness      0.0
Class            0.0
dtype: float64

In [8]: round(train['Class'].value_counts()*100/len(train),2)
Out[8]: Çerçevevik      52.0
Ürgüp Sivrisi      48.0
Name: Class, dtype: float64

In [9]: train['Class']=train['Class'].replace({'Çerçevevik':0,'Ürgüp Sivrisi':1})
train['Class'].value_counts()
Out[9]: 0    1300
1    1200
Name: Class, dtype: int64
```

```
In [10]: !pip install lazypredict

Collecting lazypredict
  Downloading lazypredict-0.2.12-py2.py3-none-any.whl (12 kB)
Requirement already satisfied: click in /opt/conda/lib/python3.7/site-packages (from lazypredict) (8.0.3)
Requirement already satisfied: xgboost in /opt/conda/lib/python3.7/site-packages (from lazypredict) (1.5.2)
Requirement already satisfied: scikit-learn in /opt/conda/lib/python3.7/site-packages (from lazypredict) (0.23.2)
Requirement already satisfied: joblib in /opt/conda/lib/python3.7/site-packages (from lazypredict) (1.1.0)
Requirement already satisfied: pandas in /opt/conda/lib/python3.7/site-packages (from lazypredict) (1.3.5)
Requirement already satisfied: lightgbm in /opt/conda/lib/python3.7/site-packages (from lazypredict) (3.3.1)
Requirement already satisfied: tqdm in /opt/conda/lib/python3.7/site-packages (from lazypredict) (4.62.3)
Requirement already satisfied: importlib-metadata in /opt/conda/lib/python3.7/site-packages (from click->lazypredict)
(4.10.1)
Requirement already satisfied: numpy in /opt/conda/lib/python3.7/site-packages (from lightgbm->lazypredict) (1.20.3)
Requirement already satisfied: scipy in /opt/conda/lib/python3.7/site-packages (from lightgbm->lazypredict) (1.7.3)
Requirement already satisfied: wheel in /opt/conda/lib/python3.7/site-packages (from lightgbm->lazypredict) (0.37.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in /opt/conda/lib/python3.7/site-packages (from scikit-learn->lazypredict) (3.0.0)
Requirement already satisfied: python-dateutil>=2.7.3 in /opt/conda/lib/python3.7/site-packages (from pandas->lazypredict) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/lib/python3.7/site-packages (from pandas->lazypredict) (2021.3)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.7/site-packages (from python-dateutil>=2.7.3->pandas->lazypredict) (1.16.0)
Requirement already satisfied: typing-extensions>=3.6.4 in /opt/conda/lib/python3.7/site-packages (from importlib-metadata->click->lazypredict) (4.0.1)
Requirement already satisfied: zipp>=0.5 in /opt/conda/lib/python3.7/site-packages (from importlib-metadata->click->lazypredict) (3.6.0)
Installing collected packages: lazypredict
Successfully installed lazypredict-0.2.12
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
```

```
In [11]: y = train.pop('Class')
X = train
```

```
In [12]: from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, shuffle=True, stratify=y)
```

```
In [13]: from lazypredict.Supervised import LazyClassifier
```

```
In [14]: clf = LazyClassifier(verbose=0, predictions=True)
models, predictions = clf.fit(X_train, X_test, y_train, y_test)
models
```

```
90%|██████████| 26/29 [00:02<00:00, 11.98it/s]
```

```
[21:59:03] WARNING: .../src/learner.cc:1115: Starting in XGBoost 1.3.0, the default evaluation metric used with the objective 'binary:logistic' was changed from 'error' to 'logloss'. Explicitly set eval_metric if you'd like to restore the old behavior.
```

