Метод K-ближайших соседей для решения задачи классификации

Метод K-ближайшего соседа (англ.: *k-nearest neighbors method*, *k-NN*) – один из методов решения задачи классификации.

Предполагается, что уже имеется какое-то количество объектов с точной классификацией (т.е. для каждого них точно известно, какому классу он принадлежит). Нужно выработать *правило, позволяющее отнести новый объект к одному из возможных классов* (т.е. сами классы известны заранее).

В основе k-NN лежит следующее правило: *объект считается принадлежащим тому классу, к которому относится большинство его ближайших соседей.* Под «соседями» здесь понимаются объекты, близкие к исследуемому в том или ином смысле.

Заметим, что здесь необходимо уметь определять, насколько объекты близки друг к другу, т.е. уметь измерять «расстояние» между объектами. Это не обязательно евклидово расстояние. Это может быть мера близости объектов, например, по цвету, форме, вкусу, запаху, интересам (если речь идёт о формировании групп людей), особенностям поведения и т.д. Следовательно, для применения метода k-NN в пространстве признаков объектов должна быть введена некоторая *метрика* (т.е. *функция расстояния*).

Предполагается, что объекты с близкими значениями одних признаков будут близки и по другим признакам (т.е. относиться к одному и тому же классу).

Рассмотрим работу метода k-NN на простом примере [1, стр. 67].

Продукт	**Сладость**	**Хруст**	**Класс**
яблоко	9	8	фрукт
бекон	1	4	протеин
банан	10	1	фрукт
…	…	…	…

Здесь качества продуктов (сладость и хруст) оцениваются по 10-балльной шкале. Эти значения можно рассматривать как координаты точек (продуктов) в 2-мерном пространстве. По оси будем откладывать степень сладости продукта, по оси ординат – степень хруста. Получим график, изображённый на Рис.3.

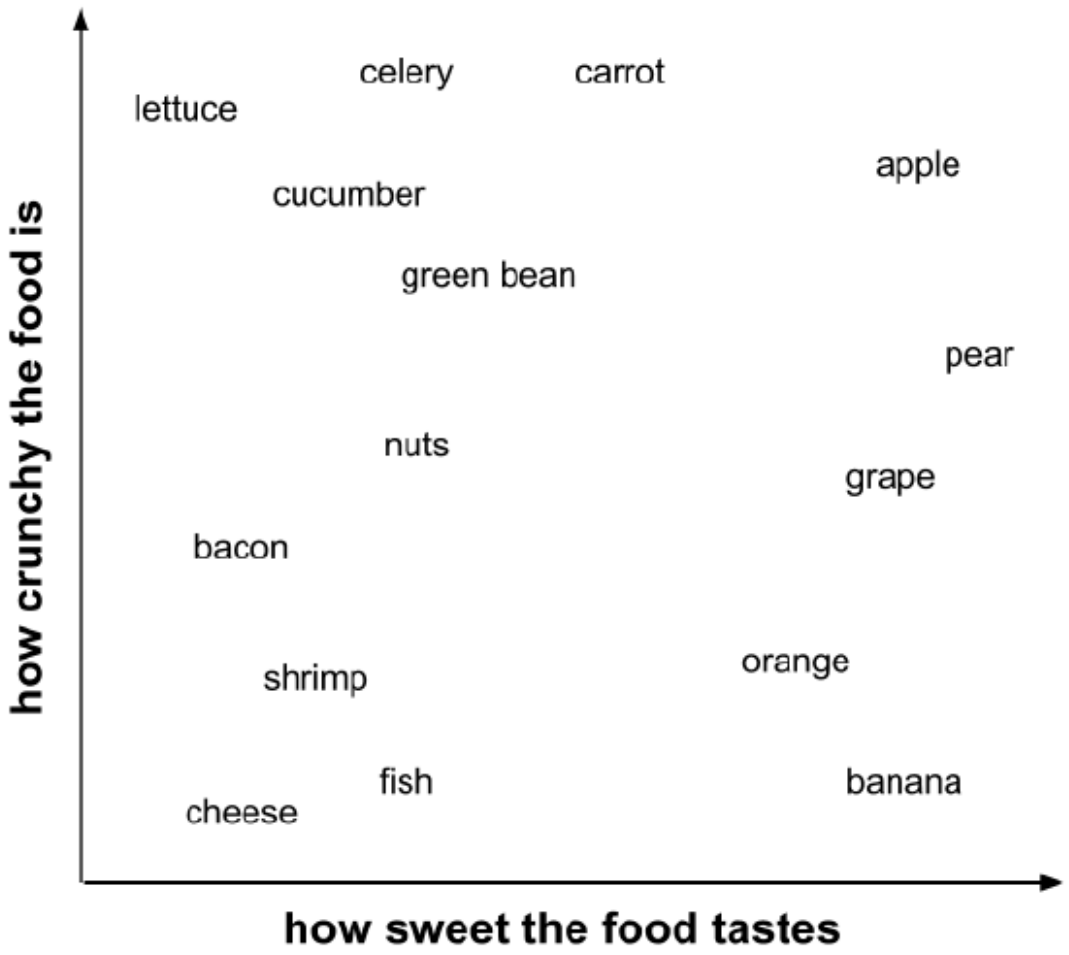

Рис.1 Результат визуализации данных

Для каждого из названных продуктов мы точно знаем тип (класс) – см. последний столбец таблицы. Можно заметить, что точки (продукты) на графике можно разбить на классы:

