ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Образовательная программа «Прикладная математика и информатика»

Отчет о программном проекте

на тему разработка системы предсказания успешного завершения учебной дисциплины
(промежуточный, этап 1)

Выполнил:
студент группы БПМИ186 _____ Чутларов Денис Андреевич
Подпись И.О. Фамилия
7.02.2020
Дата

Принял:
руководитель проекта Андрей Андреевич Ларионов
Имя, Отчество, Фамилия
младший научный сотрудник
Должность
МНУЛ ИССА ФКН НИУ ВШЭ
Место работы

Дата 07.02.2020 10 _____
Оценка (по 10-тибалльной шкале) Подпись

Москва 2020

Содержание

Задача первого этапа проекта 3
Агломеративная кластеризация 3
Спектральная кластеризация 4
Ссылки на литературу 7

Задача первого этапа проекта

Изучение и сравнение алгоритмов кластеризации

Агломеративная кластеризация

Идея

Объединять объекты в кластер, используя некоторую меру сходства или расстояние между объектами.

Шаги алгоритма

1) Присваиваем каждому объекту свой кластер

2) Сортируем попарные расстояния между кластерами

3) Берём пару ближайших кластеров, склеиваем их в один

4) Повторяем 2 и 3 пункт пока все объекты не попадут в один кластер

Методы объединения точек

1) *Метод одиночной связи* – в основе этого метода лежит минимальное расстояние

2) *Метод полной связи* – в основе этого метода лежит максимальное расстояние между объектами

3) *Метод средней связи* – в основе этого метода лежит среднее расстояние между каждой парой объектов из разных кластеров

4) *Центроидный метод* – в основе этого метода лежит минимальное расстояние между центрами разных кластеров

