УДК 004.8(075.8)

Библиотека визуализации результатов иерархической агломеративной кластеризации

С.А. Ермоченко[*1], Н.А. Ильина[*2]

[*1] к.ф.-м.н, доцент, Витебский государственный университет имени П.М. Машерова, yermochenkosa@vsu.by

[*2] магистрант, Витебский государственный университет имени П.М. Машерова, nadya_ilina_1998@mail.ru

Ермоченко С.А., Ильина Н.А. Библиотека визуализации результатов иерархической агломеративной кластеризации. *Рассматривается разработка библиотеки на языке программирования JavaScript, которая позволяет выполнять построение дендрограммы, представляющей собой графическое изображение иерархической структуры результатов кластеризации.*

Ключевые слова: *иерархическая агломеративная кластеризация, дендрограмма, язык программирования JavaScript, JSON.*

Введение

Методы кластеризации являются важным инструментом обработки больших объёмов данных, так как позволяют существенно сократить объём анализируемых данных за счёт выявления близких по характеристикам объектов предметной области и замены множества объектов одним обобщающим объектом.

Одним из самых часто используемых методов является метод k средних, основным достоинством которого является высокая скорость работы. Но данных метод удобно применять, когда исследователю хорошо известна специфика анализируемых данных. Данный метод предполагает определение количества кластеров заранее, перед выполнением процедуры кластеризации. Если обрабатывается большой массив данных, но количество кластеров выбрано специалистом не удачно, то для некоторых особенных объектов, характеристики которых выбиваются из общей совокупности, их специфика может быть исключена из рассмотрения из-за ошибочного включения этих объектов в кластеры, в которых большинство объектов такой спецификой не обладают [1].

При обработке данных о предметной области, особенности которой исследователю не известны или известны не в полной мере, когда заранее предположить количество кластеров является затруднительно, удобнее применять метод иерархической кластеризации, применяющих подход в постепенном формировании иерархической структуры вложенных друг в друга кластеров [2].

Агломеративная кластеризация строит иерархию кластеров начиная с мелких кластеров (изначально рассматриваются кластеры, в каждый из которых включается ровно один исходный объект), далее эти кластеры объединяются в более крупные кластеры до тех пор, пока все кластеры не будут объединены в один единый кластер. Самым сложным этапом является анализ результатов кластеризации. Когда количество объектов в иерархической структуре будет достаточно большим, а сама иерархическая структура становится сложной и разветвлённой, то выбор уровня, на котором можно кластер из некоторого количества объектов рассматривать как совокупность одинаковых объектов.

Для более удобного анализа выполняется построение дендограммы. Дендрограмма – это графическое изображение иерархической структуры кластеров, представляющей собой дерево, листовыми узлами которого являются исходные объекты, не листовыми узлами – сформированные кластеры (на дендрограмме изображаются точками). Важной особенностью дендрограммы является расстояние между кластерами различных уровней иерархии, которое должно быть пропорциональным условному расстоянию между самими кластерами, которое в свою очередь зависит от выбранной исследователем метрики степени схожести рассматриваемых объектов.

Построение дендрограмм – это стандартный функционал специализированного программного обеспечения (например, Statistica) или библиотек языков программирования, специализирующихся на обработке данных с помощью математических методов, таких как R или Python. Но такие дендрограммы, как правило, сложно стилизовать, т.е. изменять внешний вид дендрограммы и направление самого дерева иерархии. Также стандартные инструменты визуализируют сразу всё дерево, что делает дендрограмму слишком громоздкой и практически нечитаемой при большом количестве элементов. Как правило, уже при 100 исходных объектах

листовые узлы на такой дендрограмме практически невозможно рассмотреть. Что уж говорить про выборку из нескольких тысяч объектов.

Удобным решением является визуализация дендрограммы в виде интерактивного изображения, поддерживающего скрытие и отображение отдельных ветвей иерархического дерева. Такой подход позволяет облегчить работу с деревом, содержащим большое количество объектов, оптимизировать использование памяти для отображения дендрограммы с большим количеством объектов за счёт возможности постепенной подгрузки в память отдельных ветвей дендрограммы при отображении требуемой ветви.

