

Короткое повторение

обучение без учителя

Мы не учим модель по правильным ответам.
Мы ищем структуру, похожесть и группы в самих данных.



Что мы уже знаем

три типа задач анализа данных

Регрессия

есть цель-числ: цена, риск, температура

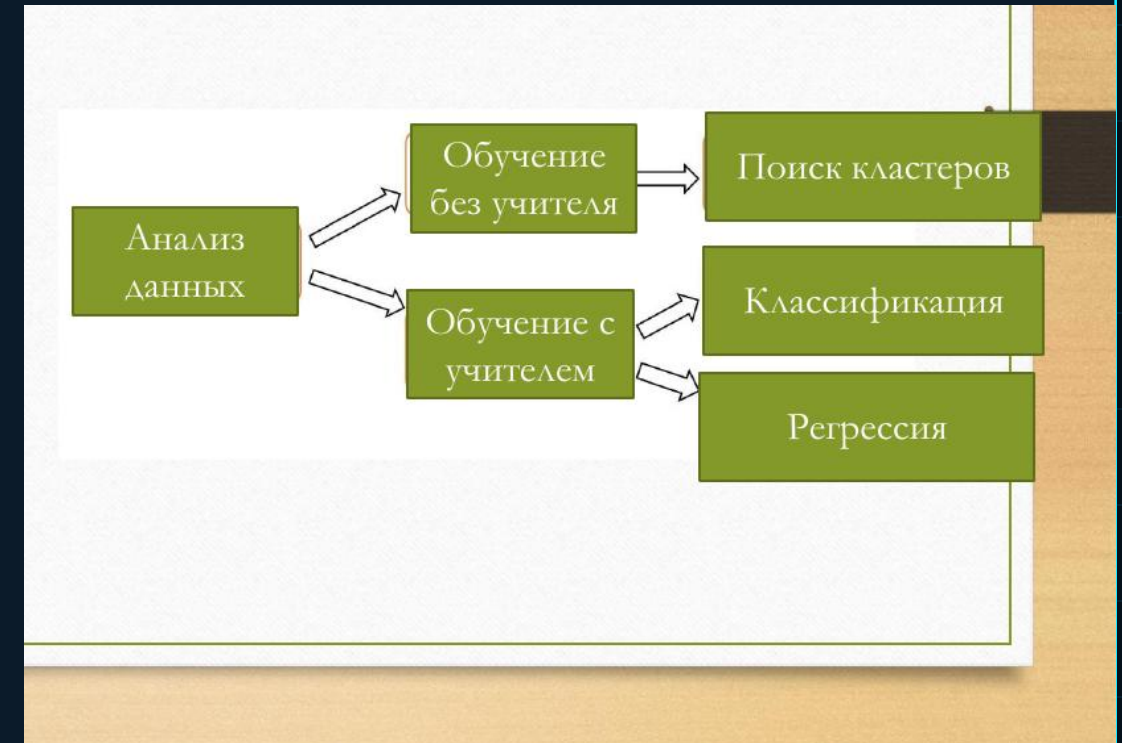
Классификация

есть цель-категория: класс, диагноз, тип

Без учителя

Целевой переменной нет:
ищем группы, структуру и
странные объекты

Сегодня нас интересует нижняя ветка: не предсказать метку, а обнаружить похожесть.



Главное отличие

цвета заранее не даны

С учителем

Дана целевая переменная
- правильные цвета

Без учителя

цвета исчезли: модель
видит только признаки

Задача меняется: теперь мы не проверяем готовые группы, а пытаемся их обнаружить.



Что мы ищем без учителя?

не ответ, а структуру

Группы

объекты внутри
группы похожи друг
на друга

Расстояния

надо договориться,
что значит «похожи»

Уровни

иногда группы бывают
крупными и мелкими

Выбросы

странные объекты
могут быть важнее
средних

Кластеризация — это способ навести порядок, когда никто не дал нам готовую раскладку.

Обучение без учителя похоже на уборку в комнате без инструкции: порядок можно навести, но сначала надо решить, что считать порядком.

ключевой вопрос: что считать похожестью?

Простой пример: монеты

группы можно искать даже без подписей

Мы можем не знать названия монет, но всё равно заметить группы по размеру, цвету или форме.