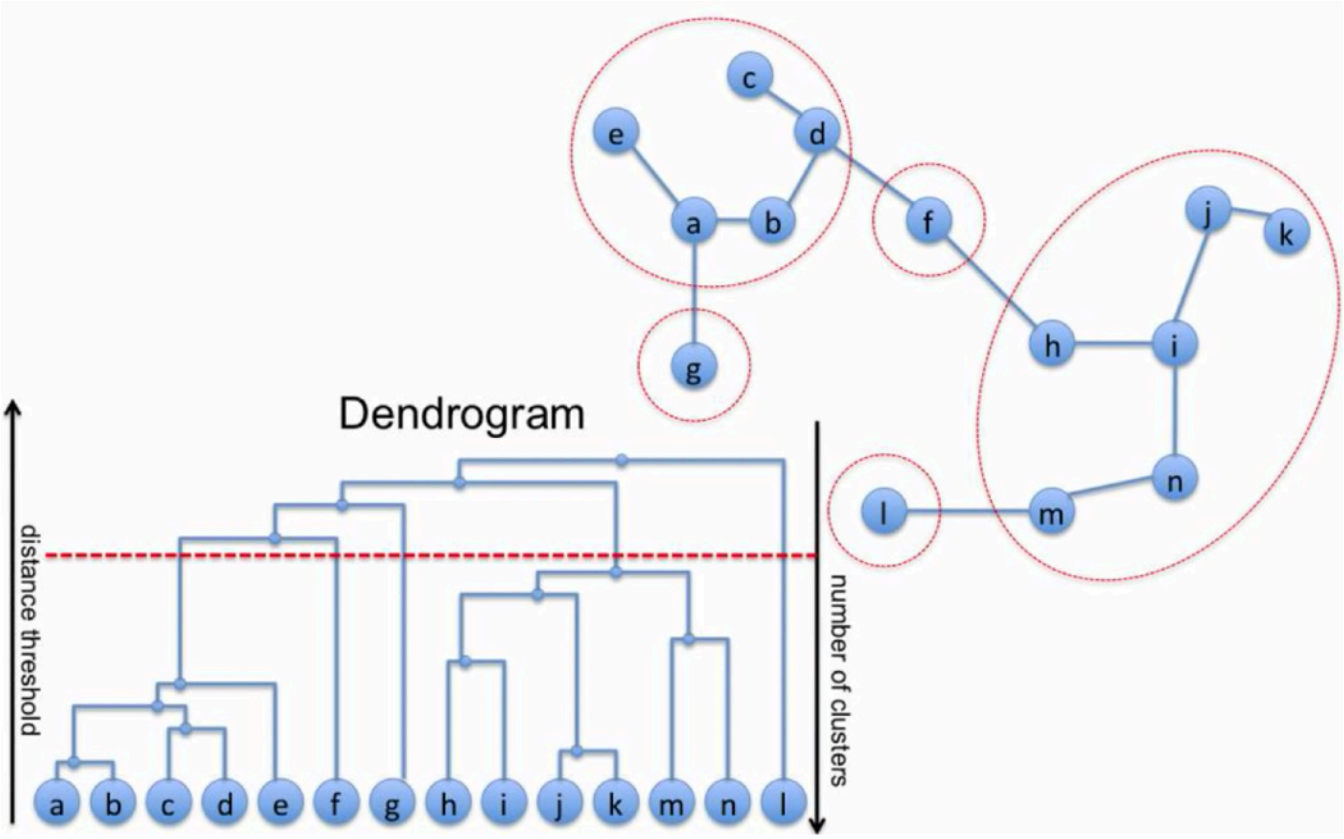
Сложность алгоритма

Для одиночной и полной связи: $O(n^2c + n^3)$

Для средней связи: $O(n^2c + n^3d)$, $d -$ время вычисления расстояния в одной паре

Результат работы такого алгоритма

Дерево склеивания кластеров - граф показывающий работу алгоритма, как следствие показывает, на каком этапе нужно остановить алгоритм



Спектральная кластеризация

Определение

Спектральной кластеризацией называются все методы, которые разбивают множество на кластеры с помощью собственных векторов матрицы S или других матриц, полученных из нее

Алгоритмы

Матрица сходства A – элементы A_{ij} представляют меру схожести между точками данных с индексами i и j.

D – диагональная матрица $D_{ii} = \sum_j A_{ij}$

1) Алгоритм нормализованных сечений:
Алгоритм разбивает точки на два множества, основываясь на собственном векторе, соответствующем второму по величине собственному значению симметрично нормализованной матрицы Кирхгофа, задаваемой формулой:

$L^{norm} := I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$

2) Другой способ:

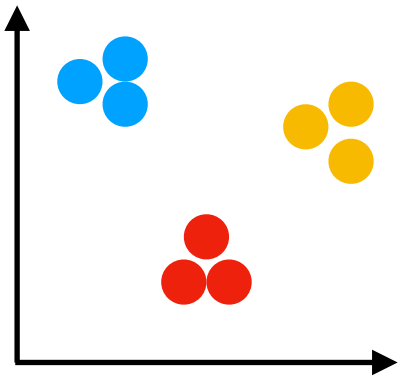
$L := D - A$, в этом случае разбиение можно делать несколькими способами, в нашем случае это:

1) Вычисление медианы компонент второго наименьшего собственного вектора v
2) Затем точки, чьи компоненты в v больше медианы, кладем в 1 кластер, остальные в другой
3) Повторяем 1 и 2 пункт

Изменение представления, созданного собственными векторами, позволяет более очевидным образом задать свойства исходного набора кластеров.

