**Лекция №7. Методы машинного обучения с учителем. Кластеризация. Метод k-средних алгоритма кластерного анализа**

Обучение без учителя (unsupervised learning) – один из разделов машинного обучения. Изучает широкий класс задач обработки данных, в которых известны только описания множества объектов (обучающей выборки), и требуется обнаружить внутренние взаимосвязи, зависимости, закономерности, существующие между объектами.

Нейросеть получает на входе неразмеченные данные и старается сама найти в них общие признаки и связи. То есть сначала нейросети скармливают очень много разных данных, а потом выпускают работать в «реальный мир», с настоящими задачами и все с такими же неразмеченными данными.

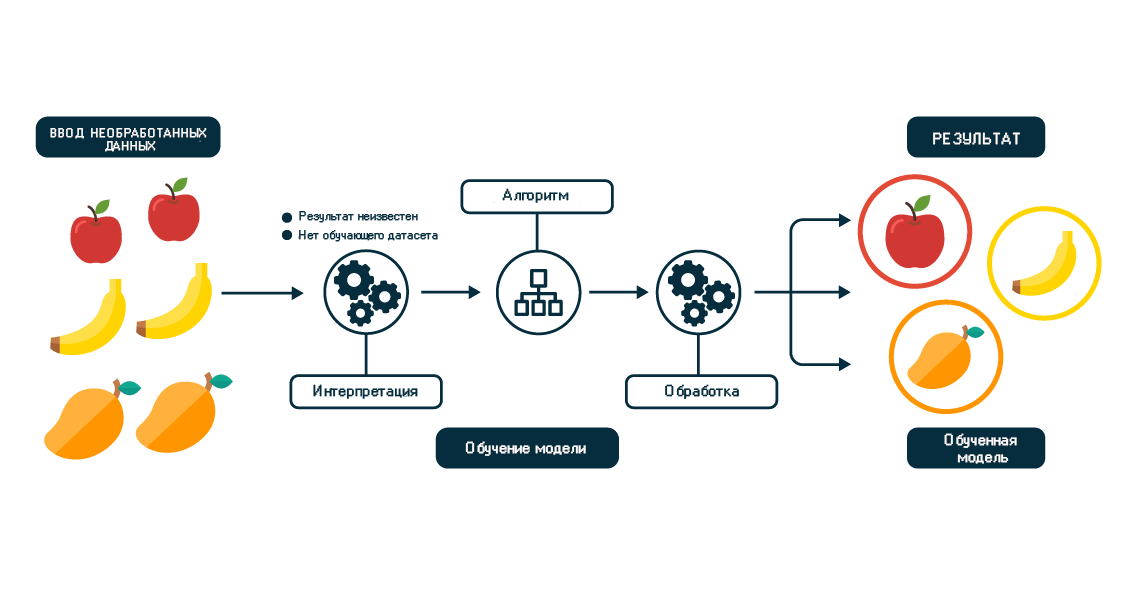


Рисунок 1 – Обучение без учителя

Иногда совсем без разметки нейросеть обучиться не способна, но много размеченных данных получить нельзя. Тогда используют частичное привлечение учителя – часть данных размечают, а часть передают в сыром виде. Это позволяет ускорить обучение и сделать его более точным. Например, так часто делают с медицинскими данными – передают рентгеновские снимки, на малом числе которых патология обозначена однозначно, а на других нейросеть учится ее распознавать.

Такую нейросеть можно использовать для следующих задач:

1. Проводить ***кластерный анализ***. Обученные без учителя нейронные сети лучше всего справляются с задачей ***кластеризации*** – делением большого массива данных на группы или кластеры таким образом, чтобы внутри группы наблюдения были более похожи друг на друга, чем на объекты другого кластера. При этом мы заранее не знаем на какие кластеры необходимо разбить наши данные. Это связано с тем, что мы обучаем модель на неразмеченных данных, то есть без целевой переменной, компонента y. Именно поэтому в данном случае говорят про машинное обучение без учителя. Нейросеть способна выделить критерии отличия и поделить данные на их основе. А потом с группами уже смогут работать люди, находя нужные тенденции и проводя аналитику.

**Кластеризация** – задача разделения объектов на группы, обладающие некоторыми свойствами. Примером может служить кластеризация документов из электронной библиотеки или кластеризация абонентов мобильного оператора.