– в левом верхнем углу «группируются» овощи (огурец, морковь, салат-латук, сельдерей) – они хрустящие и несладкие,

– в левом нижнем – продукты, богатые протеином (бекон, креветки, сыр, рыба, орехи) – они нехрустящие и несладкие,

– справа «выстроились» фрукты (яблоко, груша, виноград, апельсин, банан) – они сладкие по сравнению с другими классами, но неоднородны в отношении хруста.

Разбиение на классы показано на Рис.2.

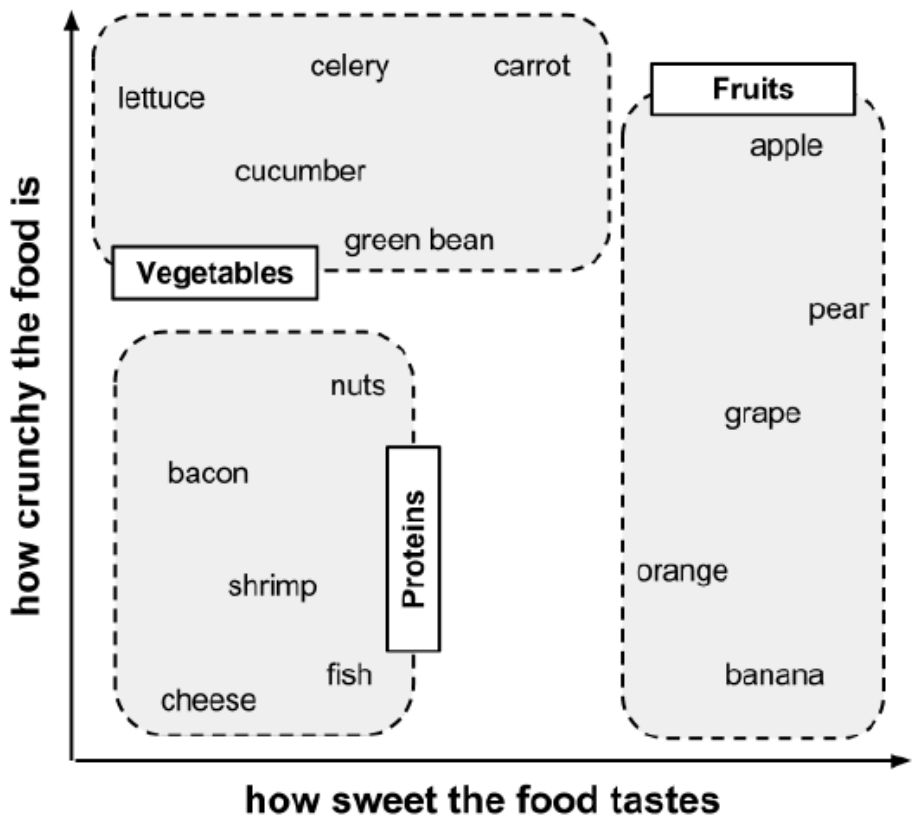

Рис.2. Продукты очевидно образуют 3 класса

Предположим теперь, что нам предложен новый продукт, и мы должны определить, к какому классу он относится – см. Рис. 3.

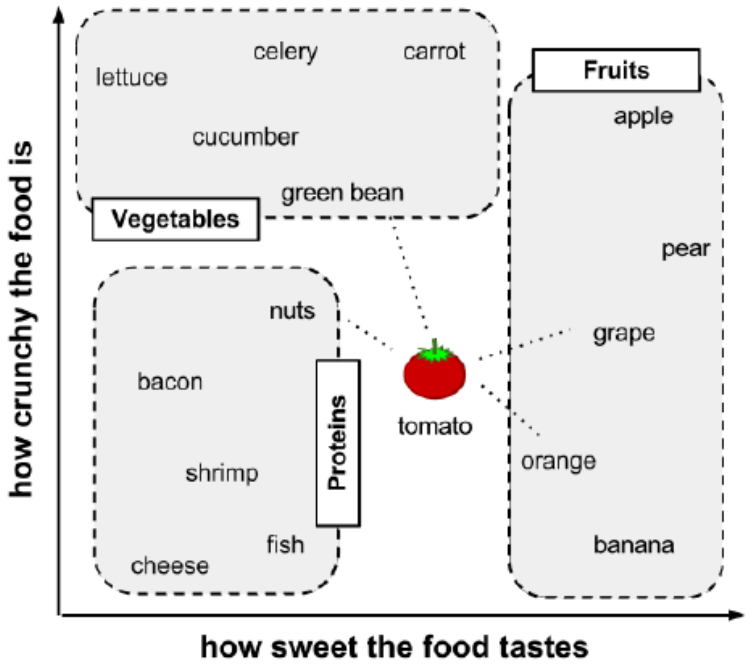

Рис.3. Помидор – это овощ или фрукт?.