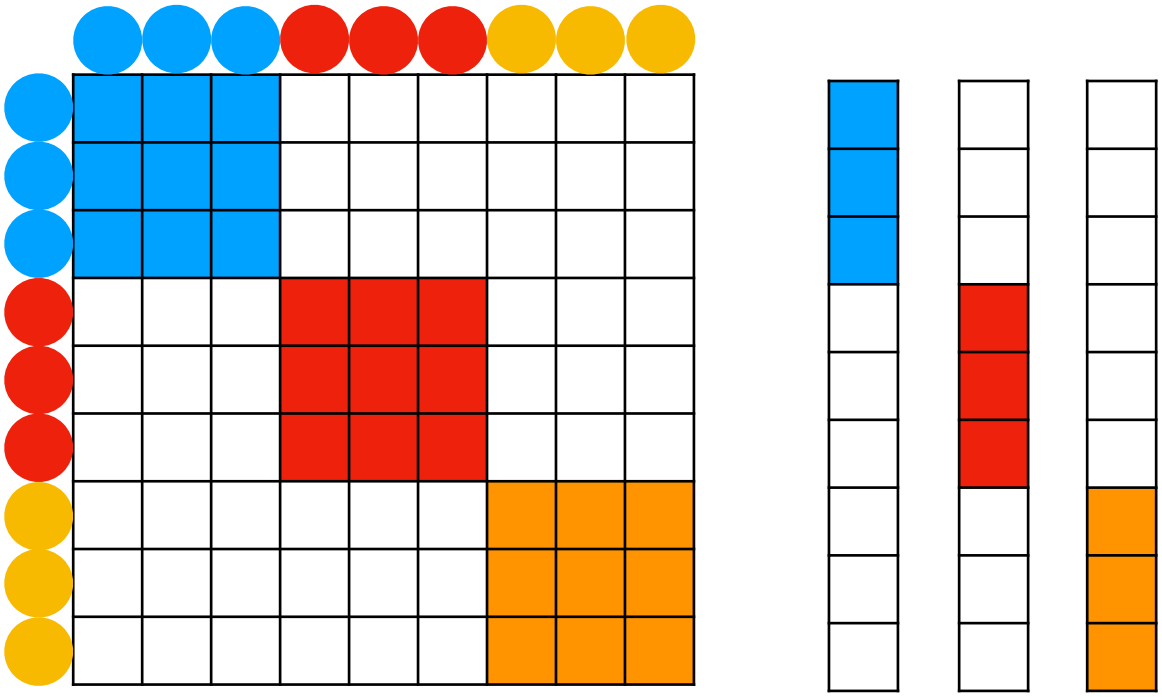
Иллюстрация работы алгоритма

Пусть нам на вход подаются следующие точки:



Цвета сделаны для удобства

Построим матрицу сходства для этих точек:



Цветом отметим места с максимальным сходством.

Найдем собственный вектор:

Видно, что собственный вектор разбил точки на нужные кластеры

Ссылки на литературу

Агломеративная кластеризация:

1) YouTube: https://www.youtube.com/watch?v=XJ3194AmH40
2) Habr: https://habr.com/ru/company/ods/blog/325654/
3) Wikipedia: https://clck.ru/JuWgj
4) https://lektsii.org/2-87430.html

Спектральная кластеризация:

1) Wikipedia: https://clck.ru/MAby3
2) Habr: https://habr.com/ru/company/ods/blog/325654/
3) YouTube: https://youtu.be/zkgm0i77jQ8
4) https://nsu.ru/xmlui/bitstream/handle/nsu/463/Text_MachulskisSV.pdf

Содержание

Задача первого этапа проекта 3
Агломеративная кластеризация 3
Спектральная кластеризация 4
Задача которого этапа проекта: 7
Клиент серверная-архитектура API 7
REST и RESTful 7
Flask 8
Ссылки на литературу 9

Задача первого этапа проекта

Изучение и сравнение алгоритмов кластеризации

Агломеративная кластеризация

Идея

Объединять объекты в кластер, используя некоторую меру сходства или расстояние между объектами.

