# Разработка моделей классификации качества вина

Сентябрь 2023 г.

# Оглавление

# Описание данных

**Таб. 1. Характеристики датасета**

```
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   type                  6497 non-null   object
 1   fixed acidity         6487 non-null   float64
 2   volatile acidity      6489 non-null   float64
 3   citric acid           6494 non-null   float64
 4   residual sugar        6495 non-null   float64
 5   chlorides             6495 non-null   float64
 6   free sulfur dioxide   6497 non-null   float64
 7   total sulfur dioxide  6497 non-null   float64
 8   density               6497 non-null   float64
 9   pH                    6488 non-null   float64
 10  sulphates             6493 non-null   float64
 11  alcohol               6497 non-null   float64
 12  quality               6497 non-null   int64
dtypes: float64(11), int64(1), object(1)
memory usage: 660.0+ KB
```

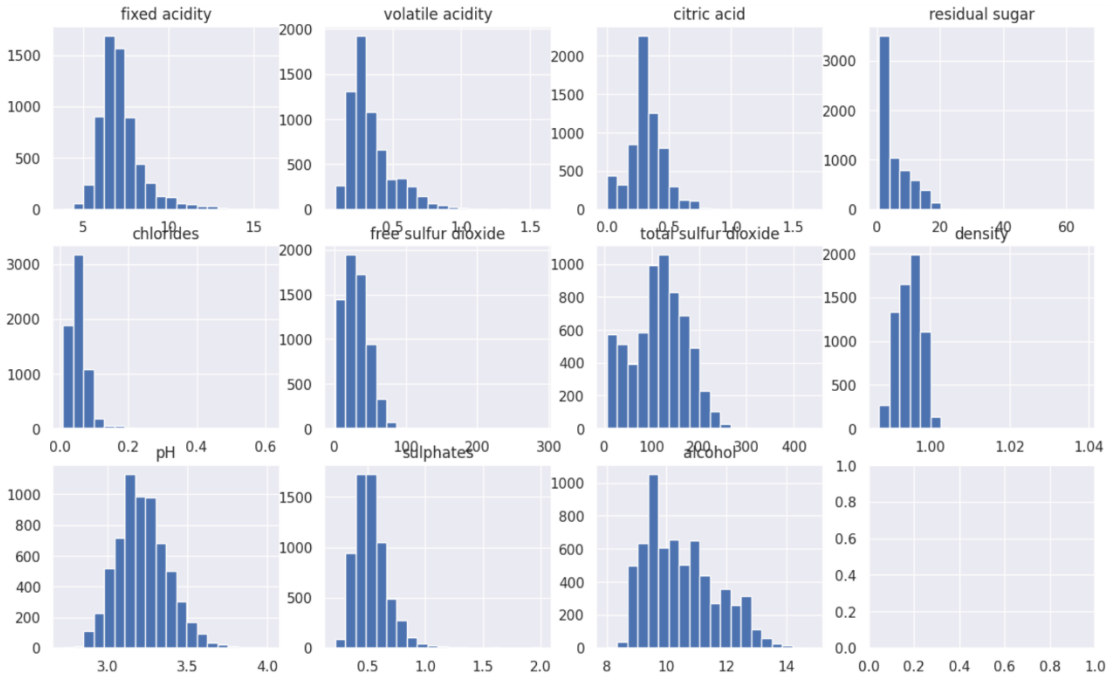
- **Общие характеристки:** Данные состоят из 12 столбцов, из которых 11 столбцов – это исходные данные (из которых первый столбец – тип вина, строковое значение, остальные - вещественные), последний (quality) – это классификация качества вина. Размерность данных – **6 497 строк**
- **Качество данных**: Из 10 столбцов числовых значений 7 имеют пропущенные значения. К ним относятся: 'fixed acidity', 'pH', 'sulphates‘, 'volatile acidity', 'citric acid', 'residual sugar', 'chlorides'

**Целью задачи** является прогноз качества вина на основе значений его химических характеристик.

# Обработка данных.

Часть 1

**Рис. 1. Виды распределений входных параметров датасета**




- Анализ 7 столбцов, которые имеют пропущенные значения, показывает, что 3 из них: **fixed acidity, pH, sulphates** - близки к симметричным распределениям, поэтому пропущенные значения лучше заполнить средним. Остальные **'volatile acidity', 'citric acid', 'residual sugar', 'chlorides'** – ассиметричны, поэтому их лучше заполнить модой

# Обработка данных.

Часть 2

**Таб. 2. Распределение вина типам**

| Wine type | Quantity |
|---|---|
| red | 1599 |
| white | 4898 |

**Таб. 3. Распределение вин по классам**

| Wine class | Quantity |
|---|---|
| 6 | 2 824 |
| 5 | 2 134 |
| 7 | 1 077 |
| 4 | 215 |
| 8 | 193 |
| 3 | 28 |
| 9 | 5 |

