Рубежный контроль №2

Дюжев С.А. Группа ИУ5Ц-83Б (ГУИМЦ)

Вариант 27

Задача. Для заданного набора данных (по Вашему варианту) постройте модели классификации или регрессии (в зависимости от конкретной задачи, рассматриваемой в наборе данных). Для построения моделей используйте методы 1 и 2 (по варианту для Вашей группы). Оцените качество моделей на основе подходящих метрик качества (не менее двух метрик). Какие метрики качества Вы использовали и почему? Какие выводы Вы можете сделать о качестве построенных моделей? Для построения моделей необходимо выполнить требуемую предобработку данных: заполнение пропусков, кодирование категориальных признаков, и т.д.

Методы для ИУ5-63Б. Метод №1: "Дерево решений". Метод №2: "Случайный лес".

Набор данных: Company Bankruptcy Prediction.

Импорт библиотек:

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
```

Загрузка и первичная подготовка данных:

```
1 # загрузка набора данных (файл: impeachment-polls.csv)
2 data = pd.read_csv('data.csv', sep=",")
3 # размер набора данных
4 data.shape
```

```
(6819, 96)
```

```
1 # первые 5 строк набора данных
2 data.head(5)
```

	Bankrupt?	ROA(C) before interest and depreciation before interest	ROA(A) before interest and % after tax	ROA(B) before interest and depreciation after tax	Operating Gross Margin	Realized Sales Gross Margin	Opera Pr
0	1	0.370594	0.424389	0.405750	0.601457	0.601457	0.9
1	1	0.464291	0.538214	0.516730	0.610235	0.610235	0.9

```
1 data.info()
```

```
 40   Borrowing dependency                                    6819 non-nul
 41   Contingent liabilities/Net worth                        6819 non-nul
 42   Operating profit/Paid-in capital                        6819 non-nul
 43   Net profit before tax/Paid-in capital                   6819 non-nul
 44   Inventory and accounts receivable/Net value             6819 non-nul
 45   Total Asset Turnover                                    6819 non-nul
 46   Accounts Receivable Turnover                            6819 non-nul
 47   Average Collection Days                                 6819 non-nul
 48   Inventory Turnover Rate (times)                         6819 non-nul
 49   Fixed Assets Turnover Frequency                         6819 non-nul
 50   Net Worth Turnover Rate (times)                         6819 non-nul
 51   Revenue per person                                      6819 non-nul
 52   Operating profit per person                             6819 non-nul
 53   Allocation rate per person                              6819 non-nul
 54   Working Capital to Total Assets                         6819 non-nul
 55   Quick Assets/Total Assets                               6819 non-nul
 56   Current Assets/Total Assets                             6819 non-nul
 57   Cash/Total Assets                                       6819 non-nul
 58   Quick Assets/Current Liability                          6819 non-nul
 59   Cash/Current Liability                                  6819 non-nul
 60   Current Liability to Assets                             6819 non-nul
 61   Operating Funds to Liability                            6819 non-nul
 62   Inventory/Working Capital                               6819 non-nul
 63   Inventory/Current Liability                             6819 non-nul
 64   Current Liabilities/Liability                           6819 non-nul
 65   Working Capital/Equity                                  6819 non-nul
 66   Current Liabilities/Equity                              6819 non-nul
 67   Long-term Liability to Current Assets                   6819 non-nul
 68   Retained Earnings to Total Assets                       6819 non-nul
 69   Total income/Total expense                              6819 non-nul
 70   Total expense/Assets                                    6819 non-nul
 71   Current Asset Turnover Rate                             6819 non-nul
 72   Quick Asset Turnover Rate                               6819 non-nul
 73   Working capitcal Turnover Rate                          6819 non-nul
 74   Cash Turnover Rate                                      6819 non-nul
 75   Cash Flow to Sales                                      6819 non-nul
 76   Fixed Assets to Assets                                  6819 non-nul
 77   Current Liability to Liability                          6819 non-nul
 78   Current Liability to Equity                             6819 non-nul
 79   Equity to Long-term Liability                           6819 non-nul
 80   Cash Flow to Total Assets                               6819 non-nul
 81   Cash Flow to Liability                                  6819 non-nul
 82   CFO to Assets                                           6819 non-nul
 83   Cash Flow to Equity                                     6819 non-nul
 84   Current Liability to Current Assets                     6819 non-nul
 85   Liability-Assets Flag                                   6819 non-nul
```

```
 85   Liability-Assets Flag                                    6819 non-nul
 86   Net Income to Total Assets                               6819 non-nul
 87   Total assets to GNP price                                6819 non-nul
 88   No-credit Interval                                       6819 non-nul
 89   Gross Profit to Sales                                    6819 non-nul
 90   Net Income to Stockholder's Equity                       6819 non-nul
 91   Liability to Equity                                      6819 non-nul
 92   Degree of Financial Leverage (DFL)                       6819 non-nul
 93   Interest Coverage Ratio (Interest expense to EBIT)       6819 non-nul
 94   Net Income Flag                                          6819 non-nul
 95   Equity to Liability                                      6819 non-nul
dtypes: float64(93), int64(3)
memory usage: 5.0 MB
```

```
count = 0
not_scaled = []
for col in data.columns:
    if max(data[col])>1:
        print("not scaled : ", col)
        count += 1
        not_scaled.append(col)
```