Шаги алгоритма

1) Присваиваем каждому объекту свой кластер

2) Сортируем попарные расстояния между кластерами

3) Берём пару ближайших кластеров, склеиваем их в один

4) Повторяем 2 и 3 пункт пока все объекты не попадут в один кластер

Методы объединения точек

1) *Метод одиночной связи* – в основе этого метода лежит минимальное расстояние

2) *Метод полной связи* – в основе этого метода лежит максимальное расстояние между объектами

3) *Метод средней связи* – в основе этого метода лежит среднее расстояние между каждой парой объектов из разных кластеров

4) *Центроидный метод* – в основе этого метода лежит минимальное расстояние между центрами разных кластеров

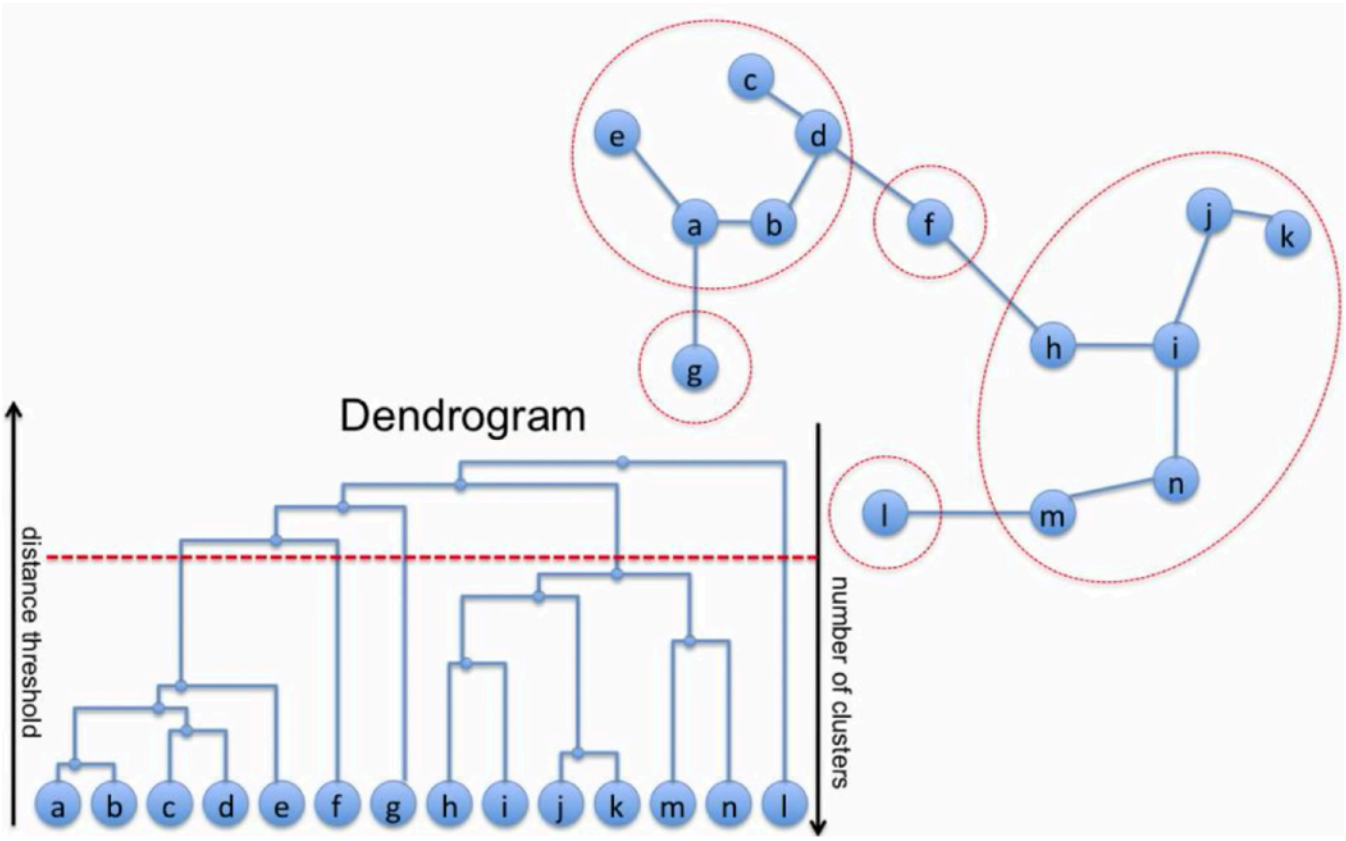
Сложность алгоритма

Для одиночной и полной связи: $O(n^2c + n^3)$

Для средней связи: $O(n^2c + n^3d),\ d -$ время вычисления расстояния в одной паре