```
100%|██████████| 29/29 [00:03<00:00,  8.90it/s]
```

Out[14]:

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score	Time Taken
BaggingClassifier	0.89	0.89	0.89	0.89	0.15
RandomForestClassifier	0.88	0.88	0.88	0.88	0.49
ExtraTreesClassifier	0.88	0.88	0.88	0.88	0.26
SVC	0.88	0.88	0.88	0.88	0.09
LGBMClassifier	0.88	0.88	0.88	0.88	0.15
RidgeClassifier	0.88	0.87	0.87	0.88	0.02
CalibratedClassifierCV	0.87	0.87	0.87	0.87	0.41
RidgeClassifierCV	0.87	0.87	0.87	0.87	0.03
LinearSVC	0.87	0.87	0.87	0.87	0.15
QuadraticDiscriminantAnalysis	0.87	0.87	0.87	0.87	0.03
AdaBoostClassifier	0.87	0.87	0.87	0.87	0.24
SGDClassifier	0.87	0.87	0.87	0.87	0.03
NuSVC	0.87	0.87	0.87	0.87	0.12
LinearDiscriminantAnalysis	0.87	0.87	0.87	0.87	0.03
LogisticRegression	0.86	0.86	0.86	0.86	0.03
XGBClassifier	0.86	0.86	0.86	0.86	0.47
Perceptron	0.86	0.86	0.86	0.86	0.01
GaussianNB	0.86	0.86	0.86	0.86	0.01
KNeighborsClassifier	0.86	0.86	0.86	0.86	0.03
NearestCentroid	0.85	0.85	0.85	0.85	0.01
BernoulliNB	0.85	0.85	0.85	0.85	0.02
LabelSpreading	0.84	0.84	0.84	0.84	0.20
LabelPropagation	0.84	0.83	0.83	0.84	0.18
ExtraTreeClassifier	0.82	0.82	0.82	0.82	0.01
DecisionTreeClassifier	0.81	0.81	0.81	0.81	0.03
PassiveAggressiveClassifier	0.78	0.78	0.78	0.78	0.02
DummyClassifier	0.49	0.49	0.49	0.49	0.01

In [15]: predictions.head()

	AdaBoostClassifier	BaggingClassifier	BernoulliNB	CalibratedClassifierCV	DecisionTreeClassifier	DummyClassifier	ExtraTreeClassifier	Extral
0	1	1	1		1	1	0	1
1	0	0	0		0	0	1	0
2	0	0	0		0	1	1	1
3	1	1	1		1	1	1	1
4	0	0	0		0	0	0	0

5 rows × 27 columns