***Цель кластеризации*** данных состоит в том, чтобы выделить группы примеров с похожими чертами и определить соответствие примеров и кластеров.

2. Выявлять аномалии. Для этого нейросеть сначала обучают на нормальных данных, и, когда в реальности она встречается с выбросами, то должна об этом сигнализировать. Так можно быстро обнаруживать необычное поведение покупателей, например резкий рост спроса на конкретный товар.

3. Находить ассоциации. Нейросеть может определить критерии похожести одних объектов на другие и строить связи. Например, рекомендовать товары в дополнение к купленным. Или показывать, что параметры сложной системы неожиданно связаны и влияют друг на друга.

***Как же разбить данные на кластеры?***

Векторы данных можно сравнивать между собой (оценивать их схожесть), измеряя расстояние между ними. Кластерный анализ использует именно этот подход. Мы измеряем расстояние между точками и на основе этого измерения принимаем решение к какому кластеру отнести то или иное наблюдение.

В рамках этой лекции мы поговорим про алгоритм, который называется методом k-средних.

***Метод k-средних***

Наиболее популярным алгоритмом кластеризации данных является метод k-средних. Это итеративный алгоритм кластеризации, основанный на минимизации суммарных квадратичных отклонений точек кластеров от центроидов (средних координат) этих кластеров.

Давайте пошагово разберемся в том, как работает этот алгоритм.

Шаг 1. Вначале возьмем данные и самостоятельно выберем желаемое количество кластеров и обозначим их буквой k (отсюда название метода). Пусть в данном случае их будет 3.

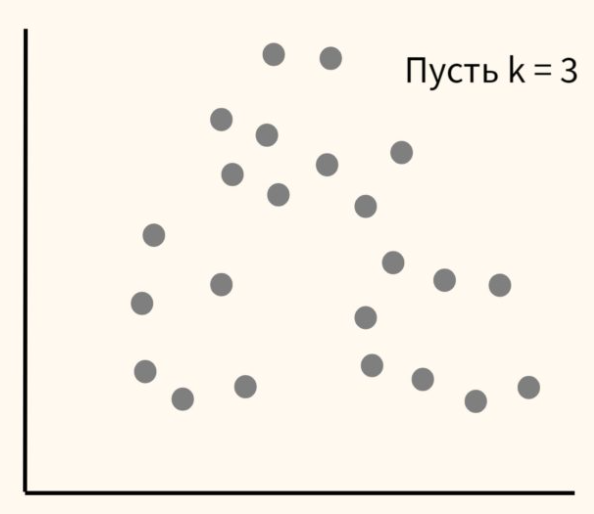


Рисунок 2 – Шаг 1

Шаг 2. Расположим несколько точек. Их количество будет равно количеству кластеров. Эти точки называются центроидами. Посчитаем расстояние от наших данных (точек) до каждого из центроидов. Логично отнести данные к тому центроиду, который находится ближе.

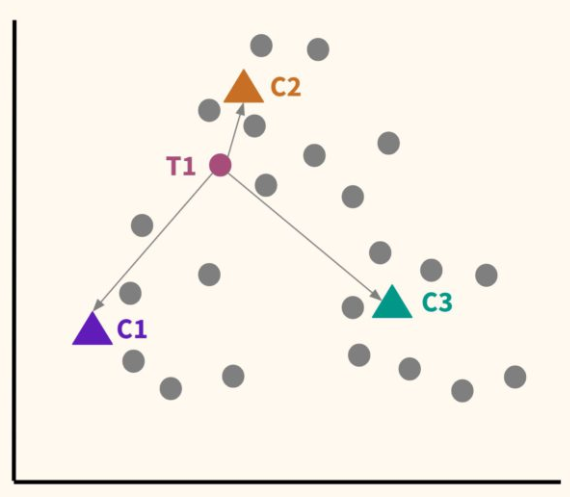


Рисунок 3 – Шаг 2

В частности, T1 будет отнесена к C2.

Шаг 3. Таким образом, каждая точка будет отнесена к определенному центроиду (кластеру).

