К ВОПРОСУ ОПТИМИЗАЦИИ ПАРАМЕТРОВ СГЛАЖИВАЮЩЕГО СПЛАЙНА

Е.Ю. Репина, Даньни У, Р.П. Затеев
Е. А. Кочегурова
Томский политехнический университет
eyr4@tpu.ru

Введение

Понятие "информация" стало одним из ключевых понятий нашего времени. При этом особую ценность представляет информация, полученная при работе с данными, поступающими в режиме реального времени (РРВ). После поступления текущие данные обрабатываются, и система выдает отклик еще до прибытия новой порции данных.

Обработанные данные используются в различных задачах интерпретации: фильтрации, прогнозирования, дифференцирования и др. При этом все данные содержат шум. Поэтому важную роль играет аппарат фильтрации и сглаживания данных, при помощи которого значительно повышается точность задачи интерпретации.

Описание задачи

Одной из форм представления цифровых данных являются временные ряды. Сглаживание временных рядов лежит в основе многих прикладных задач.

Применение сплайна позволяет получить искомое гладкое решение [1]. Сглаживающий сплайн S(t) основан на оптимизации специального вида функционала и представлен для РРВ следующей формулой:

$$J(S) = (1-\rho)(h\Delta t)^2 \int_{t_0^i}^{t_h^i} [S''(t)]^2 dt + \rho \sum_{j=0}^{h} [S(t_j^i) - y(t_j^i)]^2 \quad (1)$$

где ρ∈ [0, 1] — весовой коэффициент, устанавливающий баланс между сглаживающими и интерполяционными свойствами сплайна S(t);

Δt — интервал дискретизации наблюдаемого процесса;

h — количество измерений внутри i-го звена сплайна, далее h = const для всех звеньев сплайна на интервале наблюдения данных.

Все параметры сплайна оказывают определенное влияние на его свойства. Наиболее значимый вклад в точность аппроксимации вносит параметр h. Современные методы оптимизации, включая биоинсперированные и другие метаэвристики, позволяют получить только постоянное значение параметра h для всего интервала наблюдения. Однако это приводит к повышению погрешности в конечном результате.

Одним из возможных способов разбиения временного ряда на сегменты с переменным параметром числом измерений h является кластерный анализ.

Классификация алгоритмов кластеризации

Кластеризацией называют разбиение множества объектов на группы (кластеры).

Все алгоритмы кластеризации принято делить на иерархические и неиерархические (по данным, получаемым на выходе). Также, существует классификация алгоритмов кластеризации по принципам кластеризации: итеративные, плотностные, сетевые, модельные и концептуальные. Остановимся подробно на двух первых методах, как на самых распространенных.

Итеративные алгоритмы кластеризации – предполагают пошаговое перераспределение объектов между классами. Одним из популярных представителей данного семейства алгоритмов является алгоритм k-means. Основная идея данного алгоритма - пошаговая минимизация расстояний между объектами в кластерах до тех пор, пока это возможно. Главным недостатком алгоритма, применительно к **РРВ**, является необходимость заранее задавать желаемое количество кластеров, что недопустимо в РРВ.

Плотностные алгоритмы кластеризации - алгоритмы, определяющие кластер как группу объектов, расположенных кучно, т.е. так, чтобы в эпсилон-окрестности точки находилось минимально заданное число других объектов (соседей). Представителем этого класса алгоритмов кластеризации является алгоритмы DBScan, OPTICS.

Описание алгоритма DBSCAN

Алгоритм **DBScan (**Density-based spatial clustering of applications with noise) допускает кластеризацию пространственных данных в присутствии шума. Алгоритм был предложен в 1996 году М. Эстером и его коллегами для разбиения данных на кластеры произвольной формы. Для работы алгоритма используются два входных параметра: ε - окрестность, в которой будет требоваться наличие минимального количества объектов Minpts.

Было проведено исследование данного алгоритма в задаче оценки параметра h в условиях работы с данными с шумом, как и в случае реальных временных рядов в РРВ.

После реализации алгоритма DBScan в MATLAB, была проведена его апробация на модельных данных. В качестве тестовых примеров выбраны следующие функции:

$$f_1(t) = 10 \cdot \sin\left(\frac{2\pi \cdot i}{100}\right) \quad (2)$$

$$f_2(t) = \sin\left(\frac{\pi \cdot i}{20}\right) \cdot e^{0.02 \cdot t} + 3 \quad (3)$$

На данные, подготовленные для указанных функций, был наложен шум σ_ξ = 10% от разницы максимального и минимального значения функции.