Важная мысль

кластеры зависят от того, какие признаки мы дали алгоритму

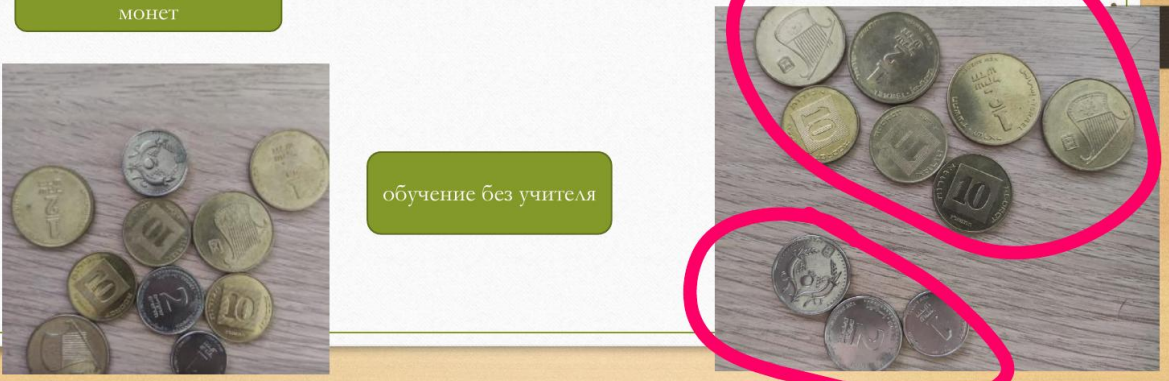
Если измерять только размер — получится одно. Если добавить цвет и материал — другое.

Монеты сами не говорят нам “раздели меня вот так”, и данные обычно тоже не говорят.

Обучение без учителя - пример

данные: изображения монет

обучение без учителя



Кластеры вокруг нас

изображения тоже можно группировать

На картинке нет учителя

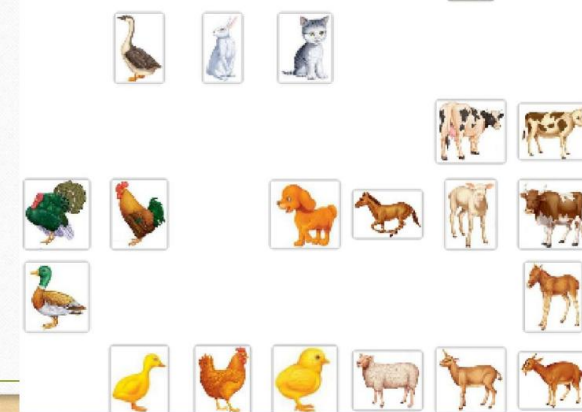
алгоритм ищет соседство объектов в пространстве признаков

Но человек всё равно нужен

после группировки надо понять, есть ли смысл у найденных групп

Кластеризация помогает увидеть порядок, но не гарантирует правильную интерпретацию.

Пример – кластеризация изображений



Что такое кластер?

короткое рабочее определение

Кластер — это группа объектов, которые внутри похожи сильнее, чем на объекты из других групп.

Внутри

маленькие расстояния

Между группами

большие расстояния

Почти все методы кластеризации по-разному реализуют эту простую идею.

Обучение без учителя

- **Цель:** «разделить объекты на группы (кластеры) так чтобы, объекты в одном кластере были сильно *похожи* друг на друга, а объекты из разных кластеров сильно *отличались*»

Сколько кластеров вы видите?

первый тест на неоднозначность

Даже на простой картинке люди не всегда сразу договорятся, сколько групп “правильно”.

Кто-то видит 2

крупные группы

Кто-то видит 4

если разделить
ближние облачка

вопрос к аудитории: где провести границу?

Упражнение 1

- Сколько кластеров вы видите?



Не все кластеры круглые

форма данных тоже важна

Упражнение 2

- Сколько кластеров вы видите?



Упражнение 3

- Сколько кластеров вы видите?



Иногда

видна вложенность

Иногда

видны длинные формы

И тогда

простые методы могут
ошибаться

Данные редко спрашивают алгоритм, удобно ли ему будет их делить.

Что показали упражнения?

главный вывод из простых картинок

2D легко

на плоскости мы ещё
можем спорить глазами

Многомерность

в текстах будут тысячи
признаков

Неоднозначность

даже красивая картинка не всегда даёт один честный
ответ

Со сказками Гримм глазами уже не справиться: нам
понадобится алгоритм.

- Было не очень сложно 😊
- Наши данные были двумерными – две независимые переменные 😊
- Не всегда ответ однозначен 😞
- Для многомерных данных визуально определить кластеры может быть очень сложно 😞

Зачем это вообще нужно?

кластеризация как инструмент понимания

Клиенты

разные группы —
разные
предложения

Пациенты

разные группы —
разная срочность

Тексты

НОВОСТИ, ОТЗЫВЫ,
сказки

Модели

для разных групп
можно строить разные
модели

Кластеризация не заменяет эксперта, но помогает эксперту быстрее увидеть карту данных.

Обучение без учителя — почему важно?

- Задача регрессии — можно разбить данные на кластеры и для каждого кластера построить свою модель
- Кластеризация упорядочивает данные и позволяет их лучше понять
- Разные кластеры больных в приемном отделении могут требовать лечения разной срочности
- Разные кластеры клиентов могут предпочитать разную рекламу
- Разные кластеры рисунков могут принадлежать художникам разных эпох

Ещё одна польза: выбросы

странный объект может быть главным сюжетом

Иногда самый интересный объект — не тот, кто хорошо попал в группу, а тот, кто торчит сбоку.

