МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**КУРСОВАЯ РАБОТА**

**АТОМАТИЗАЦИЯ ОБРАБОТКИ БОЛЬШИХ ДАННЫХ В ЯЗЫКЕ PYTHON**

Работу выполнил Н.В. Богайчук   
(подпись)

Направление подготовки 02.03.02 «Фундаментальная информатика и информационные технологии» курс 3

Направленность (профиль) «Вычислительные технологии»

Научный руководитель

канд. техн. наук доц. Т.А.Приходько (подпись)

Нормоконкролер

канд. техн. наук доц. Е.Е.Полупанова (подпись)

Краснодар 2019

Содержание

[ВВЕДЕНИЕ 3](#_Toc20828313)

[1 Реализация 4](#_Toc20828314)

[1.1 Постановка задачи 4](#_Toc20828315)

## ВВЕДЕНИЕ

# 1 Реализация

## 1.1 Постановка задачи

Обработка больших данных в Python

Язык программирования **Python** в последнее время все чаще используется для анализа данных, как в науке, так и коммерческой сфере. Этому способствует простота языка, а также большое разнообразие открытых библиотек.

## Создание классификатора в Python

Scikit-learn, библиотека Python для машинного обучения может использоваться для построения классификатора в Python. Шаги для создания классификатора в Python следующие:

Кластеризация

Кластеризация (или кластерный анализ) — 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.

Применение кластерного анализа в общем виде сводится к следующим этапам:

Отбор выборки объектов для кластеризации.

Определение множества переменных, по которым будут оцениваться объекты в выборке. При необходимости – нормализация значений переменных.

Вычисление значений меры сходства между объектами.

Применение метода кластерного анализа для создания групп сходных объектов (кластеров).

Представление результатов анализа.

После получения и анализа результатов возможна корректировка выбранной метрики и метода кластеризации до получения оптимального результата.

Меры расстояний

Итак, как же определять «похожесть» объектов? Для начала нужно составить вектор характеристик для каждого объекта — как правило, это набор числовых значений, например, рост-вес человека. Однако существуют также алгоритмы, работающие с качественными (т.н. категорийными) характеристиками.

После того, как мы определили вектор характеристик, можно провести нормализацию, чтобы все компоненты давали одинаковый вклад при расчете «расстояния». В процессе нормализации все значения приводятся к некоторому диапазону, например, [-1, -1] или [0, 1].

Наконец, для каждой пары объектов измеряется «расстояние» между ними — степень похожести. Существует множество метрик, вот лишь основные из них:

Евклидово расстояние

Наиболее распространенная функция расстояния. Представляет собой геометрическим расстоянием в многомерном пространстве:

Квадрат евклидова расстояния

Применяется для придания большего веса более отдаленным друг от друга объектам. Это расстояние вычисляется следующим образом:

Расстояние городских кварталов (манхэттенское расстояние)

Это расстояние является средним разностей по координатам. В большинстве случаев эта мера расстояния приводит к таким же результатам, как и для обычного расстояния Евклида. Однако для этой меры влияние отдельных больших разностей (выбросов) уменьшается (т.к. они не возводятся в квадрат). Формула для расчета манхэттенского расстояния:

Расстояние Чебышева

Это расстояние может оказаться полезным, когда нужно определить два объекта как «различные», если они различаются по какой-либо одной координате. Расстояние Чебышева вычисляется по формуле:

Степенное расстояние

Применяется в случае, когда необходимо увеличить или уменьшить вес, относящийся к размерности, для которой соответствующие объекты сильно отличаются. Степенное расстояние вычисляется по следующей формуле:

,

где r и p – параметры, определяемые пользователем. Параметр p ответственен за постепенное взвешивание разностей по отдельным координатам, параметр r ответственен за прогрессивное взвешивание больших расстояний между объектами. Если оба параметра – r и p — равны двум, то это расстояние совпадает с расстоянием Евклида.

Выбор метрики полностью лежит на исследователе, поскольку результаты кластеризации могут существенно отличаться при использовании разных мер.

Классификация алгоритмов

Для себя я выделил две основные классификации алгоритмов кластеризации.

Иерархические и плоские.

Иерархические алгоритмы (также называемые алгоритмами таксономии) строят не одно разбиение выборки на непересекающиеся кластеры, а систему вложенных разбиений. Т.о. на выходе мы получаем дерево кластеров, корнем которого является вся выборка, а листьями — наиболее мелкие кластера.

Плоские алгоритмы строят одно разбиение объектов на кластеры.

Четкие и нечеткие.