Результат работы такого алгоритма

Дерево склеивания кластеров - граф показывающий работу алгоритма, как следствие показывает, на каком этапе нужно остановить алгоритм



Спектральная кластеризация

Определение

Спектральной кластеризацией называются все методы, которые разбивают множество на кластеры с помощью собственных векторов матрицы S или других матриц, полученных из нее

Алгоритмы

Матрица сходства A – элементы A_{ij} представляют меру схожести между точками данных с индексами i и j.

D – диагональная матрица $D_{ii} = \sum_j A_{ij}$

1) Алгоритм нормализованных сечений:

Алгоритм разбивает точки на два множества, основываясь на собственном векторе, соответствующем второму по величине собственному значению симметрично нормализованной матрицы Кирхгофа, задаваемой формулой:

$L^{norm} := I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$

2) Другой способ:

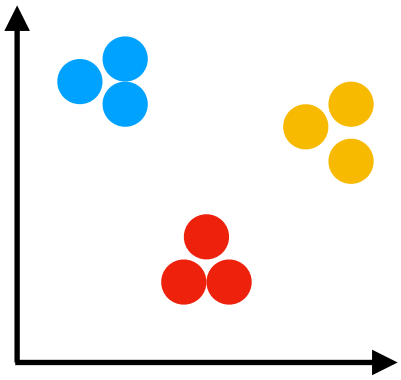
$L := D - A$, в этом случае разбиение можно делать несколькими способами, в нашем случае это:

1) Вычисление медианы компонент второго наименьшего собственного вектора v
2) Затем точки, чьи компоненты в v больше медианы, кладем в 1 кластер, остальные в другой
3) Повторяем 1 и 2 пункт

Изменение представления, созданного собственными векторами, позволяет более очевидным образом задать свойства исходного набора кластеров.