```
from sklearn.metrics import classification_report
for i in predictions.columns.tolist():
    print('\t\t', i, '\n')
    print(classification_report(y_test, predictions[i]), '\n')
```

AdaBoostClassifier

	precision	recall	f1-score	support
0	0.87	0.89	0.88	260
1	0.88	0.85	0.86	240
accuracy			0.87	500
macro avg	0.87	0.87	0.87	500
weighted avg	0.87	0.87	0.87	500

BaggingClassifier

	precision	recall	f1-score	support
0	0.86	0.93	0.90	260
1	0.92	0.84	0.88	240
accuracy			0.89	500
macro avg	0.89	0.89	0.89	500
weighted avg	0.89	0.89	0.89	500

BernoulliNB

	precision	recall	f1-score	support
0	0.84	0.87	0.86	260
1	0.86	0.82	0.84	240
accuracy			0.85	500
macro avg	0.85	0.85	0.85	500
weighted avg	0.85	0.85	0.85	500

CalibratedClassifierCV

	precision	recall	f1-score	support
0	0.86	0.91	0.88	260
1	0.90	0.83	0.86	240
accuracy			0.87	500
macro avg	0.88	0.87	0.87	500
weighted avg	0.88	0.87	0.87	500

DecisionTreeClassifier

	precision	recall	f1-score	support
0	0.83	0.80	0.82	260
1	0.80	0.82	0.81	240
accuracy			0.81	500
macro avg	0.81	0.81	0.81	500
weighted avg	0.81	0.81	0.81	500

DummyClassifier

	precision	recall	f1-score	support
0	0.51	0.50	0.50	260
1	0.47	0.48	0.47	240
accuracy			0.49	500
macro avg	0.49	0.49	0.49	500
weighted avg	0.49	0.49	0.49	500

ExtraTreeClassifier

	precision	recall	f1-score	support
0	0.83	0.83	0.83	260
1	0.82	0.81	0.81	240
accuracy			0.82	500
macro avg	0.82	0.82	0.82	500
weighted avg	0.82	0.82	0.82	500

ExtraTreesClassifier

	precision	recall	f1-score	support
0	0.86	0.93	0.89	260
1	0.91	0.84	0.87	240
accuracy			0.88	500
macro avg	0.89	0.88	0.88	500
weighted avg	0.89	0.88	0.88	500

GaussianNB

	precision	recall	f1-score	support
0	0.85	0.89	0.87	260
1	0.87	0.82	0.85	240
accuracy			0.86	500
macro avg	0.86	0.86	0.86	500
weighted avg	0.86	0.86	0.86	500

KNeighborsClassifier

	precision	recall	f1-score	support
0	0.83	0.91	0.87	260
1	0.89	0.80	0.84	240
accuracy			0.86	500
macro avg	0.86	0.86	0.86	500
weighted avg	0.86	0.86	0.86	500

LabelPropagation

	precision	recall	f1-score	support
0	0.82	0.88	0.85	260
1	0.86	0.79	0.82	240
accuracy			0.84	500
macro avg	0.84	0.83	0.84	500
weighted avg	0.84	0.84	0.84	500

LabelSpreading

	precision	recall	f1-score	support
0	0.83	0.87	0.85	260
1	0.85	0.80	0.83	240
accuracy			0.84	500
macro avg	0.84	0.84	0.84	500
weighted avg	0.84	0.84	0.84	500

LinearDiscriminantAnalysis

	precision	recall	f1-score	support
0	0.85	0.90	0.88	260
1	0.89	0.83	0.86	240
accuracy			0.87	500
macro avg	0.87	0.87	0.87	500
weighted avg	0.87	0.87	0.87	500

LinearSVC

	precision	recall	f1-score	support
0	0.85	0.92	0.88	260
1	0.90	0.83	0.86	240
accuracy			0.87	500
macro avg	0.88	0.87	0.87	500
weighted avg	0.88	0.87	0.87	500

LogisticRegression

	precision	recall	f1-score	support
0	0.85	0.90	0.87	260
1	0.88	0.83	0.85	240
accuracy			0.86	500
macro avg	0.87	0.86	0.86	500
weighted avg	0.86	0.86	0.86	500

NearestCentroid

	precision	recall	f1-score	support
0	0.85	0.88	0.86	260
1	0.86	0.83	0.85	240
accuracy			0.85	500
macro avg	0.85	0.85	0.85	500
weighted avg	0.85	0.85	0.85	500

NuSVC

	precision	recall	f1-score	support
0	0.85	0.90	0.88	260
1	0.89	0.83	0.86	240
accuracy			0.87	500
macro avg	0.87	0.87	0.87	500
weighted avg	0.87	0.87	0.87	500

PassiveAggressiveClassifier

	precision	recall	f1-score	support
0	0.84	0.70	0.77	260
1	0.73	0.85	0.79	240
accuracy			0.78	500
macro avg	0.78	0.78	0.78	500
weighted avg	0.79	0.78	0.78	500

Perceptron

	precision	recall	f1-score	support
0	0.84	0.91	0.87	260
1	0.89	0.81	0.85	240
accuracy			0.86	500
macro avg	0.87	0.86	0.86	500
weighted avg	0.87	0.86	0.86	500

QuadraticDiscriminantAnalysis

	precision	recall	f1-score	support
0	0.84	0.93	0.88	260
1	0.92	0.81	0.86	240
accuracy			0.87	500
macro avg	0.88	0.87	0.87	500
weighted avg	0.88	0.87	0.87	500

RandomForestClassifier

	precision	recall	f1-score	support
0	0.87	0.92	0.89	260
1	0.90	0.85	0.88	240
accuracy			0.88	500
macro avg	0.89	0.88	0.88	500
weighted avg	0.88	0.88	0.88	500

RidgeClassifier

	precision	recall	f1-score	support
0	0.85	0.92	0.89	260
1	0.90	0.83	0.87	240
accuracy			0.88	500
macro avg	0.88	0.87	0.88	500
weighted avg	0.88	0.88	0.88	500

RidgeClassifierCV

	precision	recall	f1-score	support
0	0.86	0.91	0.88	260
1	0.90	0.83	0.86	240
accuracy			0.87	500
macro avg	0.88	0.87	0.87	500
weighted avg	0.88	0.87	0.87	500

SGDClassifier

	precision	recall	f1-score	support
0	0.82	0.97	0.89	260
1	0.96	0.77	0.85	240
accuracy			0.87	500
macro avg	0.89	0.87	0.87	500

weighted avg 0.89 0.87 0.87 500

SVC

	precision	recall	f1-score	support
0	0.85	0.92	0.89	260
1	0.91	0.83	0.87	240
accuracy			0.88	500
macro avg	0.88	0.88	0.88	500
weighted avg	0.88	0.88	0.88	500

XGBClassifier

	precision	recall	f1-score	support
0	0.85	0.90	0.87	260
1	0.89	0.82	0.85	240
accuracy			0.86	500
macro avg	0.87	0.86	0.86	500
weighted avg	0.87	0.86	0.86	500

LGBMClassifier

	precision	recall	f1-score	support
0	0.85	0.93	0.89	260
1	0.91	0.82	0.87	240
accuracy			0.88	500
macro avg	0.88	0.88	0.88	500
weighted avg	0.88	0.88	0.88	500