Четкие (или непересекающиеся) алгоритмы каждому объекту выборки ставят в соответствие номер кластера, т.е. каждый объект принадлежит только одному кластеру. Нечеткие (или пересекающиеся) алгоритмы каждому объекту ставят в соответствие набор вещественных значений, показывающих степень отношения объекта к кластерам. Т.е. каждый объект относится к каждому кластеру с некоторой вероятностью.

Объединение кластеров

В случае использования иерархических алгоритмов встает вопрос, как объединять между собой кластера, как вычислять «расстояния» между ними. Существует несколько метрик:

Одиночная связь (расстояния ближайшего соседа)

В этом методе расстояние между двумя кластерами определяется расстоянием между двумя наиболее близкими объектами (ближайшими соседями) в различных кластерах. Результирующие кластеры имеют тенденцию объединяться в цепочки.

Полная связь (расстояние наиболее удаленных соседей)

В этом методе расстояния между кластерами определяются наибольшим расстоянием между любыми двумя объектами в различных кластерах (т.е. наиболее удаленными соседями). Этот метод обычно работает очень хорошо, когда объекты происходят из отдельных групп. Если же кластеры имеют удлиненную форму или их естественный тип является «цепочечным», то этот метод непригоден.

Невзвешенное попарное среднее

В этом методе расстояние между двумя различными кластерами вычисляется как среднее расстояние между всеми парами объектов в них. Метод эффективен, когда объекты формируют различные группы, однако он работает одинаково хорошо и в случаях протяженных («цепочечного» типа) кластеров.

Взвешенное попарное среднее

Метод идентичен методу невзвешенного попарного среднего, за исключением того, что при вычислениях размер соответствующих кластеров (т.е. число объектов, содержащихся в них) используется в качестве весового коэффициента. Поэтому данный метод должен быть использован, когда предполагаются неравные размеры кластеров.

Невзвешенный центроидный метод

В этом методе расстояние между двумя кластерами определяется как расстояние между их центрами тяжести.

Взвешенный центроидный метод (медиана)

Этот метод идентичен предыдущему, за исключением того, что при вычислениях используются веса для учета разницы между размерами кластеров. Поэтому, если имеются или подозреваются значительные отличия в размерах кластеров, этот метод оказывается предпочтительнее предыдущего.

Обзор алгоритмов

Алгоритмы иерархической кластеризации

Среди алгоритмов иерархической 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. Результаты таких алгоритмов обычно представляют в виде дерева – дендрограммы. Классический пример такого дерева – классификация животных и растений.

Для вычисления расстояний между кластерами чаще все пользуются двумя расстояниями: одиночной связью или полной связью (см. обзор мер расстояний между кластерами).

К недостатку иерархических алгоритмов можно отнести систему полных разбиений, которая может являться излишней в контексте решаемой задачи.

Алгоритмы квадратичной ошибки

Задачу кластеризации можно рассматривать как построение оптимального разбиения объектов на группы. При этом оптимальность может быть определена как требование минимизации среднеквадратической ошибки разбиения:

где cj — «центр масс» кластера j (точка со средними значениями характеристик для данного кластера).

Алгоритмы квадратичной ошибки относятся к типу плоских алгоритмов. Самым распространенным алгоритмом этой категории является метод k-средних. Этот алгоритм строит заданное число кластеров, расположенных как можно дальше друг от друга. Работа алгоритма делится на несколько этапов:

Случайно выбрать k точек, являющихся начальными «центрами масс» кластеров.

Отнести каждый объект к кластеру с ближайшим «центром масс».

Пересчитать «центры масс» кластеров согласно их текущему составу.

Если критерий остановки алгоритма не удовлетворен, вернуться к п. 2.

В качестве критерия остановки работы алгоритма обычно выбирают минимальное изменение среднеквадратической ошибки. Так же возможно останавливать работу алгоритма, если на шаге 2 не было объектов, переместившихся из кластера в кластер.

К недостаткам данного алгоритма можно отнести необходимость задавать количество кластеров для разбиения.

Нечеткие алгоритмы

Наиболее популярным алгоритмом нечеткой кластеризации является алгоритм c-средних (c-means). Он представляет собой модификацию метода k-средних. Шаги работы алгоритма:

Выбрать начальное нечеткое разбиение n объектов на k кластеров путем выбора матрицы принадлежности U размера n x k.

Используя матрицу U, найти значение критерия нечеткой ошибки:

,

где ck — «центр масс» нечеткого кластера k:

.

Перегруппировать объекты с целью уменьшения этого значения критерия нечеткой ошибки.

Возвращаться в п. 2 до тех пор, пока изменения матрицы U не станут незначительными.

