УДК 004.624

Классификатор n-ближайших соседей на примере сортов ириса с использованием библиотеки sklearn

Кизянов Антон Олегович
Приамурский государственный университет имени Шолом-Алейхема
Студент

Научный руководитель: Баженов Руслан Иванович
Приамурский государственный университет им.Шолом-Алейхема
к.п.н, доцент, зав.кафедрой информационных систем, математики и правовой информатики

Аннотация
В данной статье описан метод n-ближайших соседей и разобран пример использования его на классификации ирисов. Для анализа применялись слудеющие математические модули: sklearn, sympy, numpy. Данное описание позволит лучше разобраться при использовании этого метода при анализе других данных.
Ключевые слова: Python, sklearn, машинное обучение

Classifier n-closest neighbors on the example of varieties of iris using the library sklearn

Kizyanov Anton Olegovich
Sholom-Aleichem Priamursky State University
Student

Scientific director: Bazhenov Ruslan Ivanovich Sholom-Aleichem Priamursky State University Candidate of pedagogical sciences, associate professor, Head of the Department of Information Systems, Mathematics and legal informatics

Abstract
This article describes the method of n-nearest neighbors and disassembled an example of its use on the classification of irises. The following mathematical modules were used for the analysis: sklearn, sympy, numpy. This description will allow you to better understand when using this method when analyzing other data.
Keywords: Python, sklearn, machine learning

Построение систем машинного обучения является на сегодняшний день одной из самых популярных, актуальных и современных областей человеческой деятельности на стыке информационных технологий, математического анализа и статистики. Машинное обучение все глубже проникает в нашу жизнь посредством пользовательских продуктов,

созданных с помощью методов искусственного интеллекта. Очевидно, что данные технологии будут развиваться и дальше, постепенно становясь частью повседневной рутины во многих областях человеческой профессиональной деятельности. Однако со времен своего появления, машинное обучение успело обзавестись многочисленными проблемами, главная из которых - достаточно высокая трудоемкость. Построение систем машинного обучения требует огромного количества времени высокопрофессиональных специалистов как в сфере искусственного интеллекта, так и в той предметной области, к которой эта технология применяется.

Цель исследования – написать классификатор определения сорта ириса с использованием метода n-ближайших соседей.

Ранее этим вопросом интересовался М.С. Артамошкин развивал тему «Принятие решений в электронной коммерции с использованием библиотеки scikit-learn» [1] в которой приведен краткий обзор Python-библиотеки Scikit-learn, с примерами ее использования. Также рассматривается ее применение на реальном примере, где проводится анализ клиентов интернет-магазина для оптимизации продаж. Тему «Классификация животных методом машинного обучения k-ближайших соседей» [2], разобрала А.А. Третьякова и описала классический метод машинного обучения с учителем k-ближайших соседей. На данных о видах животных и об характеристиках животных построена модель. Данная модель построена на части известных данных. Остальные данные используются для проверки адекватности модели. М.В. Коротеев опубликовал статью «Обзор некоторых современных тенденций в технологии машинного обучения» [3] рассказал про основные новации в области методологии машинного обучения, которые могут оказать значительное влияние на развитие данной отрасли. Выполнен анализ современной научной литературы, посвященной вопросам развития методологии и областей прикладного использования рассматриваемых тем. Так же А.А. Ярыгин в своей статье «Актуальные вопросы машинного обучения с подкреплением интеллектуальных агентов в задачах принятия решений» [4] расписал подробнее тему машинного обучения, в связи с широким проникновением интеллектуальных автоматизированных систем во многие сферы человеческой деятельности. Данные методы используются для решения следующих задач: адаптивное управление роботами, биржевой технический анализ, промышленная интеллектуальная автоматизация, идентификация личности, проектирование автомобильных автопилотов и многие другие. К.С. Сидоров, и Р.К. Ахунжанов в статье «Разработка и внедрение методических материалов к курсу по машинному обучению в астраханском государственном университете» описали проблемы машинного обучения в Астраханском государственном университете и необходимость внедрения адаптированных методических материалов к курсу «Машинное обучение» [5]. Упомянуты современные тенденции машинного обучения и их связь с прикладными задачами как информатики,