Ошибка?

возможно, мусор в данных

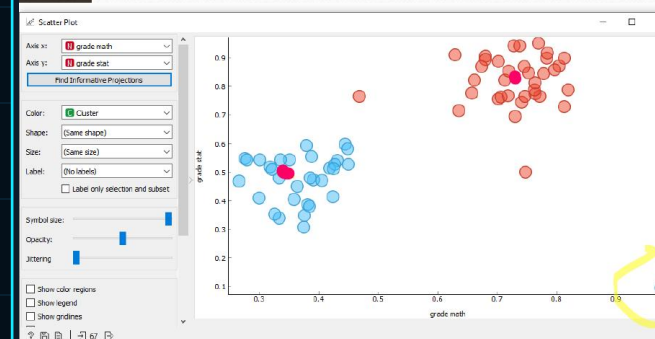
Находка?

возможно, важный необычный случай

Позже “странный” рассказ про волка станет именно таким моментом расследования.

Обучение без учителя — почему важно?

- Кластеризация может помочь обнаружить выбросы - outliers



Идея алгоритмов кластеризации

одна мысль, много реализаций

**Хотим, чтобы внутри группы
объекты были близко, а
разные группы были далеко.**

Внутри

минимизируем
расстояния

Между

стараяемся
разделить группы



Разные методы отличаются тем, как именно они воплощают эту идею.

В Orange есть несколько методов

нам нужны два главных героя урока

Hierarchical

строит дерево
отношений между
объектами

K-means

строит группы вокруг
центров

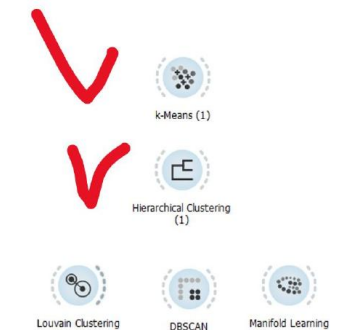
Главное

не один метод “лучше”, а разные методы отвечают на
разные вопросы

На уроке сравним и познакомимся с этими методами

Кластеризация в Orange

- Orange поддерживаем несколько алгоритмов кластеризации
- Мы изучим два самых популярных и простых алгоритма



Summary

часть 0: что нужно вспомнить перед Grimm

Нет target

модель не получает правильные ответы

Ищем структуру

группы, близость, выбросы

Нужна похожесть

без метрики нет кластеризации

Результат не истина

это гипотеза о данных

Теперь можно перейти к настоящей задаче: попробуем найти группы в сказках братьев Гримм.

Вопрос от преподавателя

проверка перед основной задачей

?

Если алгоритм нашёл три красивых кластера, значит ли это, что в данных действительно существуют ровно три настоящие группы?

Подумайте не про картинку, а про то, от чего зависит результат кластеризации.

Ответ

кластеры — это гипотеза, а не приговор



Нет. Не обязательно.

Мы можем сказать только так: при выбранных признаках, метрике, алгоритме и параметрах данные хорошо разделились на три группы.

Смысл этих групп должен проверить человек.

Переходим к сказкам Гримм

теперь повторение закончилось

У нас будет не игрушечный пример, а тексты сказок.

Объекты

сказки

Признаки

слова и частоты

Похожесть

cosine distance

Методы

дерево и
центры

Следующий блок начинается с вопроса: как превратить сказку в данные?

АНАЛИЗ ДАННЫХ В ВЕК ИИ

Урок 10 · блок 10.2

Дело братьев Гримм

готовим сказки к кластеризации

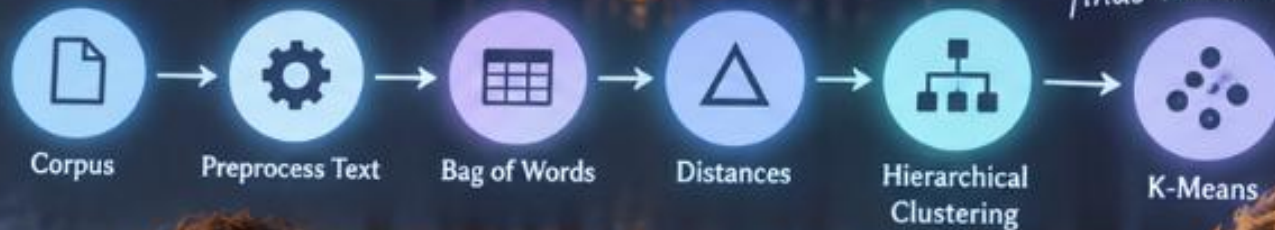
Сначала превращаем текст в числа. Только потом ищем группы.

Доктор Игорь Клейнер



GRIMM
TALES
+
DATA SCIENCE
=
NEW
INSIGHTS

From stories to clusters



Unsupervised learning
finds natural groups

Similar stories
similar characters
natural clusters
— no labels needed!

