Алгоритм максиминного расстояния. Алгоритм сокращения эталонов в задачах классификации

- Выбор первого эталона (центра)
- Выбор второго эталона (центра)
- Условие определения последующих эталонов
- Параметр алгоритма
- Выбор начальных эталонов
- Отмеченная строка и отмеченный столбец
- Матрица расстояний
- Функционал качества

Алгоритм, основанный на принципе максимально-минимального расстояния, представляет собой одну простую эвристическую процедуру, использующую евклидово расстояние. Алгоритм предназначен для решения задачи кластерного анализа.

Пусть $S = S_1, S_2, \ldots, S_m$, где $S_i = {}_{i1}, {}_{i2}, \ldots, {}_{in}$.

Шаг1. Один из объектов $S_i \, S$ произвольным образом назначается центром первого класса Z_1. Например: $Z_1 = S_1$.

Шаг 2. Отыскивается объект, отстоящий от объекта S_1 на наибольшее расстояние

$$\rho(S_1, S_t) = \max_{S_i \in S} \rho(S_1, S_i), \text{ где } S_t \, S.$$

Найденный объект S_t назначается центром второго кластера $Z_2 = S_t$.

Шаг 3. Вычисляются расстояния от каждого объекта $S_i \, S \setminus \{S_1, S_t\}$ до центров Z_1, Z_2. В каждой паре этих расстояний выделяется минимальное. Число таких минимальных расстояний будет *m-2*. После этого выделяется максимальное из этих минимальных расстояний

$$_t = \max_{S_i \in S | S_1, S_t|} \min \; (S_i, Z_1), \; (S_i, Z_2) \; . \qquad (2.26)$$

Шаг 4. Проверяется условие назначения следующего центра, т.е. если значение $_t$ составляет значительную часть расстояния между центрами кластеров Z_1 и Z_2 (скажем, по меньшей мере, половину этого расстояния), то соответствующий значению $_t$, объект S_t назначается центром кластера Z_3.

В противном случае выполнение алгоритма прекращается.

Шаг 5. Здесь вычисляется расстояние между тремя выделенными центрами кластеров и всеми остальными выборочными образами; в каждой группе из трех расстояний выбирается минимальное. После того, как и на предыдущем шаге, находится максимальное из этих минимальных расстояний. Если последнее составляет значительную часть “ типичных” предыдущих максимальных расстояний, то соответствующий объект назначается центром Z_4 следующего кластера. В противном случае выполнение алгоритма прекращается.

Таким образом, представлен алгоритм нахождения центров исходного множества *S*.

Вторым этапом алгоритма может служить любой алгоритм распознавания. Эталонами кластеров являются найденные центры.

Пример. Возьмем множество $S = S_1, S_2, \ldots, S_{10}$, где $S_1=(0, 0)$, $S_2=(3, 8)$, $S_3=(2, 2)$, $S_4=(1, 1)$, $S_5=(5, 3)$, $S_6=(4, 8)$,

$S_7=(6, 3)$, $S_8=(5, 4)$, $S_9=(6, 4)$, $S_{10}=(7, 5)$.

Используя алгоритм, определяем количество кластеров на основе выделения центров кластеров.

На первом шаге объект S_1 назначается как центр Z_1 первого кластера.

Шаг 2. Отыскивается наиболее удаленный объект от Z_1, в данном случае это объект S_6, который назначается как центр второго кластера Z_2.

Шаг 3. Вычисляем расстояния от каждого объекта множества $S_2, S_3, S_4, S_5, S_7, S_8, S_9, S_{10}$ до центров S_1 и S_6. В каждой паре выделяем минимальные расстояния. Затем находим максимальное среди этих восьми минимальных расстояний.

Шаг 4. Проверяем условия назначения следующего центра. Максимальное расстояние $_7$,

$$\rho_t \geq \frac{\rho(S_1, S_6)}{2}.$$

Следовательно, объект S_7 назначается центром Z_3 третьего кластера. На следующем шаге алгоритма найденное максимальное значение не удовлетворяет условию шага 4, и тем самым работа алгоритма прекращается.

В этом простом примере были выделены три кластерных центра S_1, S_6 и S_7. Решая задачу распознавания, получаем три кластера:

$K_1 = S_1, S_2, S_4$; $K_2 = S_2, S_6$; $K_3 = S_5, S_9, S_{10}$.

Современный этап развития распознавания образов и классификации характеризуется широким распространением групповых (коллективных, комитетных) методов, предусматривающих оптимальный синтез алгоритмов из составляющих базовый набор алгоритмов. С другой стороны, часто при решении прикладных задач необходимо иметь нужное число кластеров.

