

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МОЭВМ

ОТЧЕТ
по лабораторной работе №7
по дисциплине «Машинное обучение»

Студенты гр. 6304

Преподаватель

Тимофеев А.А.

Жангиров Т.Р.

Санкт-Петербург

2020

Цель работы

Ознакомиться с методами классификации модуля Sklearn

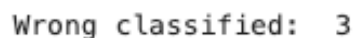
Ход работы

Загрузка данных

1. Был создан датафрейм Pandas на основе загруженного датасета (<https://archive.ics.uci.edu/ml/datasets/iris>)
2. Были выделены данные и их метки, тексты меток были преобразованы к числам при помощи *LabelEncoder*.
3. Выборка была разбита на обучающую и тестовую при помощи *train_test_split*.

Байесовские методы

1. Была проведена классификация данных методом *GaussianNB*, выведено количество неправильно классифицированных наблюдений (представлено на рисунке 1).



```
Wrong classified: 3
```

Рисунок 1 – Количество неправильно классифицированных наблюдений

2. С помощью метода *score* была получена точность классификации, которая составила 96%.
3. Описание атрибутов метода *GaussianNB* представлено в таблице 1.

Таблица 1 – Описание атрибутов метода *GaussianNB*

Название	Описание	Тип возвращаемого значения
<code>class_count_</code>	Количество наблюдений в обучающих выборках для каждого класса	<code>ndarray of shape (n_classes,)</code>

class_prior_	Вероятность встречи наблюдения для каждого класса	ndarray of shape (n_classes,)
classes_	Метки класса известные классификатору	ndarray of shape (n_classes,)
epsilon_	Величина аддитивной дисперсии	float
sigma_	Дисперсия каждого признака по классу	ndarray of shape (n_classes, n_features)
theta_	Среднее каждого признака по классу	ndarray of shape (n_classes, n_features)

4. Были построены графики зависимости неправильно классифицированных наблюдений и точности классификации от размера тестовой выборки для метода *GaussianNB*. Графики представлен на рисунке 2.

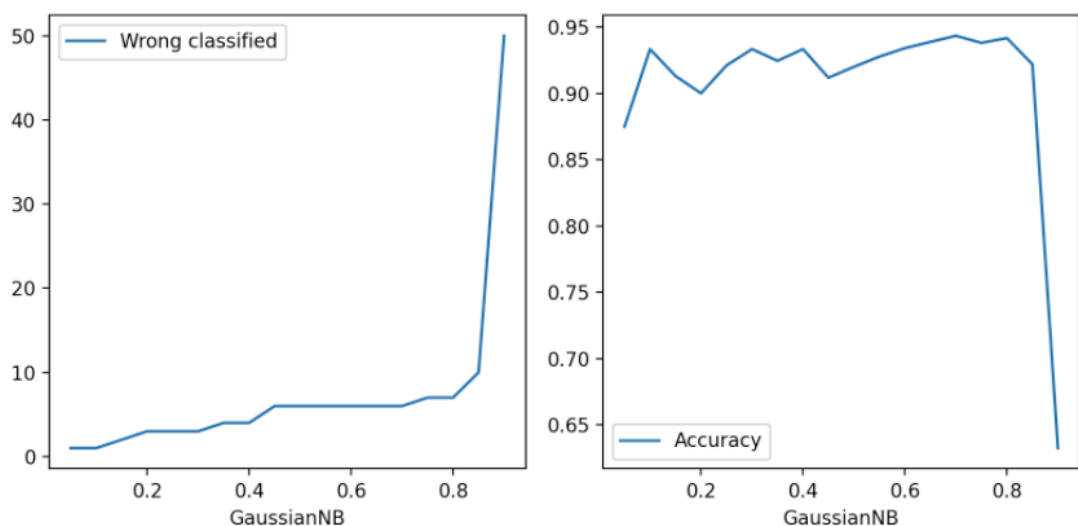


Рисунок 2 – Графики для метода *GaussianNB*

Точность классификации не падает с увеличением тестовой выборки вплоть до 90% от всей выборки. Скорее всего такая хорошая классифицируемость данных связана с их распределением в выборке.

5. Была проведена классификация другими байесовскими классификаторами, представленными в модуле Sklearn. Лучшие результаты классификации представлены в таблице 2.

Метод	Размер тестовой выборки	Кол-во неправильно класс. данных	Точность
<i>GaussianNB</i>	0.7	6	0.94
<i>MultinomialNB</i>	0.25	1	0.97
<i>ComplementNB</i>	0.2	9	0.7
<i>BernoulliNB</i>	0.8	80	0.33

Наилучший результат показал метод *MultinomialNB*.