так и других областей науки. Проблемы обучения новых кадров для построения экспертных систем разобрали следующие авторы А.В. Шмид, К.А. Лычагин в статье «Машинное обучение в экспертных системах: подготовка специалистов» [6]. Много статей есть и на английском языке, например «Machine learning by imitating human learning»[7] авторов K.Ch. Chang, T.P. Hong, Sh.Sh. Tseng про изучение общих понятий в несовершенных средах, поскольку учебные примеры часто включают в себя шумные данные, неубедительные данные, неполные данные, неизвестные атрибуты, неизвестные значения атрибутов и другие препятствия для эффективного обучения.

Sklearn – это бесплатная библиотека машинного обучения для языка программирования Python. Она имеет различные алгоритмы классификации, регрессии и кластеризации, включая машины опорных векторов, случайные леса, повышение градиента, k-средних и DBSCAN, и предназначена для взаимодействия с числовыми и научными библиотеками Python NumPy и SciPy.[8]

KNN (K-ближайших соседей) — это простой контролируемый алгоритм классификации, который может быть использован для регрессии, KNN не делает никаких предположений о распределении данных, поэтому он является непараметрическим. Он сохраняет все обучающие данные для будущих прогнозов, вычисляя сходство между входной выборкой и каждым обучающим экземпляром.

Алгоритм K-ближайшего соседа, по существу, сводится к формированию большинства голосов между K наиболее похожими случаями с данным «невидимым» наблюдением. Сходство определяется в соответствии с метрикой расстояния между двумя точками данных.

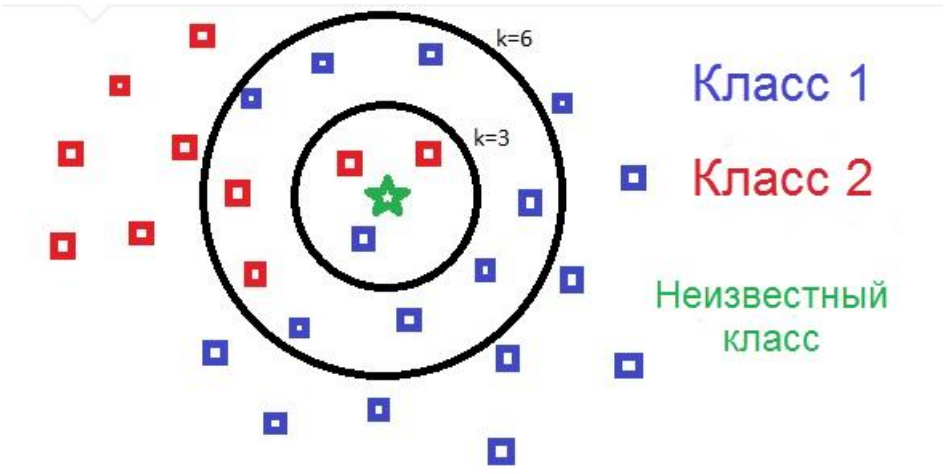

Рис. 1 Визуализация работы метода KNN

При K = 3 будет назначен класс 2, при K = 6 будет назначен класс 1.

Ниже приведен пример реализации KNN в наборе данных iris[9] с использованием библиотеки scikit-learn. Набор данных Iris содержит 50 образцов для каждого вида цветов Iris (всего 150). Для каждого образца у

нас есть длина чашелистика, ширина и длина и ширина лепестка, а также название вида.