Этот алгоритм может не подойти, если заранее неизвестно число кластеров, либо необходимо однозначно отнести каждый объект к одному кластеру.

Алгоритмы, основанные на теории графов

Суть таких алгоритмов заключается в том, что выборка объектов представляется в виде графа G=(V, E), вершинам которого соответствуют объекты, а ребра имеют вес, равный «расстоянию» между объектами. Достоинством графовых алгоритмов кластеризации являются наглядность, относительная простота реализации и возможность вносения различных усовершенствований, основанные на геометрических соображениях. Основными алгоритмам являются алгоритм выделения связных компонент, алгоритм построения минимального покрывающего (остовного) дерева и алгоритм послойной кластеризации.

Алгоритм выделения связных компонент

В алгоритме выделения связных компонент задается входной параметр R и в графе удаляются все ребра, для которых «расстояния» больше R. Соединенными остаются только наиболее близкие пары объектов. Смысл алгоритма заключается в том, чтобы подобрать такое значение R, лежащее в диапазон всех «расстояний», при котором граф «развалится» на несколько связных компонент. Полученные компоненты и есть кластеры.

Для подбора параметра R обычно строится гистограмма распределений попарных расстояний. В задачах с хорошо выраженной кластерной структурой данных на гистограмме будет два пика – один соответствует внутрикластерным расстояниям, второй – межкластерным расстояния. Параметр R подбирается из зоны минимума между этими пиками. При этом управлять количеством кластеров при помощи порога расстояния довольно затруднительно.

Алгоритм минимального покрывающего дерева

Алгоритм минимального покрывающего дерева сначала строит на графе минимальное покрывающее дерево, а затем последовательно удаляет ребра с наибольшим весом. На рисунке изображено минимальное покрывающее дерево, полученное для девяти объектов.

Путём удаления связи, помеченной CD, с длиной равной 6 единицам (ребро с максимальным расстоянием), получаем два кластера: {A, B, C} и {D, E, F, G, H, I}. Второй кластер в дальнейшем может быть разделён ещё на два кластера путём удаления ребра EF, которое имеет длину, равную 4,5 единицам.

Послойная кластеризация

Алгоритм послойной кластеризации основан на выделении связных компонент графа на некотором уровне расстояний между объектами (вершинами). Уровень расстояния задается порогом расстояния c. Например, если расстояние между объектами , то .

Алгоритм послойной кластеризации формирует последовательность подграфов графа G, которые отражают иерархические связи между кластерами:

,

где Gt = (V, Et) — граф на уровне сt,

,

сt – t-ый порог расстояния,

m – количество уровней иерархии,

G0 = (V, o), o – пустое множество ребер графа, получаемое при t0 = 1,

Gm = G, то есть граф объектов без ограничений на расстояние (длину ребер графа), поскольку tm = 1.

Посредством изменения порогов расстояния {с0, …, сm}, где 0 = с0 < с1 < …< сm = 1, возможно контролировать глубину иерархии получаемых кластеров. Таким образом, алгоритм послойной кластеризации способен создавать как плоское разбиение данных, так и иерархическое.

Немного о применении

В своей работе мне нужно было из иерархических структур (деревьев) выделять отдельные области. Т.е. по сути необходимо было разрезать исходное дерево на несколько более мелких деревьев. Поскольку ориентированное дерево – это частный случай графа, то естественным образом подходят алгоритмы, основанными на теории графов.

В отличие от полносвязного графа, в ориентированном дереве не все вершины соединены ребрами, при этом общее количество ребер равно n–1, где n – число вершин. Т.е. применительно к узлам дерева, работа алгоритма выделения связных компонент упростится, поскольку удаление любого количества ребер «развалит» дерево на связные компоненты (отдельные деревья). Алгоритм минимального покрывающего дерева в данном случае будет совпадать с алгоритмом выделения связных компонент – путем удаления самых длинных ребер исходное дерево разбивается на несколько деревьев. При этом очевидно, что фаза построения самого минимального покрывающего дерева пропускается.