В методе *MultinomialNB* распределение для каждого класса параметризуется векторами, содержащими вероятности вхождения признаков в элемент выборки, соответствующий данному классу.

Метод *ComplementNB* – это адаптация стандартного полиномиального наивного байесовского алгоритма (*MNB*), который особенно подходит для несбалансированных наборов данных. В частности, *CNB* использует статистику из дополнения каждого класса для вычисления весов модели.

BernoulliNB реализует наивные байесовские алгоритмы для данных, которые распределяются согласно многомерному распределению Бернулли; предполагается, что каждый признак является двоичной (логической) переменной.

Классифицирующие деревья

1. Была проведена классификация данных методом *DecisionTreeClassifier*, выведено количество неправильно классифицированных наблюдений (представлено на рисунке 3).

Wrong classified: 4

Рисунок 3 – Количество неправильно классифицированных наблюдений

2. С помощью метода *score* была получена точность классификации, которая составила 95%.
3. Были выведены количество листьев и глубина с помощью функций *get_n_leaves* и *get_depth* соответственно (представлено на рисунке 4).

Num of leaves: 6
Depth: 4

Рисунок 4 – Количество листьев и глубина

4. Было выведено изображение полученного дерева (представлено на рисунке 5).

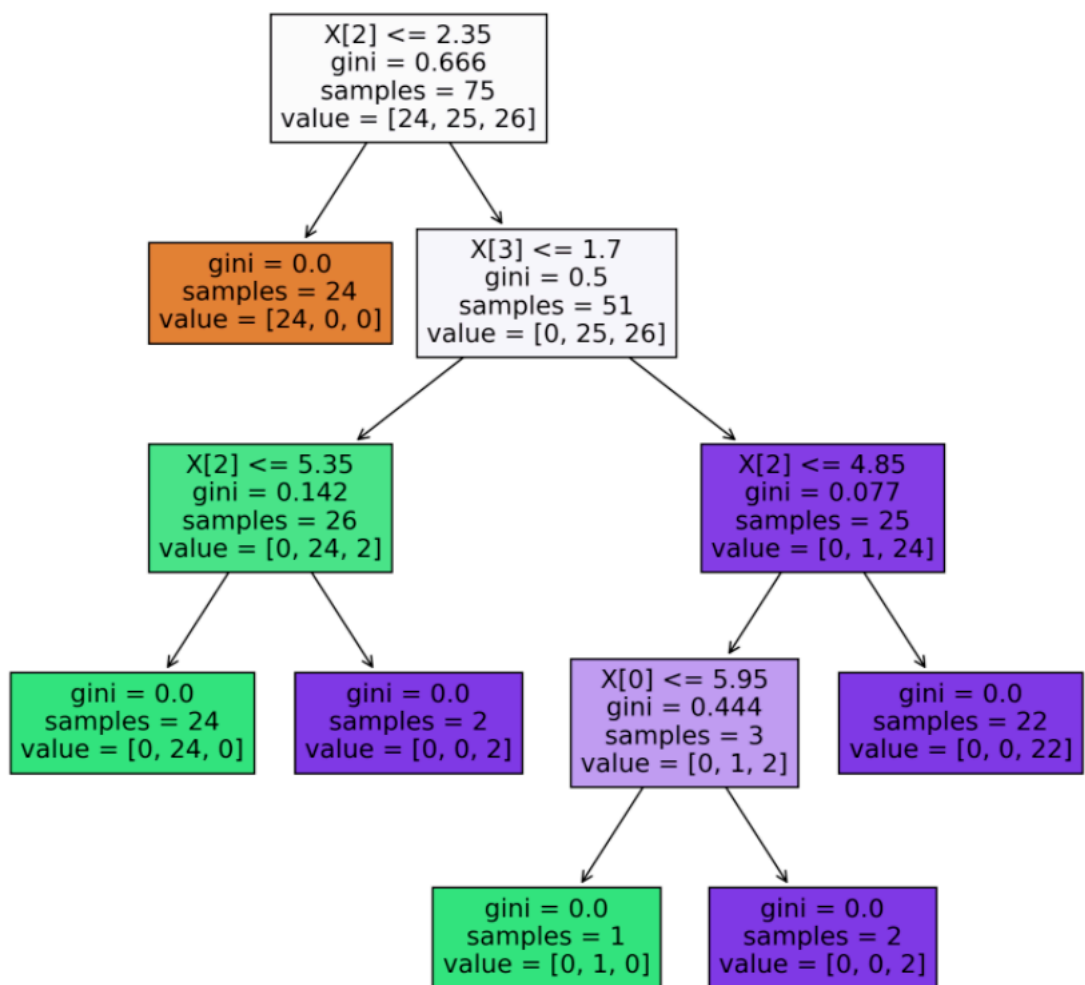


Рисунок 5 – Изображение дерева

Для каждого узла на самой верхней строке указывается условие для разбиения. Далее на каждом листе следует значение примеси Джини, количество наблюдений в узле/листе, а также распределение узлов по

классам. Чем больше объектов в узле/листе принадлежит одному классу, тем насыщеннее его цвет.