Данные разбиты на столбцы по 4 параметрам, описанным выше, и выглядят как на рисунке 2.

```
[5.1 3.5 1.4 0.2]
[4.9 3.  1.4 0.2]
[4.7 3.2 1.3 0.2]
```

Рис. 2 Представление данных

Следующий код позволяет отобразить эти данные в виде графика.

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn import datasets


iris = datasets.load iris()


X = iris.data[:, :2]
y = iris.target
h = .02

x_min, x_max = X[:, 0].min() - 1, X[:, 0].max() + 1
y_min, y_max = X[:, 1].min() - 1, X[:, 1].max() + 1
xx, yy = np.meshgrid(np.arange(x_min, x_max, h), np.arange(y_min, y_max,
h))

plt.figure()
plt.scatter(X[:, 0], X[:, 1])
plt.xlim(xx.min(), xx.max())
plt.ylim(yy.min(), yy.max())
plt.title("Точки")
plt.show()
```

Результатом будет рисунок 3.

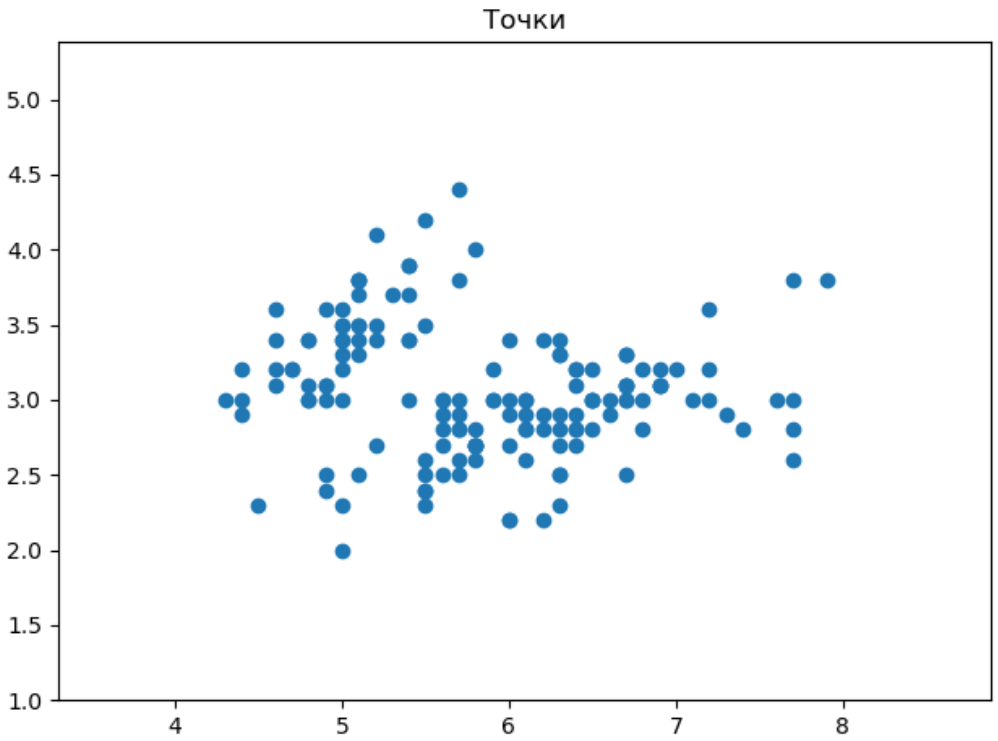

Рис. 3 Графическое представление данных про ирисы

На рисунке 3 уже хорошо видно скопление точек в некоторых местах, это и есть разные сорта ириса, только это отображение одного из 4 параметров цветов.

Использование метода n-ближайших соседей заключается в импортировании класса KNeighborsClassifier из библиотеки sklearn и «скармливание» данных методу fit его экземпляра.

Код с применением этого метода представлен ниже.

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.colors import ListedColormap
from sklearn import neighbors, datasets

n_neighbors = 6

iris = datasets.load_iris()

X = iris.data[:, :2]
y = iris.target
h = .02

cmap_light = ListedColormap(['#FFAAAA', '#AAFFAA','#00AAFF'])
cmap_bold = ListedColormap(['#FF0000', '#00FF00','#00AAFF'])

clf = neighbors.KNeighborsClassifier(n_neighbors, weights='distance')
clf.fit(X, y)

x_min, x_max = X[:, 0].min() - 1, X[:, 0].max() + 1
y_min, y_max = X[:, 1].min() - 1, X[:, 1].max() + 1
xx, yy = np.meshgrid(np.arange(x_min, x_max, h),
np.arange(y_min, y_max, h))
```

```
Z = clf.predict(np.c_[xx.ravel(), yy.ravel()])

Z = Z.reshape(xx.shape)
plt.figure()
plt.pcolormesh(xx, yy, Z, cmap=cmap_light)

plt.scatter(X[:, 0], X[:, 1], c=y, cmap=cmap_bold)
plt.xlim(xx.min(), xx.max())
plt.ylim(yy.min(), yy.max())
plt.title("3-Класса разбиения при (k = %i)" % (n_neighbors))
plt.show()
```