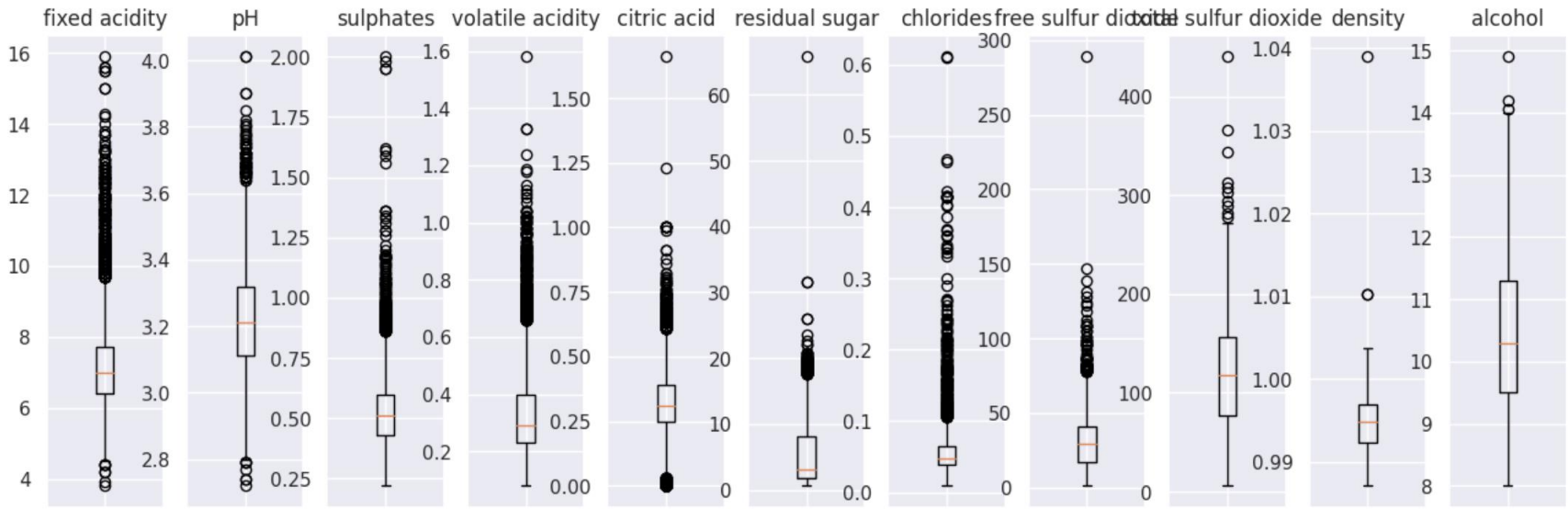
- Переменная входного параметра «wine type» преобразуется с помощью LabelEncoding

- Анализ частот распределения классов выходного показателя классификации показывает, что классы несбалансированы. Наименее часто встречающийся класс – 9 встречается всего 5 раз, в то время как наиболее частый – класс 6 встречается 2824 раза. Для устранения несбалансированности был применен метод oversampling (Synthetic Minority Oversampling Technique - SMOTE)

# Обработка данных.

Часть 3

**Рис. 2. Анализ входных данных на выбросы**




- Анализ на выбросы показывает, что к ним можно отнести точки в столбце 'chlorides’ со значением > 0.5, 'residual sugar‘ со значением > 60, 'free sulfur dioxide‘ со значением > 200, 'density’ со значением > 1.01, 'alcohol‘со значением > 14, 'citric acid‘со значением > 0.98), 'sulphates‘ со значением > 1.95), 'volatile acidity‘ ] со значением > 1.5), 'residual sugar‘ со значением > 25)
- Всего таких точек 21, что составляет 0.3% от общей выборки. Таким образом удаление этих данных не окажет существенное влияние на качество данных

# Подбор оптимальной модели классификации

- Данные разбиваются на 2 группы (train, test). Размер train – 80% от всей выборки, test – 20%
- Для выбора оптимальной модели был проведен сравнительный анализ 3 моделей: RandomForestClassifier, ExtraTreeClassifier, SVC.
- В свою очередь для каждой из моделей с помощью GridSearchCV был проведен поиск оптимальных параметров, результаты поиска приведены в таблице ниже

**Таб. 4. Показатели качества моделей классификации**

| | model | model parameters | accuray train | accuracy test |
|---|---|---|---|---|
| 0 | (DecisionTreeClassifier(max_features='sqrt', r... | {'criterion': 'gini', 'n_estimators': 100, 'ra... | 0.882596 | 0.644290 |
| 1 | ExtraTreeClassifier(random_state=13) | {'criterion': 'gini', 'random_state': 13} | 0.786828 | 0.539352 |
| 2 | SVC(kernel='poly', random_state=13) | {'kernel': 'poly', 'random_state': 13} | 0.364747 | 0.132716 |

- Как видно из приведенной таблицы наилучшими параметрами обладает RandomForestClassifier, который на тестовой выборке дает точность **64%**