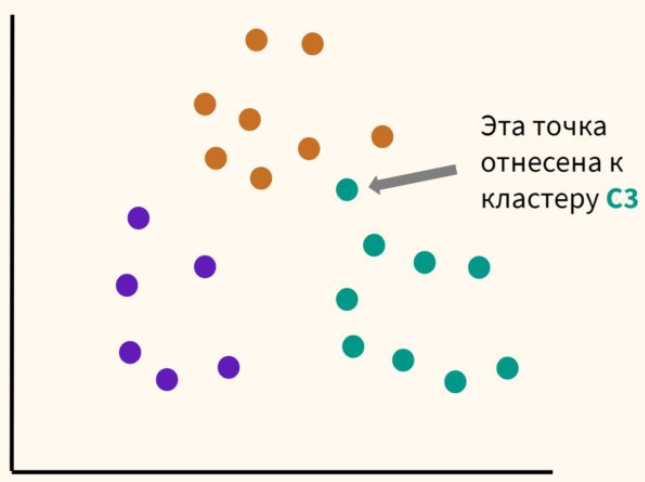


Рисунок 4 – Шаг 3

Шаг 4. Сместим наши центроиды в центр получившихся кластеров.

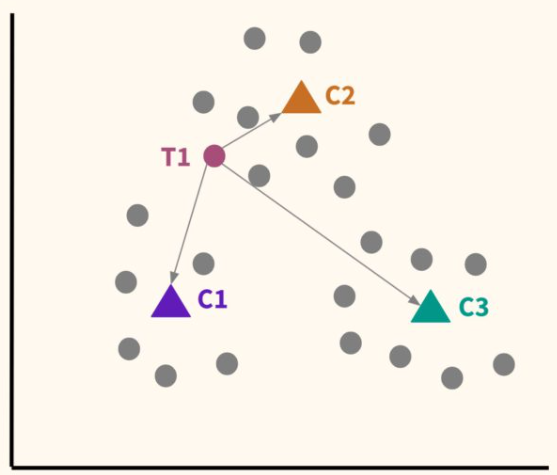


Рисунок 5 – Шаг 4

Шаг 5. Вновь посчитаем расстояние от наших данных (точек) до каждого из центроидов и отнесем их к каждому из центроидов. Некоторые наблюдения «переметнутся» к другому центроиду.

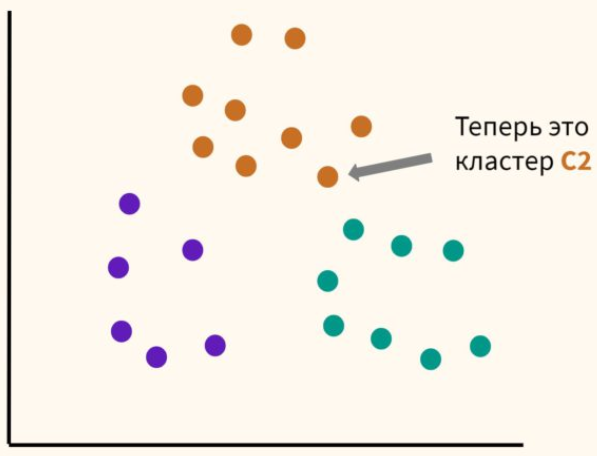
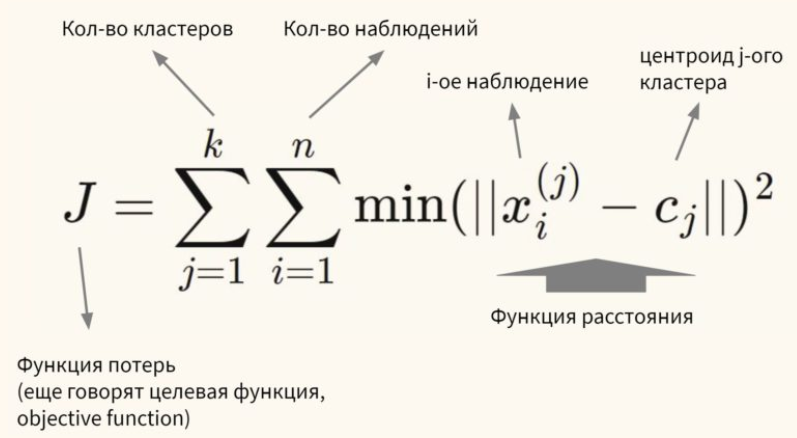


Рисунок 6 – Шаг 5

Мы будем повторять шаги 4 и 5 до тех пор, пока алгоритм не стабилизируется, то есть до тех пор, пока наблюдения не перестанут переходить от одного центроида (кластера) к другому.