Таким образом, целью данной работы является разработка библиотеки для визуализации иерархического дерева, содержащего результаты агломеративной иерархической кластеризации.

В качестве средства реализации библиотеки выбран язык программирования JavaScript. Это основной язык программирования для разработки клиентской части веб-приложений, которые, в свою очередь, являются наиболее востребованным типом приложений в современных информационных технологиях.

К разрабатываемой библиотеке предъявляются требования гибкой настройки визуального отображения дендрограмм, такие как цвета, толщина и стили линий, направление дендрограммы (вертикально сверху вниз, вертикально снизу вверх, горизонтально слева направо, горизонтально справа налево), количество отображаемых по умолчанию уровней иерархии, шрифтов для подписей листовых узлов и т.д.

Существующих библиотек с подобным функционалом авторами найдено не было. Поэтому работа представляется авторами актуальной в сфере обработки больших объёмов информации.

Результаты и их обсуждение

Рассмотрим формальную постановку задачи кластеризации:

Пусть X – множество объектов, Y – множество номеров (названий, меток) кластеров. Задана функция расстояния между объектами $\rho(x, x')$. Имеется конечная обучающая выборка объектов $X^m = (x_1, \ldots, x_m) \subset X$. Требуется разбить выборку на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из объектов, близких в смысле метрики ρ, а объекты разных кластеров существенно отличались. При этом каждому объекту $x_i \in X^m$ приписывается номер кластера y_i.

Алгоритм кластеризации – это функция $\alpha: X \to Y$, которая любому объекту $x \in X$ ставит в соответствие номер кластера $y \in Y$. Множество Y в некоторых случаях известно заранее (например, для алгоритма k средних), однако чаще ставится задача определить оптимальное число кластеров, с точки зрения того или иного критерия качества кластеризации.

Кластеризация (обучение без учителя) отличается от классификации (обучения с учителем) тем, что метки исходных объектов y_i изначально не заданы, и даже может быть неизвестно само множество Y.

Решение задачи кластеризации принципиально неоднозначно, и тому есть несколько причин:

– не существует однозначно наилучшего критерия качества кластеризации. Известен целый ряд эвристических критериев, а также ряд алгоритмов, не имеющих четко выраженного критерия, но осуществляющих достаточно разумную кластеризацию «по построению». Все они могут давать разные результаты.

– число кластеров, как правило, неизвестно заранее и устанавливается в соответствии с некоторым субъективным критерием.

– результат кластеризации существенно зависит от метрики, выбор которой, как правило, также субъективен и определяется экспертом.

Введём определение иерархической кластеризации – это совокупность методов и алгоритмов, которые направлены на создание иерархии (дерева) вложенных объектов. Существует два вида иерархической кластеризации:

– агломеративная – дерево создается от листьев к стволу, более мелкие кластеры объединяются в более крупные;

– дивизионная – дерево создается от ствола к листьям, более крупные кластеры делятся на более мелкие.

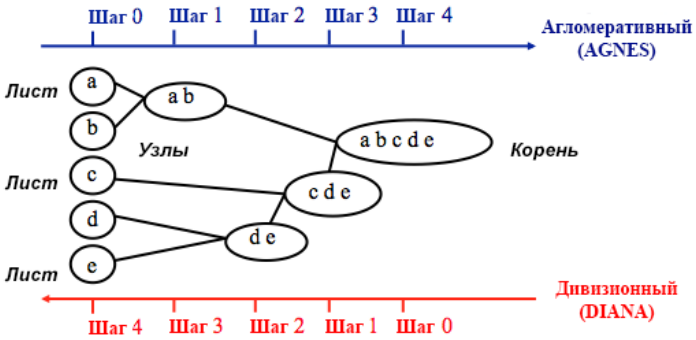

Рисунок 1 – Иерархическая кластеризация

В данной работе рассматривается агломеративная иерархическая кластеризация. Для объединения кластеров используется метод нахождения минимального расстояния между кластерами. В качестве метрики использовалась обычное Евклидово расстояние. Однако следует оговориться, что и способ вычисления расстояния между исходными объектами, и способ определения расстояния между кластерами не влияют на способ визуализации результата кластеризации. Мы приводим сведения об использованных метриках лишь для определённости, чтобы показать, на каких данных проверялась работа разработанной библиотеки.