```
not scaled :   Operating Expense Rate
not scaled :   Research and development expense rate
not scaled :   Interest-bearing debt interest rate
not scaled :   Revenue Per Share (Yuan ¥)
not scaled :   Total Asset Growth Rate
not scaled :   Net Value Growth Rate
not scaled :   Current Ratio
not scaled :   Quick Ratio
not scaled :   Total debt/Total net worth
not scaled :   Accounts Receivable Turnover
not scaled :   Average Collection Days
not scaled :   Inventory Turnover Rate (times)
not scaled :   Fixed Assets Turnover Frequency
not scaled :   Revenue per person
not scaled :   Allocation rate per person
not scaled :   Quick Assets/Current Liability
not scaled :   Cash/Current Liability
not scaled :   Inventory/Current Liability
not scaled :   Long-term Liability to Current Assets
not scaled :   Current Asset Turnover Rate
not scaled :   Quick Asset Turnover Rate
not scaled :   Cash Turnover Rate
not scaled :   Fixed Assets to Assets
not scaled :   Total assets to GNP price
```

```
len(not_scaled)
```

```
24
```

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
data[not_scaled] = scaler.fit_transform(data[not_scaled])
```

```
1 data.head()
```

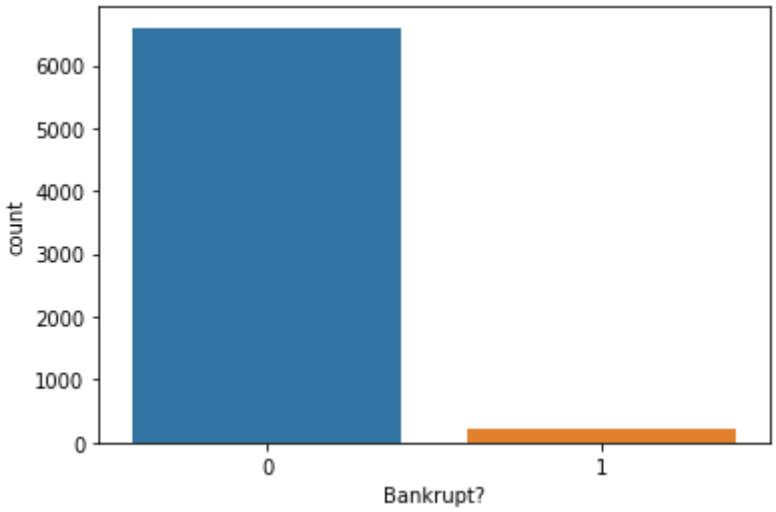
	Bankrupt?	ROA(C) before interest and depreciation before interest	ROA(A) before interest and % after tax	ROA(B) before interest and depreciation after tax	Operating Gross Margin	Realized Sales Gross Margin	Opera Pr
0	1	0.370594	0.424389	0.405750	0.601457	0.601457	0.9
1	1	0.464291	0.538214	0.516730	0.610235	0.610235	0.9
2	1	0.426071	0.499019	0.472295	0.601450	0.601364	0.9
3	1	0.399844	0.451265	0.457733	0.583541	0.583541	0.9
4	1	0.465022	0.538432	0.522298	0.598783	0.598783	0.9

5 rows × 96 columns

```
1 import seaborn as sns
```

```
1 sns.countplot(data['Bankrupt?'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarni
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7fa733f75e90>
```




```
1 X = data.drop('Bankrupt?', axis = 1)
2 y = data['Bankrupt?']
```

```
1 !sudo pip install imbalanced-learn
```

```
Requirement already satisfied: imbalanced-learn in /usr/local/lib/python3.7/d
Requirement already satisfied: numpy>=1.8.2 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: scikit-learn>=0.20 in /usr/local/lib/python3.7
Requirement already satisfied: scipy>=0.13.3 in /usr/local/lib/python3.7/dist
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.7/dist-
```

```
1 import imblearn
```

```
1 sm = SMOTE()
2 X_sm,y_sm = sm.fit_resample(X,y)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: Futur
  warnings.warn(msg, category=FutureWarning)
```

```
1
```

```
array([1, 1, 1, ..., 1, 1, 1])
```

Обработка пропусков в данных:

Построение моделей:

Разделение выборки на обучающую и тестовую

```
1 from sklearn.model_selection import train_test_split
```

```
1 data_train, data_test, data_y_train, data_y_test = train_test_split(X_sm, y_sm,
```

Модель "Дерево решений"

```
1 from sklearn.tree import DecisionTreeRegressor
```

```
1 dtc = DecisionTreeRegressor(random_state=1).fit(data_train, data_y_train)
2 data_test_predicted_dtc = dtc.predict(data_test)
```

Модель "Случайный лес"

```
1 from sklearn.ensemble import RandomForestRegressor
```

```
1 RF = RandomForestRegressor(random_state=1).fit(data_train, data_y_train)
```

```
2 data_test_predicted_rf = RF.predict(data_test)
```

Оценка качества моделей:

В качестве метрик для оценки качества моделей я использую **Mean squared error** (средняя квадратичная ошибка), как наиболее часто используемую метрику для оценки качества регрессии, и **метрику** R^2 (коэффициент детерминации), потому что эта метрика является нормированной.

```
1 from sklearn.metrics import mean_squared_error, r2_score
```

```
1 # Mean squared error - средняя квадратичная ошибка
2 print('Метрика MSE:\nДерево решений: {}\nСлучайный лес: {}'.format(mean_squared
```

```
Метрика MSE:
Дерево решений: 7.644969145389171
Случайный лес: 3.714740789719621
```

```
1 # 4) Метрика R2 или коэффициент детерминации
2 print('Метрика R\u00B2:\nДерево решений: {}\nСлучайный лес: {}'.format(r2_score
```

```
Метрика R²:
Дерево решений: 0.809032930682008
Случайный лес: 0.9131037694525025
```

Выводы о качестве построенных моделей:

Исходя из оценки качества построенных моделей можно увидеть, что модель "Случайный лес" лучше справляется с задачей по сравнению с моделью "Дерево решений", что может свидетельствовать о переобучении модели "Дерево решений".

✓ 0s completed at 4:11 PM