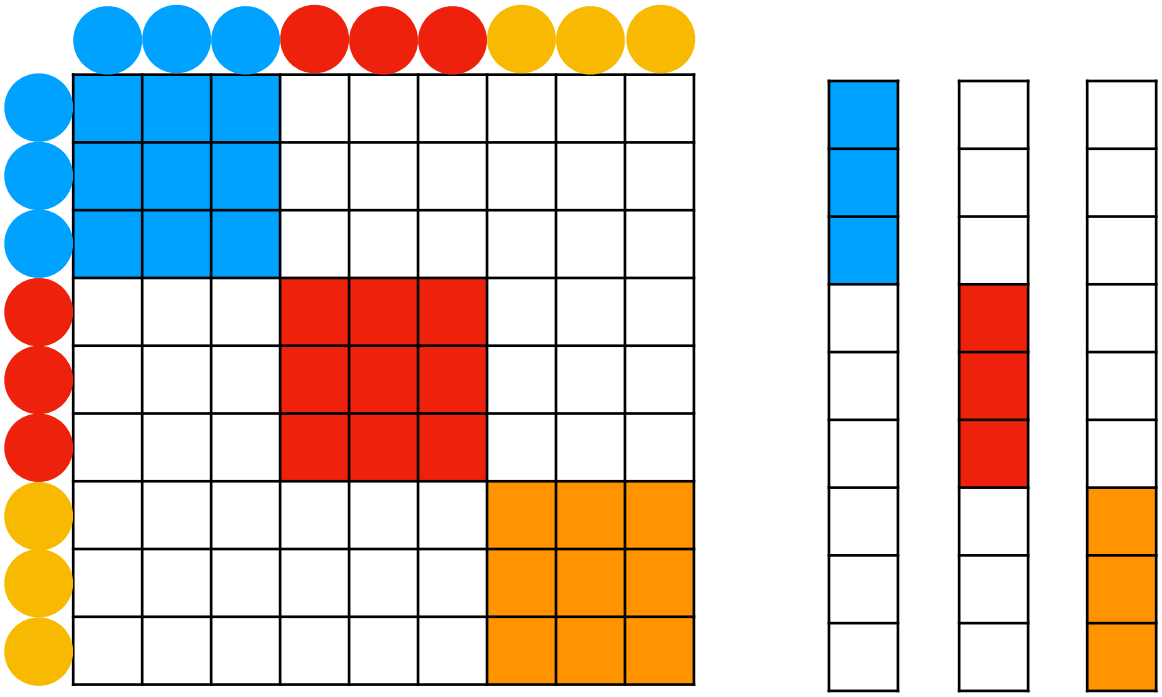
Иллюстрация работы алгоритма

Пусть нам на вход подаются следующие точки:



Цвета сделаны для удобства

Построим матрицу сходства для этих точек:



Цветом отметим места с максимальным сходством.

Найдем собственный вектор:

Видно, что собственный вектор разбил точки на нужные кластеры

Задача которого этапа проекта:

Реализация клиент-серверной архитектуры

Клиент серверная-архитектура API

API (от англ. ***a****pplication* ***p****rogramming* ***i****nterface*) — описание способов которыми одна компьютерная программа может взаимодействовать с другой программой. Обычно входит в описание какого-либо интернет-протокола программного каркаса (фреймворка). Используется программистами при написании всевозможных приложений.

REST и RESTful

REST – аббревиатура от Representational State Transfer («передача состояния представления»). Это согласованный набор архитектурных принципов для создания более масштабируемой и гибкой сети.

Архитектурные принципы:

1) **Клиент-сервер**

   Первое ограничение указывает, что сеть должна состоять из клиентов и серверов. Сервер — это компьютер, который имеет требуемые ресурсы, а клиент — это компьютер, которому нужно взаимодействовать с ресурсами, хранящимися на сервере.

2) **Отсутствие состояния**

   Клиент и сервер не отслеживают состояние друг друга. Когда клиент не взаимодействует с сервером, сервер не имеет представления о его существовании.

3) **Единообразие интерфейса**

   Ограничение гарантирует, что между серверами и клиентами существует общий язык, который позволяет каждой части быть заменяемой или изменяемой, без нарушения целостности системы. Это достигается через 4 дополнительных ограничения: идентификация ресурсов, манипуляция ресурсами через представления, «самодостаточные» сообщения и гипермедиа.

4) **Кэширование**

   Ответы сервера должны помечаться как кэшируемые или некэшируемые. Кэшируемый ответы сохраняются у пользователя.

Flask

Для реализации моего сервера я выбрал фреймворк Flask. Flask – это фреймворк, служащий для создания вебсайтов на языке Python.
Я написал два класса: Experiment и ExperimentsList.
http://127.0.0.1/experiments/get<int:exp_id>
показывает результат эксперимента<int:exp_id>

http: // 127.0.0.1/experiments/post<>
создает новый эксперимен т

Ссылки на литературу

Агломеративная кластеризация:

1) YouTube: https://www.youtube.com/watch?v=XJ3194AmH40
2) Habr: https://habr.com/ru/company/ods/blog/325654/
3) Wikipedia: https://clck.ru/JuWgj
4) https://lektsii.org/2-87430.html

Спектральная кластеризация:

1) Wikipedia: https://clck.ru/MAby3
2) Habr: https://habr.com/ru/company/ods/blog/325654/
3) YouTube: https://youtu.be/zkgm0i77jQ8
4) https://nsu.ru/xmlui/bitstream/handle/nsu/463/Text_MachulskisSV.pdf

API

1) https://ru.wikipedia.org/wiki/API

REST и RESTful

1) https://habr.com/ru/company/hexlet/blog/274675/
2) https://habr.com/ru/company/dataart/blog/277419/

Flask

1) https://www.freecodecamp.org/news/build-a-simple-json-api-in-python/
2) https://flask-restful.readthedocs.io/en/latest/