УДК 004.9

ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ РЕШЕНИЯ ЗАДАЧИ КЛАСТЕРИЗАЦИИ

Богодухов Д. М.,
научный руководитель канд. физ.-мат. наук, доцент Баранова И. В.
Сибирский Федеральный Университет,
Институт математики и фундаментальной информатики

Введение

Целью работы является исследование генетических алгоритмов и способов их применения для решения задач кластеризации.

В работе предлагается генетический алгоритм, позволяющий решать задачу кластеризации данных. Также было разработано программное приложение, реализующее работу метода k-средних и генетического алгоритма. Произведена оценка эффективности и работоспособности этой программы. В работе решается практический пример задачи кластеризации данных. Проведено сравнение кластеров, полученных генетическим алгоритмом и методом k-средних.

Кластеризация предполагает деление набора точек данных на непересекающиеся группы, или кластеры, в пределах которых точки «больше похожи» друг на друга, чем на точки в других кластерах. Термин «больше похож», применительно к кластерным точкам, как правило, означает близость. Когда набор данных кластеризован, каждая точка включается в определенный кластер и может быть охарактеризована одной описательной точкой, как правило, средней точкой в кластере. Разбиением назовем любое конкретное разделение всех точек данных в кластеры.

Основная цель кластеризации данных – уменьшение размера и сложности данных. Сокращение данных осуществляется заменой координаты каждой точки в кластере координатами описательной точки кластера. Кластеризованные данные требуют значительно меньше ресурсов памяти и ими можно манипулировать гораздо быстрее, чем исходными данными. Значение того или иного метода кластеризации будет зависеть от того, насколько точно точки представляют собой данные, а также, насколько быстро работает программа.

Метод К-средних

Кластеризация методом К-средних — хорошо известный метод определения принадлежности элементов кластерам с помощью минимизации разницы между элементами кластера и максимизации расстояния между кластерами. Слово «средние» в названии метода относится к центроидам кластеров.

Центроид — точка данных, которая выбирается произвольно, а затем итеративно уточняется, пока не начинает представлять собой истинное среднее всех точек данных кластера.

Общая идея алгоритма: выбираются опорные точки и все исходные точки ассоциируются с кластерами. Алгоритм k-средних использует центроиды кластеров в качестве опорных точек для последующего разбиения, но центроиды корректируется во время и после каждого разбиения. Если центр тяжести Z_i для точки X в группе i является ближайшей описательной точкой то, никакие коррективы не вносятся, а алгоритм переходит к следующей точке. Однако, если центр тяжести Z_j кластера j ближе к точке X, то X переносится в кластер j, центроид «усеченного» кластера i (потерявшего точку X) и «выращенный» кластер j(получившего точку X)

пересчитывается, и опорные точки Z_i и Z_j перемещаются в новые центры тяжести. На каждом шаге каждая из k опорных точек является центороидом или средним, отсюда и название «k-средние».

Алгоритм *k-средних* имеет следующий вид:

1. Случайно выбрать k точек, являющихся начальными "центрами масс" кластеров (любые k из n объектов, или вообще k случайных точек).
2. Отнести каждый объект к кластеру с ближайшим "центром масс" согласно выбранной метрики. В качестве метрики обычно используют евклидову метрику.
3. Пересчитать "центры масс" кластеров согласно текущему членству.
4. Если критерий остановки алгоритма не удовлетворен, вернуться к шагу 2.

В качестве *критерия остановки* обычно выбирают один из двух:

- Отсутствие перехода объектов из кластера в кластер на шаге 2.
- Минимальное изменение среднеквадратической ошибки.

Алгоритм чувствителен к начальному выбору "центров масс".

Генетические алгоритмы

Генетические алгоритмы — это адаптивные методы поиска, которые в последнее время используются для решения задач оптимизации. В них используются как аналог механизма генетического наследования, так и аналог естественного отбора. Впервые понятие генетического алгоритма было предложено в 1975 г. в работах Джона Холланда. Генетические алгоритмы основаны на принципах естественного отбора Ч. Дарвина. Они относятся к стохастическим методам. Эти алгоритмы успешно применяются в различных областях деятельности (экономика, физика, технические науки и т.п.).

