Многоклассовая классификация на примере датасета Wine Quality

1. Введение

Набор данных был загружен из репозитория машинного обучения UCI.

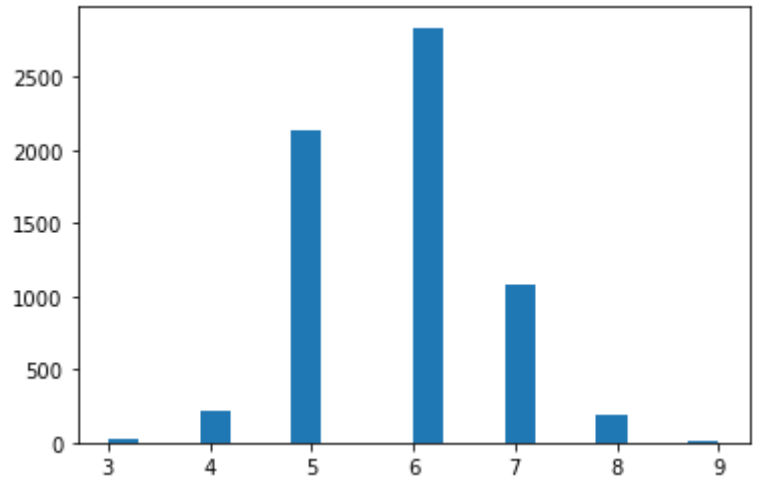
Два набора данных относятся к красному и белому вариантам португальского вина «Vinho Verde». Из-за проблем с конфиденциальностью и логистикой доступны только физико-химические (входные) и органолептические (выходные) переменные (например, нет данных о сортах винограда, марке вина, продажной цене вина и т. д.).

Задачей стоит классификация вина по его качеству “quality” (от плохого 0 до отличного 10)

2. Описание набора данных

2.1 Классы упорядочены и не сбалансированы (например, нормальных вин намного больше, чем отличных или плохих). Визуальное распределение классов можно увидеть на Графике 1.

График 1. Распределение классов таргетной переменной



2.2 Датасет состоит из 12 признаков, 7 из них имеют пропуски в данных, подробнее ознакомиться можно в Таблице 1.

Таблица 1. Кол-во значений по признакам

Признак	Кол-во значений	Процент заполненных данных
type	6497	100.00 %
fixed acidity	6487	99.84 %
volatile acidity	6489	99.87 %
citric acid	6494	99.95 %
residual sugar	6495	99.96 %
chlorides	6495	99.96 %
free sulfur dioxide	6497	100.00 %
total sulfur dioxide	6497	100.00 %
density	6497	100.00 %
pH	6488	99.86 %
sulphates	6493	99.93 %
alcohol	6497	100.00 %

3. Признаки и предобработка

3.1 Категориальный признак Type преобразуем через LabelEncoder

3.2 Избавимся от пропусков, заменив их медианными значениями. При анализе данных было выявлено, что большинство признаков имеют выбросы в значениях. Среднее значение сильно зависит от признаков, что нельзя сказать про медианное, поэтому для замены пропусков выбираем его.

3.3 Стандартизируем значения признаков, так как в данный момент признаки между собой имеют огромный разброс значений

4. Обучение модели

Для классификации будем использовать RandomForest.

В первом варианте данные будем разбивать на Кросс Валидации, используя StratifiedKFold, который разобьет все наши классы пропорционально на foldы.

Во втором варианте обучать модель будем на RandomizedSearchCV для лучшего подбора гиперпараметров

5. Результаты

При обучении и подборе гиперпараметров RandomizedSearchCV лучшая модель показала Скор на тестовых данных: 0.656028
И Скор на тренировочных данных : 0.987904

Скор по всем моделям можно посмотреть в Таблице 2.

Таблица 2. Варианты значений гиперпараметров и скор на тестовых данных

n_estimators	min_samples leaf	max_features	max_depth	criterion	mean_test_score
200	2	8	15	entropy	0.656028
25	5	5	15	entropy	0.623925
25	5	10	19	gini	0.623710
200	5	5	11	gini	0.619749
200	1	10	9	entropy	0.617547
50	8	10	19	gini	0.615788
100	9	5	9	gini	0.595996
100	6	9	7	entropy	0.579060
150	9	8	7	gini	0.577080
25	9	13	14	entropy	Nan

6. Доработки модели

Как мы видим, Рандомный лес переобучился на тренировочных данных. Для доработки я бы предложил использовать полносвязную нейронную сеть со слоями DropOut для предотвращения переобучения.