CHARACTERS
TEXT MINING
CLUSTERING
STORY WORLDS
MACHINE LEARNING
HUMAN STORIES

FAIRY TALES
TEACH
US TO SEE
PATTERNS
EVERYWHERE

PATTERNS
SIMILARITY
CLUSTERS
FAIRY TALES
REAL INSIGHTS



Royal & Noble



Forest & Wilderness



Home & Family



Magical Helpers

Different paths
Similar themes
Clusters reveal
the bigger picture.

Same data. Different stories. Meaningful clusters.

Better
Stories
with
Data

Наша задача

Целевая переменная в данных есть, но мы использовать ее не будем – ее как будто нет

**У нас есть сказки.
Групп нет. Ответов нет.
Ищем структуру сами.**

Не классификация. Не предсказание. Именно поиск похожести.

Что является объектом?

каждая строка будущей таблицы — одна сказка

Объект

одна сказка

например:

«Золушка»

«Гензель и Гретель»

«Волк и семеро козлят»

Данные не структурированы Сырые тексты

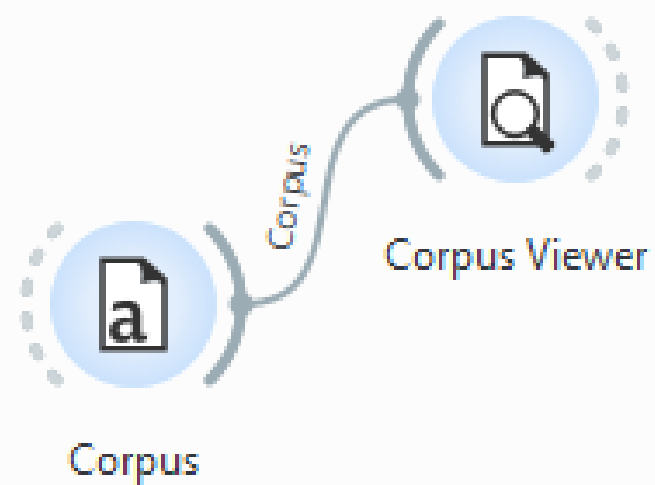
текст:

абзацы, слова, герои, повторы,
разные длины

Для нашего алгоритма кластеризации

нужны числа:

строки, столбцы, признаки,
расстояния



Corpus - Orange

File View Window Help

Corpus file

grimm-tales-selected.tab  Browse  Reload

Corpus settings

Title variable  Title

Language English

Used text features

 Content

Ignored text features

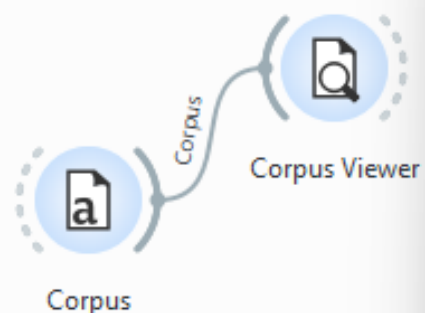
 Title

 Abstract

 ATU Numerical

Browse documentation corpora

   |  -  44



Corpus Viewer - Orange

File Edit View Window Help

Info

Tokens: n/a
Types: n/a
Matching documents: 44/44
Matches: n/a

Search features

Filter...

ATU Topic

Title

Abstract

Content

ATU Numerical

ATU Type

Display features

Filter...

ATU Topic

Title

Abstract

Content

ATU Numerical

ATU Type

☐ Show Tokens & Tags

☒ Auto send is on

RegExp Filter:

1 A Tale About the Boy Who Went ...

2 Brier Rose

3 Cat and Mouse in Partnership

4 Cinderella

5 Hansel and Gretel

6 Herr Korbes

7 Jorinda and Jorindel

8 Little Red Riding Hood

9 Mother Holle

10 Old Sultan

11 Pack of Scoundrels

12 Rapunzel

13 Rumpelstiltskin

14 Snow White

15 The Blue Light

16 The Bremen Town Musicians

17 The Crumbs on the Table

18 The Dog and the Sparrow

19 The Elves and the Shoemaker

ATU Topic: Tales of Magic

Title: Brier Rose

Abstract: An offended witch curses a young princess to die from pricking her finger on a spindle on her 15th birthday. A good witch blunts the curse by instead having her fall asleep for 100 years.