Далее рассмотрим эффективный алгоритм сокращения эталонных объектов. Быстрая сходимость алгоритма, основанная на эффективных вычислительных схемах с использованием функционала качества в процессе вычисления дает надежные результаты при решении прикладных задач.

При использовании различных алгоритмов классификации часто появляется больше эталонных объектов и соответственно кластеров, чем требуется в данной ситуации, поэтому анализ и методы сокращения несущественных кластеров являются необходимой и важной задачей. Приведем ниже и точное решение задачи разбиения на два класса.

Пусть $S = S_1, S_2, S_3, \ldots, S_m$ - допустимое множество объектов, каждый из которых описывается n параметрами, называемыми признаками, т.е.

$S_i = (a_{i1}, a_{i2}, \ldots, a_{in})$, a_{ij} - признаки, $i=1,\ldots, m$, $j=1,\ldots, n$.

Будем считать, что для каждой пары объектов S_i, S_j S определены понятие близости (мера близости).

В качестве меры близости объектов друг к другу, может служить обычное евклидово расстояние.

Пусть признаки объектов S_i, S_j однородны по своему физическому смыслу, и установлено, что все они важны с точки зрения решения вопроса об отнесении объекта к тому или иному классу.

Могут применяться и другие меры близости, такие как метрика Махаланобисского типа, Хеммингова метрика, взвешенное евклидово расстояние и меры близости типа вычисления оценок.

Исходному множеству S на основе выбранной метрики сопоставим симметрическую матрицу $C = C_{ij\ m\,m}$, каждый элемент C_{ij} которой является величиной близости объектов S_i,

S_j S $(i, j=1,2,\ldots, m)$.

Пусть C_{ij} тем меньше, чем ближе в рассматриваемом смысле S_i, S_j, причем $C_{ii}=0$, C_{ij} 0 i j.

Выбираются l объектов из S в качестве эталонных объектов. Выбор может быть произвольным, произведен экспертом, или сделан на основе предварительной обработки исходной информации S. Множество индексов выбранных эталонных объектов обозначим через P. Проводим процесс распознавания остальных объектов из множества S по эталонным объектам. Объект S_j $(j=1,\ldots, m)$ объединяется в класс с эталонным объектом S_i, если C_{ij} -

минимальный (или один из наименьших) из всех элементов C_{vj} столбца j, т.е., $C_{ij}=\min_{v} C_{vj}$.

Допускаемое разбиение характеризуется суммой m таких минимальных элементов матрицы C, такому разбиению соответствует функционал вида

$$F=\sum\sum ¿\, i \in Pj \in p_{\sigma} C_{ij}, \qquad (2.27)$$

где P_{σ}- множество индексов объектов, входящих в подмножество с эталонным объектом S_i.

Заметим, что в F входят l - нулевых элементов $C_{ii}=0$ $(i\ P)$ и $m-l$ определенных элементов C_{ij} 0 $(i\ P,\ j\ i)$. Элементы матрицы, входящие в F, и ее строки $i\ P$ будем называть отмеченными. При $l=n$ отмечены все строки и диагональные нулевые элементы $C_{ii}(i=1,\ldots, m)$.

Следует отметить, что с уменьшением значения F качество разбиения на l классов повышается, т.е. происходит объединение объектов с малыми взаимными C_{ij} и распределение по разным подмножествам объектов с большими значениями C_{ij}. На основе этого наилучшим будем считать разбиение множества S на l классов, минимизирующее значения функционала F: $F_{\min}=\min_{P,P_i} F$.

Перейдем теперь к нашей основной задаче. Пусть выбраны l эталонных объектов $(l\ \ l)$ и такому разбиению соответствует функционал $F(l)$. Требуется построить алгоритм сокращения числа эталонных объектов l к требуемому числу l. Выбор того или иного сокращаемого эталона, а значит и соответствующего кластера производится на основе анализа функционала качества F.

1. Перейдем к $l-1$ эталонным объектам

Рассмотрим столбцы, в которых находятся отмеченные элементы первой из отмеченных строк, т.е. строка, где расположен первый эталонный объект. Отыщем в каждом таком столбце среди элементов, находящихся в отмеченных строках, за исключением рассматриваемой, наименьший элемент (любой из нескольких наименьших). Найдем разность между суммами этих элементов и отмеченных элементов рассматриваемой строки.

2. Найдем такие разности для каждой из отмеченных строк матрицы C и выберем строку, соответствующую наименьшей разности $F(l)$. Сократим эту строку и ее элементы из числа отмеченных и отметим заново элементы, составляющие уменьшаемый в $F(l)$ объект, с номером исключенной строки выводимый из числа эталонных.

