



После прохождения всех тестов, пользователь будет иметь полную картину по психологическим характеристикам.

Исходя из вышесказанного, можно утверждать, что тема формирования психологического профиля молодых исследователей является актуальной как для самих студентов, так и для общества в целом. Данная тема является одним из направлений *People-аналитики*, которая в свою очередь является очень перспективной.

Литература

1. Гордеев С. Ю., Никифорова Т. В., Статья «Информационная система формирования данных для моделирования развития научных способностей творчески одаренной молодежи» в сборнике научных трудов конференции ПИТ 2017: URL: http://www.ssau.ru/pagefiles/pit_2017_sbornik.pdf (дата обращения 3.12.2018)

2. People Analytics: советы Google: URL: <https://hr-portal.ru/blog/people-analytics-sovety-google> (дата обращения: 25.05.19).

3. People analytics—A scoping review of conceptual boundaries and value propositions / Tursunbayeva Aizhan, Di Lauro Stefano, Pagliari Claudia // Int. J. Inf. Manag.— 2018 т. 43.— С. 224-247.

4. Википедия. Психологический тест: URL: <https://ru.wikipedia.org/wiki/%D0%9F> (дата обращения: 12.12.2018).

5. Дерябкин, В. П. Проектирование информационных систем по методологии UML [Текст]: метод. Указания к учебно-лабораторному практикуму / В. П. Дерябкин, В. В. Козлов; Самарск. гос. арх.-строит. ун-т. - Самара, 2008.

6. Диаграмма состояний (statechart diagram) http://www.telenir.net/uchebniki/samouchitel_uml/p6.php

Д.М. Никишин, З.Ф. Камальдинова

КЛАСТЕРИЗАЦИЯ ПИКСЕЛЕЙ ИЗОБРАЖЕНИЯ ПРИ ПОМОЩИ АЛГОРИТМА «К-СРЕДНИХ»

(Самарский государственный технический университет)

В наше время нас окружает огромное количество информации обо всём на свете. И одна из задач в работе с информацией - это разгруппировать поступающие данные на всевозможные классы, группы и т.д., для того, чтобы с этой информацией было удобно взаимодействовать. Но иногда нам сложно отнести какую-то вещь к чему-то определённом. И тогда в дело вступает кластеризация.

Кластеризация - это задача разбиения множества объектов на группы, называемые кластерами. Внутри каждой группы должны оказаться «похожие» объекты, а объекты разных группы должны быть как можно более отличны.



Главное отличие кластеризации от классификации состоит в том, что перечень групп четко не задан и определяется в процессе работы алгоритма [1].

Кластерный анализ в отличие от большинства математико-статистических методов не накладывает никаких ограничений на вид рассматриваемых объектов и позволяет исследовать множество исходных данных практически произвольной природы [2].

Обычно, применение кластерного анализа сводится к следующим этапам:

1. Отбор выборки объектов для кластеризации.
2. Определение множества переменных, по которым будут оцениваться объекты в выборке. При необходимости - нормализация значений переменных.
3. Вычисление значений меры сходства между объектами.
4. Применение метода кластерного анализа для создания групп сходных объектов (кластеров).
5. Представление результатов анализа [3].

После получения и анализа результатов возможна корректировка выбранной метрики и метода кластеризации до получения оптимального результата. Стоит также сказать, что решение задачи кластеризации принципиально неоднозначно в силу следующих причин:

1. Не существует однозначно наилучшего критерия качества кластеризации.
2. Число кластеров, как правило, неизвестно заранее и устанавливается в соответствии с некоторым субъективным критерием.
3. Результат кластеризации существенно зависит от метрики, выбор которой также субъективен и определяется экспертом.

Для человека незнакомого с кластеризацией возникает разумный вопрос: а как же определить эту самую «похожесть» объектов? Для начала нужно составить вектор характеристик для каждого объекта - как правило, это набор числовых значений, например, рост и вес человека. Однако существуют также алгоритмы, работающие с качественными (то бишь категориальными) характеристиками [4].

Среди алгоритмов кластеризации выделяются два основных типа: восходящие и нисходящие алгоритмы. Нисходящие алгоритмы работают по принципу «сверху-вниз»: в начале все объекты помещаются в один кластер, который затем разбивается на все более мелкие кластеры. Более распространены восходящие алгоритмы, которые в начале работы помещают каждый объект в отдельный кластер, а затем объединяют кластеры во все более крупные, пока все объекты выборки не будут содержаться в одном кластере. Таким образом, строится система вложенных разбиений [5].

Для данного проекта в качестве объекта кластеризации были выбраны пиксели изображения, потому что это простой, а самое главное - понятный пример. Как известно, цветной пиксель имеет три составляющих цвета - Red, Green и Blue (Красный, Зелёный и Синий). Наложение этих составляющих и создает палитру существующих цветов. В памяти компьютера каждая составляющая цвета характеризуется числом от 0 до 255. То есть комбинируя



различные значения красного, зеленого и синего, получаем палитру цветов на экране.

Среди различных реализаций был выбран на алгоритм k-средних, поскольку это простой и эффективный метод, который легко реализовать программно. Соответственно, данные, которые мы будем распределять по кластерам - наши пиксели. На их примере мы и реализуем наш алгоритм. Алгоритм k-средних - итерационный алгоритм, то есть он даст правильный результат, после *n* количества повторов некоторых математических вычислений.

Программа была написана в IDE Visual Studio 2017 Community на языке C++.

Начнём с файла Clusters.h. Для характеристики каждого пикселя была создана структура, которая состоит из трех составляющих пикселя типа double для большой точности. И здесь же был создан сам класс **Clusters**, в котором определили переменные, константы и его методы.

Прокомментируем составляющие нашего класса:

vect_pixcel - вектор для пикселей; vect_centр - вектор для центров кластеров; kolvo_klasters - количество кластеров; kolvo_pixels - количество пикселей; identify_centers() - метод для случайного выбора начальных центров среди входных пикселей; compute() и compute_s() - методы для расчета расстояния между пикселями и пересчета центров; Clusters() - конструктор класса; clustering() - сам метод кластеризации.

Это программа отлично подойдёт для целей сегментации изображений (она заключается в упрощении и/или изменении представления изображения, чтобы его было проще и легче анализировать, в частности - сегментация обычно используется для того, чтобы выделить объекты и границы (линии, кривые, и т.д.) на изображениях).

Планируется развивать данную работу в сторону точности расчётов радиусов и вычисления расстояний, а также в сторону упрощения интерфейса, чтобы пользователю было легче и комфортнее работать с программой.

Литература

1. Обзор алгоритмов кластеризации данных [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/101338/> (дата обращения: 22.12.2019).
2. Технологии анализа данных: учеб. пособие / Н.П. Садовникова, М.В. Щербakov. – Волгоград: Изд-во ВолгГТУ, 2015. – 64 с.
3. Кластерный анализ. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Кластерный_анализ (дата обращения: 23.12.2019).
4. Алгоритмы кластеризации на службе Data Mining [Электронный ресурс]. – Режим доступа: <https://basegroup.ru/community/articles/datamining> (дата обращения: 24.12.2019).
5. Кластеризация: расскажи мне, что ты покупаешь, и я скажу кто ты [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/company/datawiz/blog/248863/> (дата обращения: 22.12.2019).