Согласно методу k-NN мы отнесём его к тому классу, к которому принадлежит большинство из k его ближайших соседей. Расстояние между объектами будем понимать в смысле *Евклидовой нормы*, т.е. расстояние между объектами с координатами (x_1, y_1) и (x_2, y_2) равно $\sqrt{((x_1 - y_1)^2 + (x_2 - y_2)^2}$.

Так, если у томата показатель сладости равен 3, а показатель хруста равен 7, то его расстояния до яблока, бекона и банана равны примерно 6,1; 3,6, 9,2, соответственно. Приведём расстояния от томата до всех остальных продуктов, упорядочив их по возрастанию:

№	**Продукт**	**Класс**	**Расстояние до томата**
1	апельсин	фрукт	1,4
2	виноград	фрукт	2,2
3	креветка	протеин	3,5
4	бекон	протеин	3,6
5	орехи	протеин	3,6
6	сыр	протеин	4,0
7	бобы	протеин	4,2
8	огурец	овощ	5,9
9	яблоко	фрукт	6,1
10	морковь	овощ	6,8
11	сельдерей	овощ	7,0
12	салат-латук	овощ	7,2
13	банан	фрукт	9,2

Теперь нужно выбрать число k и определить, к какому классу принадлежат большинство из k ближайших соседей томата. Так, если k=1, то ближайший сосед – один, и это апельсин, он – фрукт. При k=2 это апельсин и виноград, оба фрукты. При k=3 мы имеем 2 фрукта (апельсин и виноград) и креветку (протеин). Значит, метод k-NN опять даст ответ: «фрукт». При k=4 ответом сбудет «фрукт или протеин с равной вероятностью». При k=5, k=6, k=7, k=8 «побеждает» протеин. Этот процесс можно продолжать и далее, увеличивая значение k. Мы видим, что *результат, получаемый методом k-NN, сильно зависит от выбора параметра k.*

Возникает вопрос: *как выбрать значение параметра k, чтобы минимизировать количество неверных ответов, полученных методом k-NN?*

Если мы выберем значение *k* слишком малым, то есть опасность, что единственным ближайшим объектом окажется «выброс», т.е. объект с неправильно определённым классом, и он даст неверное решение. Казалось бы, увеличивая значение параметра *k*, мы снижаем вероятность случайного попадания на такие «выбросы» в качестве ближайших соседей исследуемого объекта. Но здесь возникает другая опасность. Чтобы понять в чём она заключается, рассмотрим случай, когда *k* равно общему числу объектов *N*. Понятно, что тогда «победит» самый популярный (модальный) класс, и расстояние до исследуемого объекта не будет играть вообще никакой роли. Проблему выбора оптимального значения параметра k называют «*bias-variance tradeoff*», т.е. «компромисс между «выбросами» и дисперсией». На практике чаще всего полагают $k=[\sqrt{N}]$. Т.е. в нашем примере *k*=3 и результатом классификации будет то, что *помидор – фрукт*.

В том случае, если мы уверены в «чистоте» выборки, мы можем выбирать *k* меньшим. Существует также приём под названием «*weighted voting*» (т.е. буквально «взвешенное голосование»), при котором более близкие соседи исследуемого объекта имеют больший вес, чем более дальние.

Рассмотрим ещё один аспект применения метода k-NN – предварительную подготовку данных.

Подготовка данных для применения метода k-NN

Заметим, что в рассмотренном примере оба признака (уровень сладости и хруста продуктов) измерялись в одной шкале – принимали значения от 0 до 10. На практике различные признаки могут иметь разные единицы измерения и разные шкалы, что может существенно искажать реальное расстояние между объектами. Для решения этой проблемы перед применением метода k-NN производят так называемую *нормализацию* (или *масштабирование*) данных (англ.: *scaling*).

Существуют различные способы нормализации. Приведём некоторые наиболее часто используемые:

$$x_i \equiv \frac{x_i - x_{min}}{x_{max} - x_{min}}. \quad (1)$$

Формула (1) означает переход от абсолютных значений признаков к относительным. Преимущество новых переменных состоит в том, что они принимают значения от 0 до 1 (или, если перейти к процентному выражению, то о 0 до 100).

Второй способ масштабирования имеет вид:

$$x_i \equiv \frac{x_i - \bar{x}}{s}, \quad (2)$$

где $\bar{x}$ – выборочное среднее (т.е. $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$), s – выборочное среднеквадратическое отклонение (т.е. s =.$\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2}$.

Как известно, если с.в. ξ имеет нормальное распределение с параметрами μ и σ, то с.в. $\eta \equiv \frac{\xi-\mu}{\sigma}$ также является нормально распределённой, но параметры её распределения равны 0 и 1, соответственно (такие с.в. называются *стандартными* гауссовыми).

Не все признаки имеют количественное выражение. В этом случае прибегают к так называемому *dummy coding*. Например, значение признака «пол» можно обозначить 1 для мужчин и 0 для женщин.

Литература

1. Brett Lantz. Machine Learning with R. Pack Publishing. Birmongham-Mumbai, 2013.