Для демонстрации работы JavaScript-библиотеки была выполнена кластеризация записей из учебной базы данных, хранящей данные о книгах некоторой библиотеки и их использовании читателями. Книги сравнивались по различным критериям, таким как общее время использования книги, количество читателей, использовавших книгу, количество страниц в книге, автор книги (при вычислении расстояния по данному критерию расстояние между книгами одного и того же автора полагалось равным 0, а расстояния между книгами разных авторов – 1). Каждый критерий при вычислении расстояния между книгами рассматривался как координата в N-мерном пространстве, где N – количество критериев. В нашем примере $N = 4$. Расстояние, как было указано выше, вычисляется как Евклидова метрика, при этом значение каждого критерия приводится к безразмерному виду в диапазоне [0, 1].

Результат кластеризации для разработанной библиотеки представляется в формате JSON (см. рис. 2). Библиотека позволяет загружать эти данные с HTTP-сервера.

Рисунок 2 – Результат кластеризации в формате JSON

На данном рисунке для упрощения восприятия не отображаются исходные свойства объектов, на основе значений которых выполнялся расчёт расстояний.

Для визуализации графических примитивов и интерактивного взаимодействия с ними использовалась библиотека D3.js

В данном приложении используется стандартное чтение из файла с помощью языка программирования JavaScript, после загрузки данных из JSON-файла, они передаются в функцию для отображения данных

`loadData()`. Первый шаг создание графики на SVG:

```
var svg = d3.select("body").append("svg")
    .attr("width", width).attr("height", height).append("g")
    .attr("transform", "translate(" + margin.left + "," + margin.top + ")");
```

Далее создаем нашу иерархию, с помощью которой будем отрисовывать дерево:

```
root = d3.hierarchy(treeData, function(d) {
    return d.children;
});
root.x0 = height / 2;
root.y0 = 1500;
```

Передаем иерархию в функцию `update()` для интерактивной работы с данными:

1. Задаем координаты нашим кластерам и нормализуем значения (для красивого отображения):

```
var treeData = treemap(root);
var nodes = treeData.descendants(),
    links = treeData.descendants().slice(1);
nodes.forEach(function(d) {
    d.y = d.data.distance * 100
});
```

2. Обновляем положение кластеров на странице при открытии/закрытии родительского кластера:

```
var node = svg.selectAll('g.node')
    .data(nodes, function(d) {
        return d.id || (d.id = ++i);
    });
var nodeEnter = node.enter().append('g').attr('class', 'node')
    .attr("transform", function(d) {
        return "translate(" + source.y0 + "," + source.x0 + ")";
    }).on('click', click);
```

3. Задаем внешний вид (добавляем кружок и текст):

```
nodeEnter.append('circle').attr('class', 'node').attr('r', 1e-6)
    .style("fill", function(d) {
        return d._children ? "lightsteelblue" : "#fff";
    });
nodeEnter.append('text').attr("dy", ".35em")
    .attr("x", function(d) {
        return d.children || d._children ? -13 : 13;
    })
    .attr("text-anchor", function(d) {
        return d.children || d._children ? "end" : "start";
    })
    .text(function(d) {
        return d.data.name;
    });
```

4. Отрисовка диагонали (линии), которая соединяет кластеры:

```
function diagonal(s, d) {
    return `M ${s.y} ${s.x} C ${(s.y + d.y) / 2} ${s.x},
              ${(s.y + d.y) / 2} ${d.x}, ${d.y} ${d.x}`
}
```

Дополнительные методы для кликабельности кластеров и корректного отображения кластеров и линий, здесь не приведены ввиду громоздкости.