Если исключим первую отмеченную строку из числа эталонных, раскидав отмеченные элементы этой строки на основе поиска наименьшего значения по остальным отмеченным строкам, то функционал F ухудшится на значение $F(l)$.

Таким образом, получили $l-1$ эталонных объектов, которым соответствует минимальное для этого набора объектов значение функционала качества

$$F(l-1)=F(l)+\ F(l). \qquad (2.28)$$

Повторяя пункты 1), 2) последовательно, получим итоговое разбиение m объектов из S на l непустых классов, другими словами, построим алгоритм A такой, что

$$A(S)\stackrel{l}{=}_{j=1} K_j, K_i \cap K_j=\varnothing, i\neq j, K_i, K_j\neq\varnothing, \qquad (2.29)$$

где l - число классов, $i, j=1, 2,\ldots, l$.

Следует отметить, что значение F возрастает с каждым шагом. Можно показать, что на каждом шаге выбирается наилучший в смысле минимума F набор эталонных объектов из предыдущего набора. Этим определяется пошаговая оптимизация получаемого решения.

Предложенный алгоритм допускает исследование выбора решающих правил на каждом этапе.

В качестве начального набора эталонных объектов можно принять сами объекты $S_i\ S$, $(i=1,\ldots, m)$, при этом $F(n)=0$ и каждый класс –эталон. Далее можно подвергать анализу каждый эталон – объект до получения требуемого числа кластеров.

В качестве первоначальных эталонов можно использовать, например, результаты максимального алгоритма, приняв его как алгоритм предварительной обработки. В этом случае следует ожидать, что сходимость и качество разбиения улучшатся.

Получив l - классов, целесообразно отыскать в каждом из них эталонные объекты, которые могут оказаться новыми, и по найденному набору эталонных объектов скорректировать разбиение. Такая корректировка возможна и на промежуточных шагах выполнения алгоритма.

В вычислительном отношении алгоритм достаточно прост и требует отыскания минимальных элементов некоторых строк и столбцов исходной матрицы без ее преобразования.

Для случая, когда требуется получить разбиение исходного множества S на два класса, для каждой пары строк матрицы C отыщем сумму меньших из двух элементов каждого столбца. Выберем минимальную из таких сумм. Элементы, входящие в нее, определяют искомое разбиение.

Для ориентировочной проверки результатов решения при l 2, получаемого алгоритмом, целесообразно повторить пункты 1), 2) предыдущего алгоритма до получения разбиения на два класса $(K=2)$ и сравнить с точным решением разбиения на два класса.

В ситуациях, когда исследователю заранее неизвестно, на какое число классов подразделяется исходное множество S, функционалы качества разбиения $F(S)$ выбирают чаще всего в виде простой алгебраической комбинации (суммы, разности, произведения, отношения) двух функционалов $F_1(S)$ и $F_2(S)$, один из которых является убывающей функцией от числа классов и характеризует, как правило, внутриклассовый разброс объектов, а второй $F_2(S)$ – возрастающей функцией числа классов. При этом интерпретация $F_2(S)$ может быть различной. Под $F_2(S)$ понимается иногда некоторая мера взаимной удаленности (близости) классов.

С указанных позиций приводится обобщенный функционал качества разбиения.

В качестве критерия, определяющего компактность образуемых классов (внутриклассовый разброс), понимается величина, характеризующая плотность расположения объектов внутри каждого класса.

$$F_1=\frac{1}{l}\sum_{j=1}^{l} H_j(K), \text{ где} \qquad (2.30)$$

$$H_j(K)=\frac{1}{N_j}\sum_{K=1}^{m_j} \square \sum_{i=K+1}^{m_j} \rho(S_k S_i), S_i, S_k \in K_j; \quad (2.31)$$

l - число классов;

m_j - число объектов класса K_j ;

$N_j=\frac{m_j(m_j-1)}{2} N_j$ - количество расстояний между объектами в K_j ;

;

(S_k, S_i) – евклидово расстояние

В качестве критерия разделимости классов F_2 используется величина

$$F_2=\frac{1}{N^0}\sum_{j=1}^{l} \square \sum_{i=j+1}^{l} \rho(S(K_j), S(K_i)), \qquad (2.32)$$

где $S(K_j)=\frac{1}{m_j}\sum_{i=1}^{m_i} S_i$ – обобщенный эталон класса K_j ;

N^0 - число расстояний между обобщенными эталонами.

Таким образом, обобщенный функционал качества зависит от параметров F_1, F_2, т.е. $F = F(F_1, F_2)$.