Content: A king and queen once upon a time reigned in a country a great way off, where there were in those days fairies. Now this king and queen had plenty of money, and plenty of fine clothes to wear, and plenty of good things to eat and drink, and a coach to ride out in every day: but though they had been married many years they had no children, and this grieved them very much indeed. But one day as the queen was walking by the side of the river, at the bottom of the garden, she saw a poor little fish, that had thrown itself out of the water, and lay gasping and nearly dead on the bank. Then the queen took pity on the little fish, and threw it back again into the river; and before it swam away it lifted its head out of the water and said, 'I know what your wish is, and it shall be fulfilled, in return for your kindness to me-- you will soon have a daughter.' What the little fish had foretold soon came to pass; and the queen had a little girl, so very beautiful that the king could not cease looking on it for joy, and said he would hold a great feast and make merry, and show the child to all the land. So he asked his kinsmen, and nobles, and friends, and neighbours. But the queen said, 'I will have the fairies also, that they might be kind and good to our little daughter.' Now there were thirteen fairies in the kingdom; but as the king and queen had only twelve golden dishes for them to eat out of, they were forced to leave one of the fairies without asking her. So twelve fairies came, each with a high red cap on her head, and red shoes with high heels on her feet, and a long white wand in her hand: and after the feast was over they gathered round in a ring and gave all their best gifts to

≡ ? 📄 | ↩ 44 ↪ 1 | 43 | 44

Компьютер не читает как человек

он не видит сюжет напрямую

Человек видит историю.
Компьютер видит последовательность символов.
Можно добавить дополнительные свойства текстов

Человек видит сюжет

волк, лес, страх, мораль

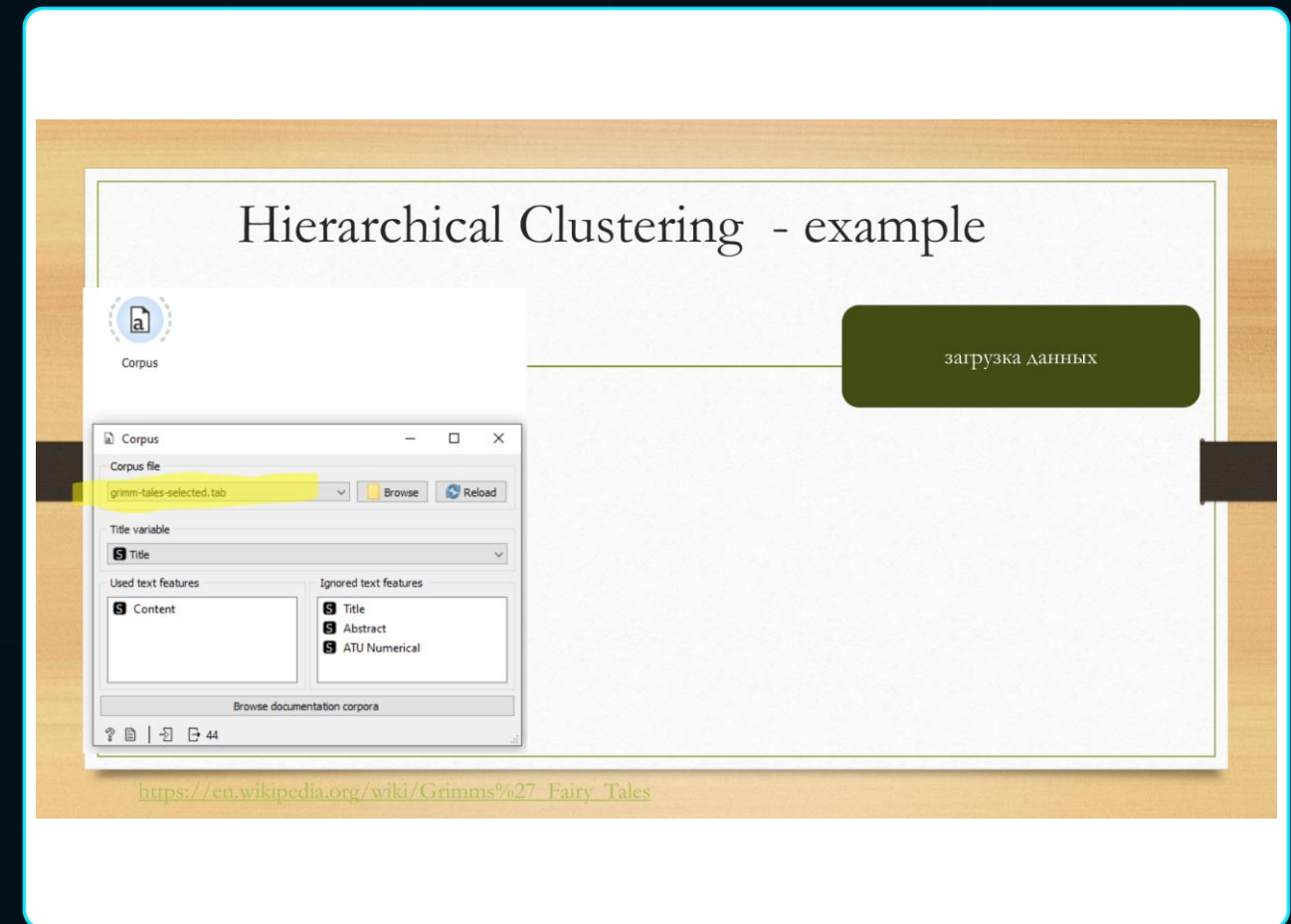
Алгоритм

слова, частоты, векторы, расстояния

Загрузка корпуса

в Orange начинаем с Corpus

- берём готовый корпус сказок
- одна запись = одна сказка
- сначала просто смотрим данные
- не нажимаем кластеризацию вслепую