Результат представлен на рисунке 4.

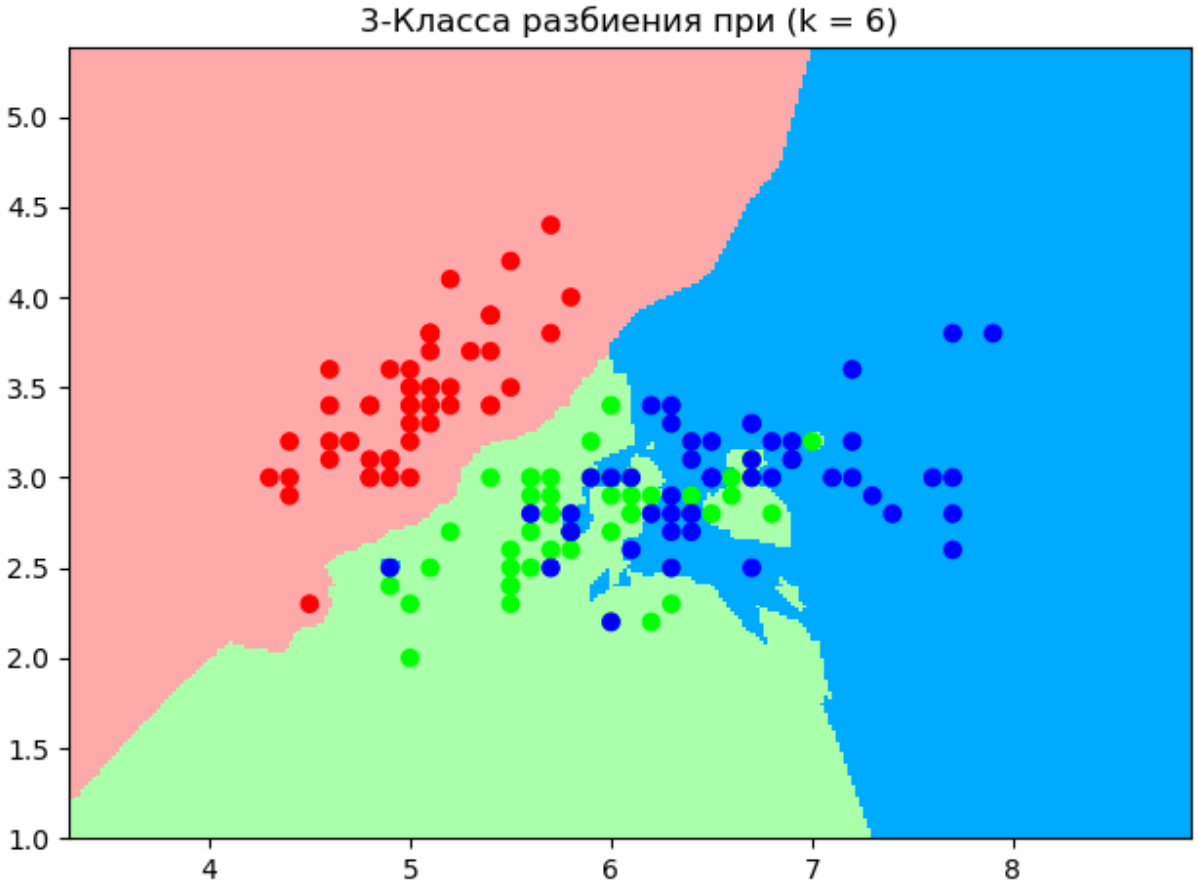

Рис. 4 Разбиение на полигоны сортов

Теперь, можно использовать эту систему для прогнозирования. Для этого нужно дать машине новые данные и на основе этой классификации она вернет сорт цветка.

Следующий код сначала тренирует машину на учебных данных, а после спрашивает у пользователя длину и ширину чашелистика и выдает прогноз.

```
from sklearn import neighbors, datasets

n_neighbors = 6

iris = datasets.load_iris()
```

```
X = iris.data[:, :2]
y = iris.target
h = .02

clf = neighbors.KNeighborsClassifier(n_neighbors, weights='distance')
clf.fit(X, y)

sl = float(input('Введите длину чашелистика (cm): '))
sw = float(input('Введите ширину чашелистика (cm): '))
dataClass = clf.predict([[sl,sw]])
print('Прогнозирование: '),

if dataClass == 0:
    print('Iris Setosa')
elif dataClass == 1:
    print('Iris Versicolour')
else:
    print('Iris Virginica')
```

Результатом выполнения кода и заполнения представлен на рисунке 5.

```
Введите длину чашелистика (cm): 6.5
Введите ширину чашелистика (cm): 2
Прогнозирование:
Iris Versicolour
```

Рис. 5 Вывод прогнозирования

Вывод

Сфера применения машинного обучения очень обширна, и она не заканчивается на анализе цветков, любые данные, представляющие собой статистику можно также пропускать через машинное обучение для нахождения зависимостей которые раньше были не очевидны.

Библиографический список