Можно выделить два подхода к алгоритмам:

* ***Иерархические и плоские.***
  1. ***Иерархические*** алгоритмы (также называемые алгоритмами таксономии) строят не одно разбиение выборки на непересекающиеся кластеры, а систему вложенных разбиений. Т.о. на выходе мы получаем дерево кластеров, корнем которого является вся выборка, а листьями — наиболее мелкие кластера.
  2. **Плоские** алгоритмы строят одно разбиение объектов на кластеры.
* ***Четкие и нечеткие.***
  1. ***Четкие*** (или непересекающиеся) алгоритмы каждому объекту выборки ставят в соответствие номер кластера, т.е. каждый объект принадлежит только одному кластеру.
  2. ***Нечеткие*** (или пересекающиеся) алгоритмы каждому объекту ставят в соответствие набор вещественных значений, показывающих степень отношения объекта к кластерам. Т.е. каждый объект относится к каждому кластеру с некоторой вероятностью.
  3. ***Алгоритмы, основанные на теории графов***
  4. Суть таких алгоритмов заключается в том, что выборка объектов представляется в виде графа G=(V, E), вершинам которого соответствуют объекты, а ребра имеют вес, равный «расстоянию» между объектами.
  5. Достоинством графовых алгоритмов кластеризации являются наглядность, относительная простота реализации и возможность внесения различных усовершенствований, основанных на геометрических соображениях.
  6. Основными алгоритмам являются алгоритм выделения связных компонент, алгоритм построения минимального покрывающего (остовного) дерева и алгоритм послойной кластеризации.
* ***Алгоритмы иерархической кластеризации***

Нисходящие 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. Результаты таких алгоритмов обычно представляют в виде дерева – ***дендрограммы***. Классический пример такого дерева – классификация животных и растений

***Классификация*** - системное распределение изучаемых предметов, явлений, процессов по родам, видам, типам, по каким-либо существенным признакам для удобства их исследования; группировка исходных понятий и расположение их в определенном порядке, отражающем степень этого сходства.

***Классификация*** - упорядоченное по некоторому принципу множество объектов, которые имеют сходные классификационные признаки (одно или несколько свойств), выбранных для определения сходства или различия между этими объектами.

*Классификация* требует соблюдения следующих правил:

* в каждом акте деления необходимо применять только одно основание;
* деление должно быть соразмерным, т.е. общий объем видовых понятий должен равняться объему делимого родового понятия;
* члены деления должны взаимно исключать друг друга, их объемы не должны перекрещиваться;
* деление должно быть последовательным.

Различают:

* вспомогательную (искусственную) *классификацию*, которая производится по внешнему признаку и служит для придания множеству предметов (процессов, явлений) нужного порядка;
* естественную *классификацию*, которая производится по существенным признакам, характеризующим внутреннюю общность предметов и явлений. Она является результатом и важным средством научного исследования, т.к. предполагает и закрепляет результаты изучения закономерностей классифицируемых объектов.

В зависимости от выбранных признаков, их сочетания и процедуры деления понятий *классификация* может быть:

* простой - деление родового понятия только по признаку и только один раз до раскрытия всех видов. Примером такой *классификации* является дихотомия, при которой членами деления бывают только два понятия, каждое из которых является противоречащим другому (т.е. соблюдается принцип: "А и не А");
* сложной - применяется для деления одного понятия по разным основаниям и синтеза таких простых делений в единое целое. Примером такой *классификации* является периодическая система химических элементов.

Под *классификацией* будем понимать отнесение объектов (наблюдений, событий) к одному из заранее известных классов.

*Классификация* - это закономерность, позволяющая делать вывод относительно определения характеристик конкретной группы. Таким образом, для проведения *классификации* должны присутствовать признаки, характеризующие группу, к которой принадлежит то или иное событие или объект (обычно при этом на основании анализа уже классифицированных событий формулируются некие правила).

*Классификация* относится к стратегии *обучения с учителем* (*supervised learning*), которое также именуют контролируемым или управляемым обучением.

Задачей *классификации* часто называют предсказание категориальной зависимой переменной (т.е. зависимой переменной, являющейся категорией) на основе выборки непрерывных и/или категориальных переменных.

Например, можно предсказать, кто из клиентов фирмы является потенциальным покупателем определенного товара, а кто - нет, кто воспользуется услугой фирмы, а кто - нет, и т.д. Этот тип задач относится к задачам ***бинарной классификации***, в них зависимая переменная может принимать только два значения (например, да или нет, 0 или 1).

Другой вариант *классификации* возникает, если зависимая переменная может принимать значения из некоторого множества предопределенных классов. Например, когда необходимо предсказать, какую марку автомобиля захочет купить клиент. В этих случаях рассматривается множество классов для зависимой переменной.

*Классификация* может быть ***одномерной*** (по одному признаку) и ***многомерной*** (по двум и более признакам).

***Многомерная классификация*** была разработана биологами при решении проблем дискриминации для классифицирования организмов. Одной из первых работ, посвященных этому направлению, считают работу Р. Фишера (1930 г.), в которой организмы разделялись на подвиды в зависимости от результатов измерений их физических параметров. Биология была и остается наиболее востребованной и удобной средой для разработки *многомерных* методов *классификации*.