

Алгоритм классификации к ближайших соседей

Задача на практику



Продолжая изучать материалы, привезенные ассистентом из экспедиции в Африку, профессор Буковски обнаружил результаты похожих исследований японского ученого Какая Икота. Профессор Икота, изучая обезьян, выделил 4 класса: Лемуры, Шимпанзе, Гориллы, Орангутанги. Каждый вид характеризуется определенным средним значением роста и веса. Экспериментальные данные профессора Икота представлены в виде таблицы:

№	Рост	Вес	Вид	№	Рост	Вес	Вид
1	33	21	Лемур	9	185	155	Горилла
2	41	13	Лемур	10	193	129	Горилла
3	18	22	Лемур	11	164	135	Горилла
4	38	34	Лемур	12	205	131	Горилла
5	62	118	Шимпанзе	13	145	55	Орангутан
6	59	137	Шимпанзе	14	168	35	Орангутан
7	95	131	Шимпанзе	15	135	47	Орангутан
8	83	110	Шимпанзе	16	138	66	Орангутан



Профессору Буковски требуется определить, к какому виду относятся обезьяны, изученные его ассистентом во время экспедиции. Для решения этой задачи Буковски собирается использовать алгоритм к ближайших соседей.

Алгоритм к ближайших соседей

Пусть у нас имеется матрица объектов **X**. Каждая строка таблицы описывает отдельный объект. Каждый столбец таблицы описывает определенный признак объекта (рост, вес и т.п.). Кроме того, в матрице есть еще один столбец, описывающий класс, к которому принадлежит объект (см. рисунок ниже).

	Признаки: рост, вес и т.п.		Класс
	→	→	→
Объекты ↓	33	21	1
	59	137	2
	164	135	3
	168	35	4

1 – лемуры
2 – шимпанзе
3 – гориллы
4 – орангутанги

Пусть имеется объект **obj**, обладающий набором признаков, который необходимо отнести к заданному классу. Например, в нашем случае, **obj** будет представлять собой отдельную особь с признаками «рост» и «вес».

Чтобы определить, к какому классу принадлежит объект, нужно найти **k** ближайших к нему соседей (точек) и посмотреть, какой класс у этих соседей встречается чаще всего. Этому классу и будет принадлежать наш классифицируемый объект.

1. Измеряем расстояние от объекта **obj** до каждого объекта в матрице **X**.
2. Сортируем полученные значения в порядке увеличения расстояния (в начале будут объекты наиболее близкие к **obj**, в конце – наиболее удаленные).
3. Берем первые **k** объектов из отсортированного списка, определяем, какой класс встречается там чаще всего. Именно этому классу и принадлежит наш объект **obj**.

Порядок решения задачи

Открыть для редактирования файл **run.py**. Заполнить матрицу **X** экспериментальными данными, собранными профессором Икота. Следует обратить внимание, что последним столбцом в этой матрице будет столбец с номером класса (от 1 до 4), к которому принадлежит данная особь.

Пользуясь функцией **input**, попросить пользователя ввести рост и вес особи, которую необходимо классифицировать. Результаты пользовательского ввода сохранить в переменных **height** и **weight** соответственно.

Открыть для редактирования файл **kNN.py**. Необходимо полностью реализовать функцию **k_nearest**, основываясь на комментариях **TODO**, приведенных в файле. Перед реализацией ознакомиться с алгоритмом классификации **k** ближайших соседей. В комментариях приведены функции, которые могут потребоваться для решения задачи. Следует использовать команду **help(<функция>)** для получения справки по функции и использовать командное окно для экспериментов.

Взять результаты, собранные ассистентом профессора Буковски (см. предыдущую практику), и выполнить классификацию **не менее 5** особей с помощью реализованного алгоритма классификации. Убедиться в достоверности результатов. Продемонстрировать.