Результат работы показан на рисунке 3

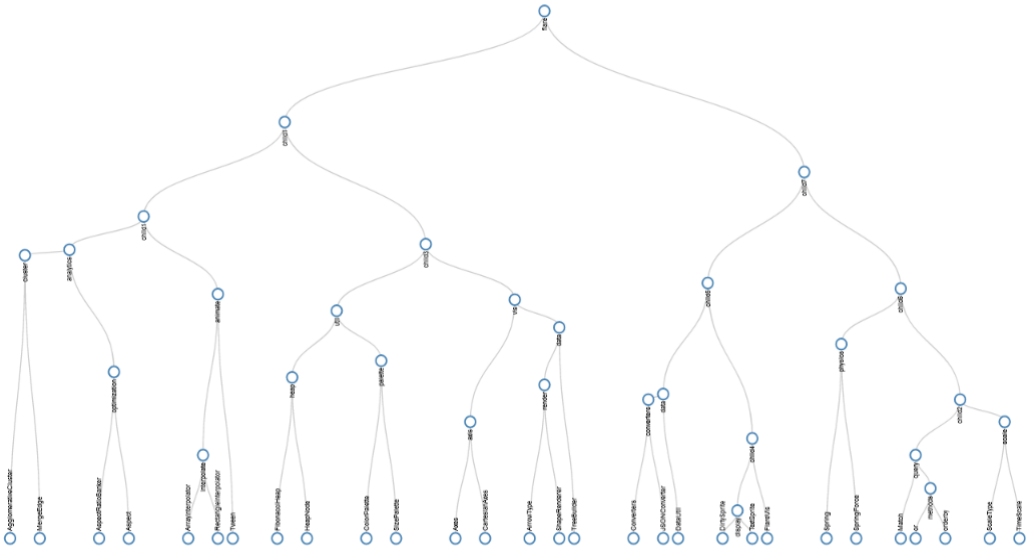

Рисунок 3 – Визуализация иерархической кластеризации

В дальнейшем планируется доработка разработанной библиотеки, добавление настроек визуализации, поддержка подгрузки данных по мере раскрытия уровней иерархии, направление дерева иерархии и т.д.

Данная библиотека используется для анализа данных об успеваемости студентов факультета математики и информационных Витебского государственного университета имени П. М. Машерова, выполняемого в рамках научно-исследовательской работы «Методы искусственного интеллекта для оптимизации образовательного процесса, №ГР 20210790» в рамках задания «Информационные технологии повышения качества образовательного процесса» государственной программы научных исследований «Цифровые и космические технологии, безопасность человека, общества и государства», подпрограммы «Цифровые технологии и космическая информатика» на 2021-2025 годы

Литература

1. Мандель, И. Д. Кластерный анализ / И. Д. Мандель. – Москва: Финансы и статистика, 1988. – 176 с.

2. Жамбю, М. Иерархический кластер-анализ и соответствия / М. Жамбю. – Москва: Финансы и статистика, 1988. – 345 с.

Ермоченко С.А., Ильина Н.А. Библиотека визуализации результатов иерархической агломеративной кластеризации. *Рассматривается разработка библиотеки на языке программирования JavaScript, которая позволяет выполнять построение дендрограммы, представляющей собой графическое изображение иерархической структуры результатов кластеризации.*

Ключевые слова: *иерархическая агломеративная кластеризация, дендрограмма, язык программирования JavaScript, JSON.*

Yermochenko Sergey, Ilyina Nadezhda. Library for visualization of hierarchical agglomerative clustering results. *The article considers the development of the library for JavaScript programming language, which allow building a dendrogram as graphical image of hierarchical structure of clustering results.*

Key words: *hierarchical agglomerative clustering, dendrogram, JavaScript programming language, JSON.*