старый маршрут: Corpus → загрузка сказок

Сначала смотрим внутрь

Corpus Viewer — не роскошь, а проверка данных

- какие есть сказки?
- есть ли title и content?
- нет ли странных пустых строк?
- понимаем, что именно кластеризуем

Hierarchical Clustering - example

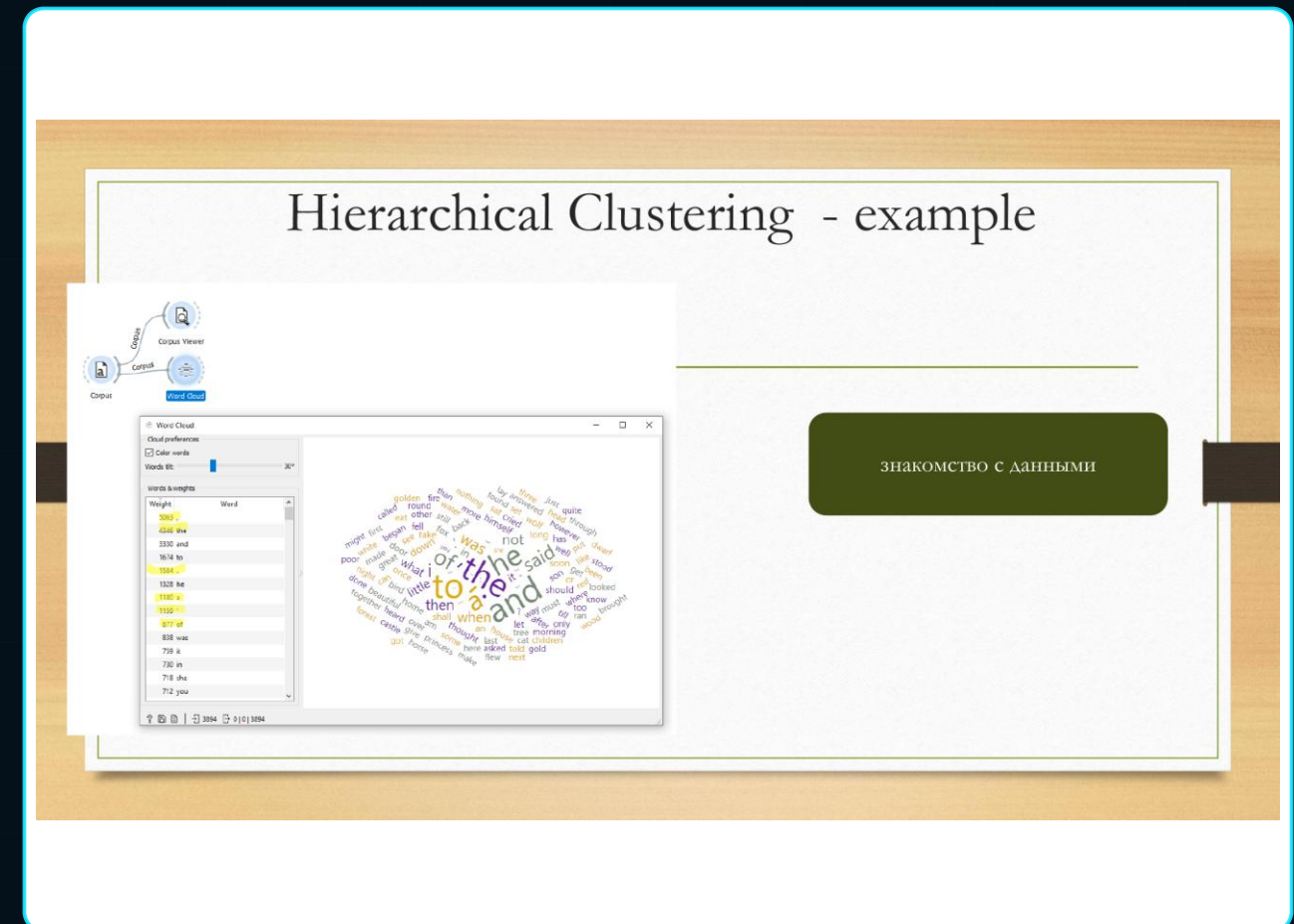
ЗНАКОМСТВО С ДАННЫМИ

просмотр названий и текстов

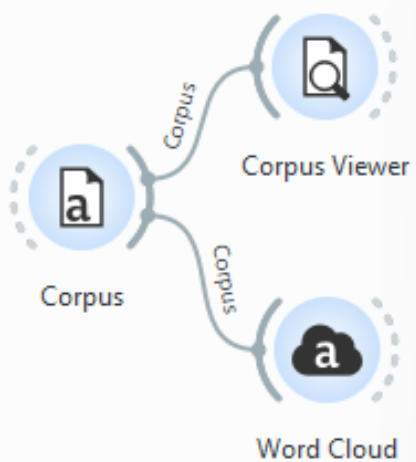
Облако слов: быстрая разведка

частое слово ещё не всегда полезное

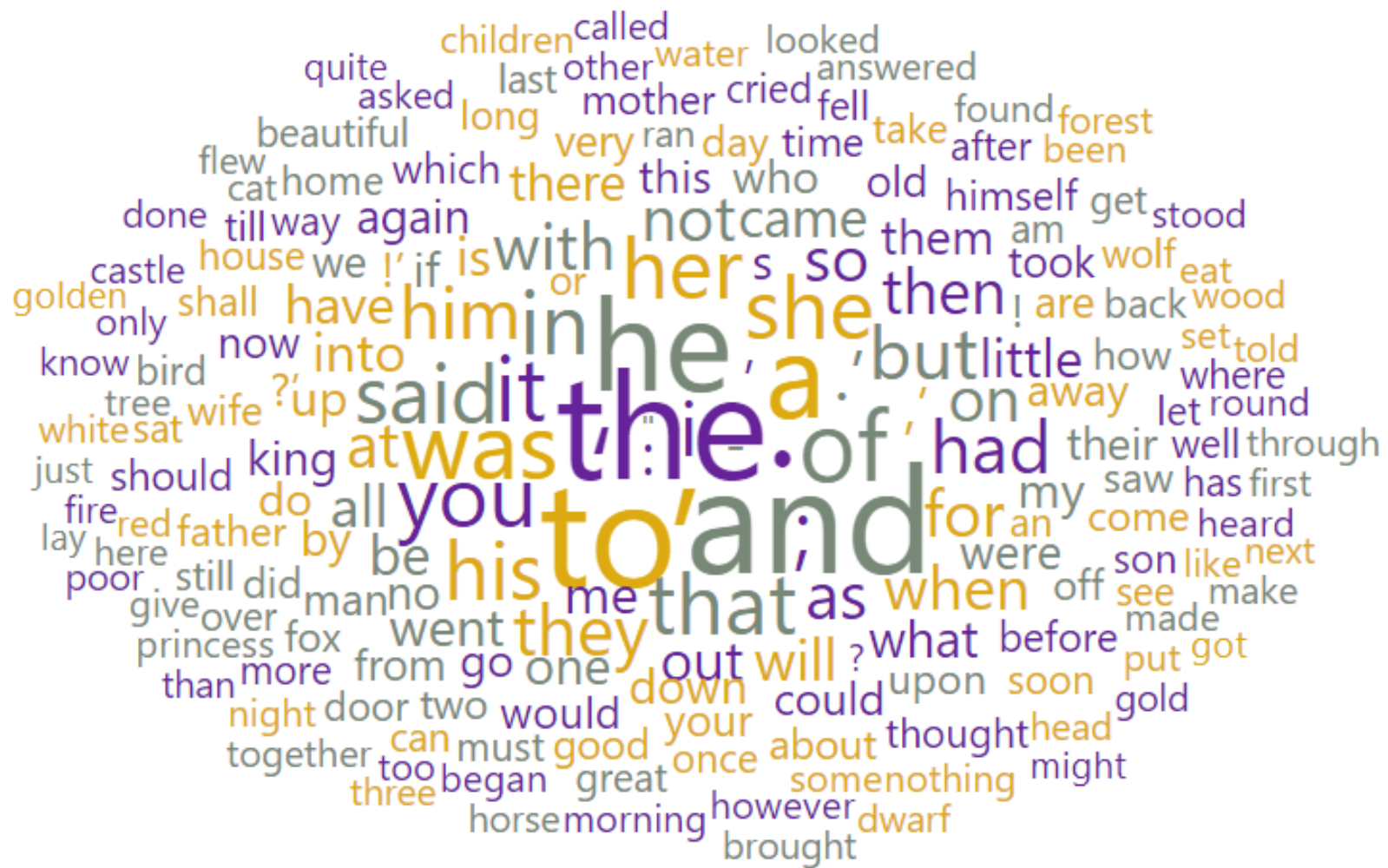
- видим словарь корпуса
- замечаем слишком общие слова
- понимаем, что текст надо
ЧИСТИТЬ
- облако слов — разведка, не
ВЫВОД



Word Cloud показывает самые частые слова



44 | 0 | 3894



Подводный камень: частые слова

не все частые слова помогают отличать сказки

Полезно

wolf
king
forest
magic
stepmother

Сомнительно

would
could
said
then
one

Опасность

частота есть,
смысла для группировки
может не быть

Самое частое слово иногда сообщает только то, что автор любит это слово.