Результаты кластеризации при варьировании параметров представлены в следующих таблицах 1 и 2.

Таблица 1. Результаты кластеризации для функции (2)

ε	Количество кластеров, шт	Процент выброса, %
(0.1 -2) σ_ξ	0	100
2.1σ_ξ	6	37,5
2.2σ_ξ	4	12,5
2.3σ_ξ	3	1
(2.4-2.5) σ_ξ	2-1	1

Таблица 2. Результаты кластеризации для функции (3)

ε	Количество кластеров, шт	Процент выброса, %
(0.1 -1.9) σ_ξ	0	100
2σ_ξ	9	40,5
2.1σ_ξ	6	24
2.2σ_ξ	3	17
2.3σ_ξ	3	8
(2.4 - 2.6) σ_ξ	2	6-5
2.7σ_ξ - …	1	1 - 2

Как видно из таблиц, в результатах присутствуют кластеры с неприемлемо большим числом данных внутри одного кластера и большое количество выбросов. Это значительно ухудшает качество кластеризации.

Анализ параметров алгоритма показал, что при малых окрестностях $\varepsilon \in [0.1\sigma_\xi, 1.9\sigma_\xi]$ все данные воспринимаются алгоритмом, как выбросы (шум). При увеличении данного параметра $\varepsilon > 2\sigma_\xi$ происходит уменьшение числа кластеров. Параметр Minpts, определяющий минимальное число соседей в группе, ограничен минимально возможным числом измерений для построения звена сплайна, и, следовательно, не может быть уменьшен. Увеличение этого параметра в задачах РРВ нелогично.

Результаты кластеризации при максимальном количестве кластеров представлены на следующих рисунках 1и 2.

В целом графические результаты показывают, что алгоритм DBScan разбивает временной ряд логически правильно для построения звеньев сплайна. Он неплохо выделяет фрагменты существенного изменения динамики, особенно для функции (2). Качество кластеризации можно было бы улучшить заданием верхней границы числа данных в кластере.

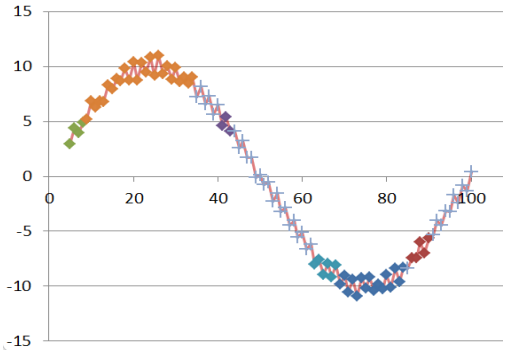

Рис. 1. Результат работы алгоритма DBScan для функции (2)

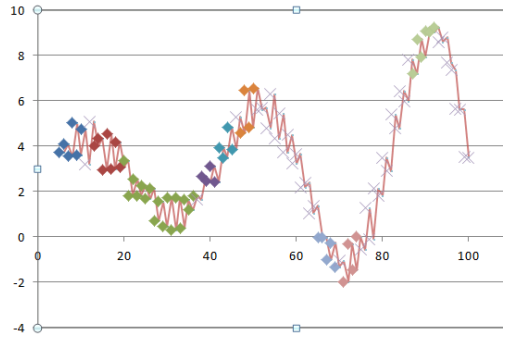

Рис. 2. Результат работы алгоритма DBScan для функции (3)

Заключение

На данном этапе исследований получено невысокое качество использования алгоритма DBScan в задаче кластеризации временных рядов. Далее предполагается модернизация данного алгоритма введением ограничения на размер кластера.

Список использованных источников

1. Кочегурова Е. А., Горохова Е. С. Текущее оценивание производной нестационарного процесса на основе рекуррентного сглаживающего сплайна// Автометрия. – 2016. - Т. 52.– № 3. – С. 79-85.
2. Плотностный алгоритм кластеризации пространственных данных с присутствием шума — DBSCAN. [Электронный ресурс] – URL: https://habr.com/post/143151/ / (дата обращения 12.10.2018).
3. Carmela Iorio, Gianluca Frasso, AntonioD'Ambrosio Parsimonious time series clustering using P-splines // Expert Systems with Applications №52, 15 June 2016, Pages 26-38.