5. Были построены графики зависимости неправильно классифицированных наблюдений и точности классификации от размера тестовой выборки для метода *DecisionTreeClassifier*. Графики представленные на рисунке 6.

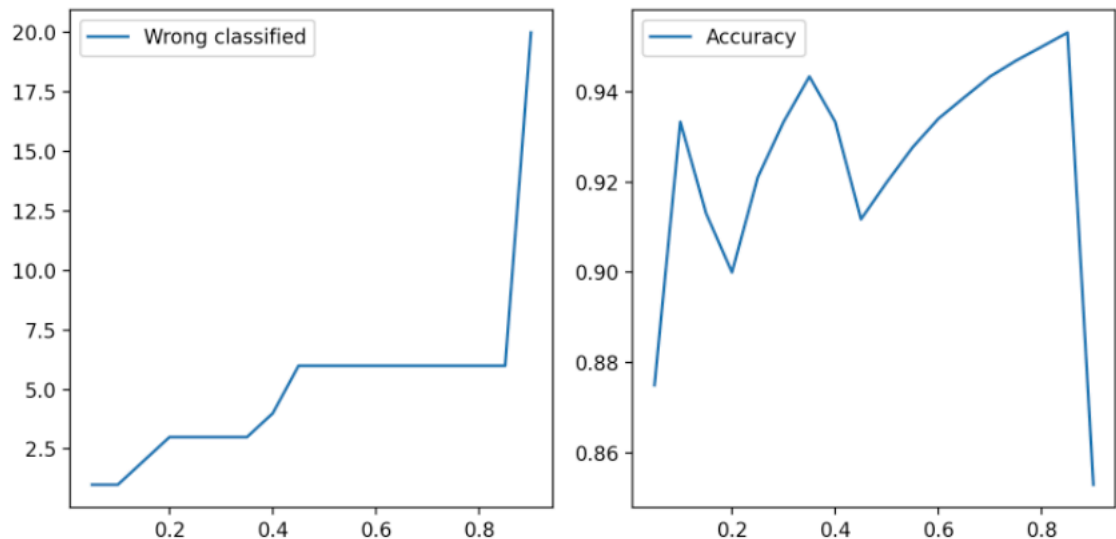


Рисунок 6 – Графики для метода *DecisionTreeClassifier*

Слабая зависимость результатов классификации от размера тестовой выборки как и в случае с байесовским классификатором подтверждает хорошую классифицируемость данных выборки.

6. Была исследована работа классифицирующего алгоритма при различных значениях параметров *criterion*, *splitter*, *max_depth*, *min_samples_split*, *min_samples_leaf*.

- a. *Criterion*

Отвечает за функцию для измерения качества разбиения. Критерием может быть или примесь Джини, или энтропия. Для обоих значений получились идентичные результаты классификации.

- b. *Splitter*

Отвечает за стратегию, используемую для выбора разделения в каждом узле. Можно выбрать или наилучшее разбиение, или наилучшее случайное разбиение. Результаты классификации при обоих значениях примерно равны (учитывая нестабильность метода).

c. `Max_depth`

Отвечает за максимальную глубину дерева. При значении 1 результат классификации заметно ухудшился, так как такой глубины недостаточно для классификации выборки. При значении 2 и выше были показаны идентичные результаты классификации.

d. `Min_samples_split`

Отвечает за минимальное число наблюдений необходимых для разбиения внутреннего узла. С увеличением значения наблюдается ухудшение классификации, однако оно не значительно в виду того, что данные выборки хорошо классифицируемы, а также для классификации достаточно небольшого количества уровней дерева.

e. `Min_samples_leaf`

Отвечает за минимальное число наблюдение, требующееся для конечного узла. Рост значения сильно сказывается на результате классификации, так как параметр начинает сильно влиять на процесс разделения, заставляя оставлять в конечных узлах большее количество наблюдений.

Выводы

В ходе выполнения данной лабораторной работы было произведено знакомство с методами классификации модуля Sklearn. Классификация производилась с помощью методов *GaussianNB*, *MultinomialNB*, *ComplementNB*, *BernoulliNB* и *DecisionTreeClassifier*.