Говоря более формально, цель алгоритма – минимизировать сумму квадратов внутрикластерных расстояний до центра кластера (сделать так, чтобы при выбранных центроидах расстояние от каждой из точек этого кластера до центра кластера было минимальным):



***Сколько кластеров выбрать?***

Есть два способа выбора количества кластеров:

1. Экспертный метод. Выбор количества кластеров будет зависеть от знаний о предметной области.

2. Метод локтя. Мы также можем (1) обучить модель используя несколько вариантов количества кластеров, (2) измерить сумму квадратов внутрикластерных расстояний и (3) выбрать тот вариант, при котором данное расстояние перестанет существенно уменьшаться.

На графике метод локтя выглядит следующим образом.

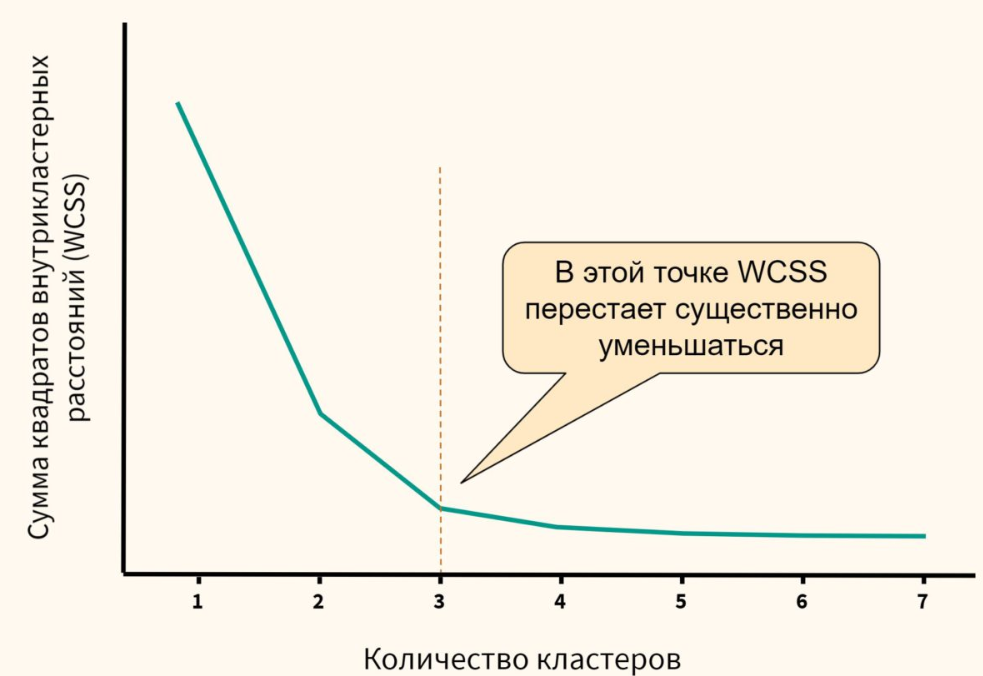


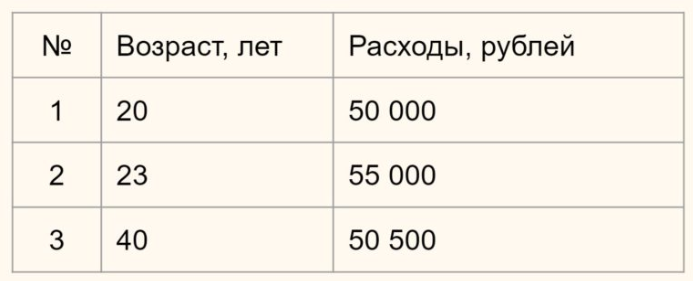
Рисунок 7 – Графике метода локтя

Как мы видим, после того как количество кластеров достигает 3, сумма квадратов внутрикластерных расстояний перестает существенно уменьшаться. Значит в данном случае 3 кластера и будет оптимальным значением.