Основные операторы:

Оператор селекции – отбирает популяцию для дальнейшего размножения. Заключается в том, что родителями могут стать только те особи, значение приспособленности которых не меньше пороговой величины, например, среднего значения приспособленности по популяции. Такой подход обеспечивает более быструю сходимость алгоритма.

Оператор скрещивания – служит для распространения хороших генов по популяции.

Оператор мутации – изменяет значение гена в хромосоме на любое другое возможное значение. Случайное изменение генов должно осуществляться с низкой вероятностью, обычно в пределах [0.001;0.01]. После процесса воспроизводства происходят мутации *(mutation*). Данный оператор необходим для «выбивания» популяции из локального экстремума и препятствует преждевременной сходимости. Это достигается за счет того, что изменяется случайно выбранный ген в хромосоме.

Основные принципы работы генетических алгоритмов заключены в следующей схеме:

1) генерируется начальная популяция из n хромосом (решений задачи) – обычно в качестве начальной выбирается случайная популяция множества решений,
2) вычисляется оценка качества пригодности для каждого решения (хромосомы) – обычно она пропорциональна $1/e^2$),
3) выбирается пара хромосом-родителей с помощью одного из способов *отбора*,
4) проводится *скрещивание* двух родителей с вероятностью p_c, производя двух потомков, т.е. создается новое решение на основе рекомбинации из существующих;

5) проводится *мутация* потомков с вероятностью p_m – создается новое решение на основе случайного незначительного изменения одного из существующих,
6) повторяются шаги 3-5, пока не будет сгенерировано новое поколение популяции, содержащее n хромосом,
7) повторяются шаги 2-6, пока не будет достигнут критерий окончания процесса.

Применение генетических алгоритмов для решения задачи кластеризации

Главным достоинством генетических алгоритмов в данном применении является то, что они ищут глобальное оптимальное решение.

Большинство популярных алгоритмов оптимизации выбирают начальное решение, которое затем изменяется в ту или иную сторону. Таким образом, получается хорошее разбиение, но не всегда - самое оптимальное. Операторы рекомбинации и мутации позволяют получить решения, не похожие на исходные.

В работе решается практический пример простейшей задачи кластеризации данных – задача разбиения множества точек на k кластеров.

Имеются исходные данные, представленные в виде множества точек с координатами ***x*** и ***y***. Необходимо провести разбиения точек на k кластеров и вычислить центры кластеров. Данная задача выполняется с помощью двух приведенных ранее алгоритмов кластеризации данных.

Пусть случайным образом выбираются начальные центры кластеров. Для получения решения методом k-средних, вычисляется расстояние от текущей точки до начальных центров, и точка относится в кластер, с наименьшим расстоянием до центра. После того как все точки распределены по кластерам, пересчитываются центры кластеров, как среднее арифметическое всех координат. Таким образом, получаем новые центры, и алгоритм повторяется, пока не буден достигнут критерий остановки.

В случае с генетическим алгоритмом мы имеем дело с начальной популяцией из n хромосом, каждая из которых представляет вариант решения, то есть точки распределяются случайным образом по кластерам. Затем вычисляется коэффициент пригодности каждой хромосомы и в соответствии с ним создается новая популяция из наиболее приспособленных особей. Для каждой популяции пересчитываем центры кластеров, используя наилучшую хромосому из текущей популяции. Получаем новые центры и алгоритм продолжает работу пока не будет, достигнут критерий остановки.

Затем производится сравнение скорости работы данных алгоритмов и полученные решения.

Генетический алгоритм не гарантирует нахождение оптимального решения, однако показывает хорошие результаты за меньшее время по сравнению с другими алгоритмами кластеризации.