1. Артамошкин М.С. Принятие решений в электронной коммерции с использованием библиотеки scikit-learn // В сборнике: XLVI Огарёвские чтения Материалы научной конференции: в 3-х частях. Ответственный за выпуск П.В. Сенин. 2018. С. 201-208.
2. Третьякова А.А. Классификация животных методом машинного обучения k-ближайших соседей // Территория инноваций. 2017. № 12 (16). С. 42-47.
3. Коротеев М.В. Обзор некоторых современных тенденций в технологии машинного обучения // E-Management. 2018. № 1. С. 26-35.
4. Ярыгин А.А. Актуальные вопросы машинного обучения с подкреплением интеллектуальных агентов в задачах принятия решений // В сборнике: Автоматизация: проблемы, идеи, решения сборник статей по итогам Международной научно-практической конференции. 2017. С. 62.
5. Сидоров К.С., Ахунжанов Р.К. Разработка и внедрение методических материалов к курсу по машинному обучению в астраханском государственном университете // Международный научно-исследовательский журнал. 2017. № 9-1 (63). С. 155-158.

6. Шмид А.В., Лычагин К.А. Машинное обучение в экспертных системах: подготовка специалистов // Образовательные ресурсы и технологии. 2014. № 2 (5). С. 102-106.
7. Chang K.Ch., Hong T.P., Tseng Sh.Sh. Machine learning by imitating human learning // Minds and Machines. 1996. Т. 6. № 2. С. 203-228.
8. Sklearn библиотека машинного обучения https://scikit-learn.org/stable/index.html (Дата обращения: 22.05.2019)
9. Набор данных про ирисы. http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data (Дата обращения: 22.05.2019)