***О важности нормализации данных***

Алгоритм очень чувствителен к масштабу признаков. В связи с этим нормализация данных приобретает особое значение. Так как при формировании кластеров мы измеряем Евклидово расстояние, то признаки с большим масштабом будут иметь больший вес.

Предположим, у нас есть данные о возрасте и ежемесячных тратах людей по кредитным картам.



Нам нужно определить, насколько человек 1 отличается (насколько велико расстояние) от человека 2 и 3. В зависимости от этого мы будем формировать наши кластеры.

Вначале давайте обратимся к здравому смыслу. Мы видим, что человек 1 и 2 схожи, потому что им обоим около 20-ти и расходы у них примерно одинаковы. Человек 3, имея схожие расходы, сильно отличается из-за своего возраста. Он почти в два раза старше. Это особенно легко увидеть на графике ниже:

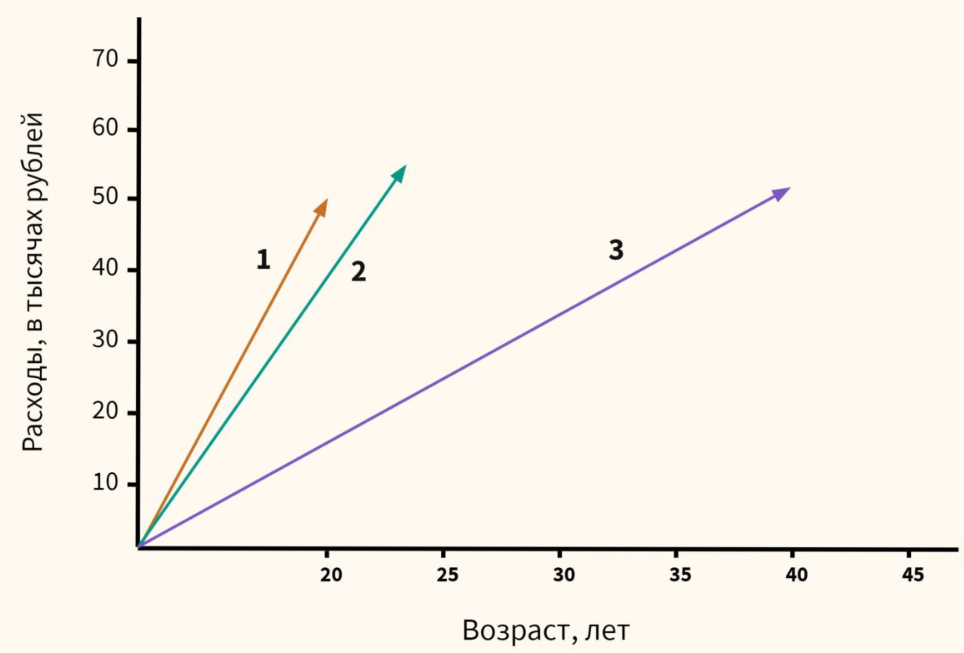
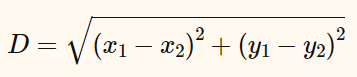


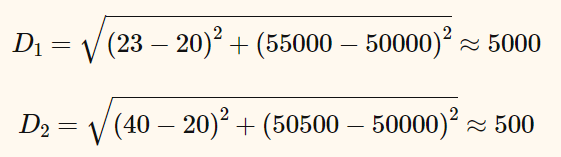
Рисунок 8 – График

Теперь посмотрим, что скажет математика, если мы оставим масштаб признаков без изменений.

Формула Евклидова расстояния:



В данном случае x1 и x2 – это возраст двух сравниваемых нами людей, а y1 и y2 – их расходы. Подставим значения в формулу:



D1 – это расстояние от человека 1 до человека 2, а D2 – от человека 1 до человека 3. Как мы видим, расстояние от человека 1 до человека 2 целых 5000 единиц (лет и рублей), в то время как до человека 3 только 500. Результат обратный тому, на который мы рассчитывали исходя из здравого смысла.

Это связано с тем, что масштаб второго признака (расходов) намного больше масштаба первого (возраст). Небольшое изменение в расходах вызывает существенное изменение расстояния, в то время как значительное изменение возраста не оказывает на него практически никакого влияния. Нам обязательно нужно будет это учитывать, когда мы будем обучать нашу модель.