Чистим текст

Preprocess Text — уменьшаем шум

- приводим слова к удобному виду
- удаляем часть стоп-слов
- можем нормализовать формы слов
- не превращаем чистку в религию

Hierarchical Clustering - example

ЗНАКОМСТВО С ДАННЫМИ

Word Cloud

Words & weights

Weight	Word
188	and
201	came
271	into
271	went
241	saw
195	long
172	it
171	time
163	could
156	would
154	man
138	old
154	took
139	upon

Preprocess Text

Preprocessors

Transformation

- ☒ Lowercase
- ☐ Remove accents
- ☐ Parse html
- ☐ Remove urls

Tokenization

- ☐ Word Punctuation
- ☐ Whitespace
- ☐ Sentence
- ☒ Regexp
- ☐ Tweet

Filtering

- ☒ Stopwords
- ☐ Lexicon
- ☐ Regexp
- ☐ Document frequency
- ☐ Most frequent tokens

пример настроек preprocessing текста

Больше чистки ≠ лучше

можно удалить не только шум

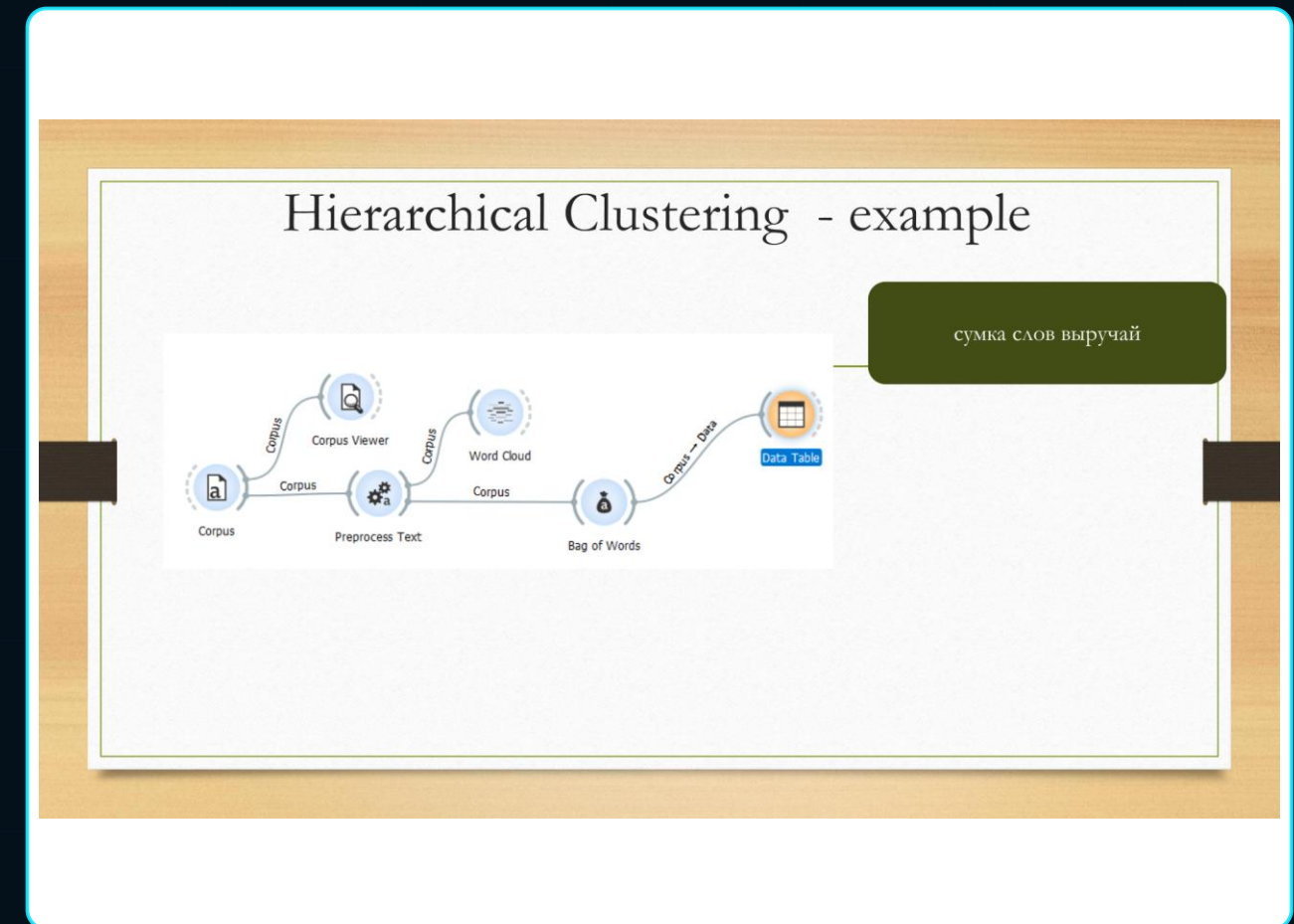
**Если вы убрали слишком много слов,
алгоритму уже нечем различать сказки.**

Предобработка — настройка микроскопа, а не волшебная уборка.

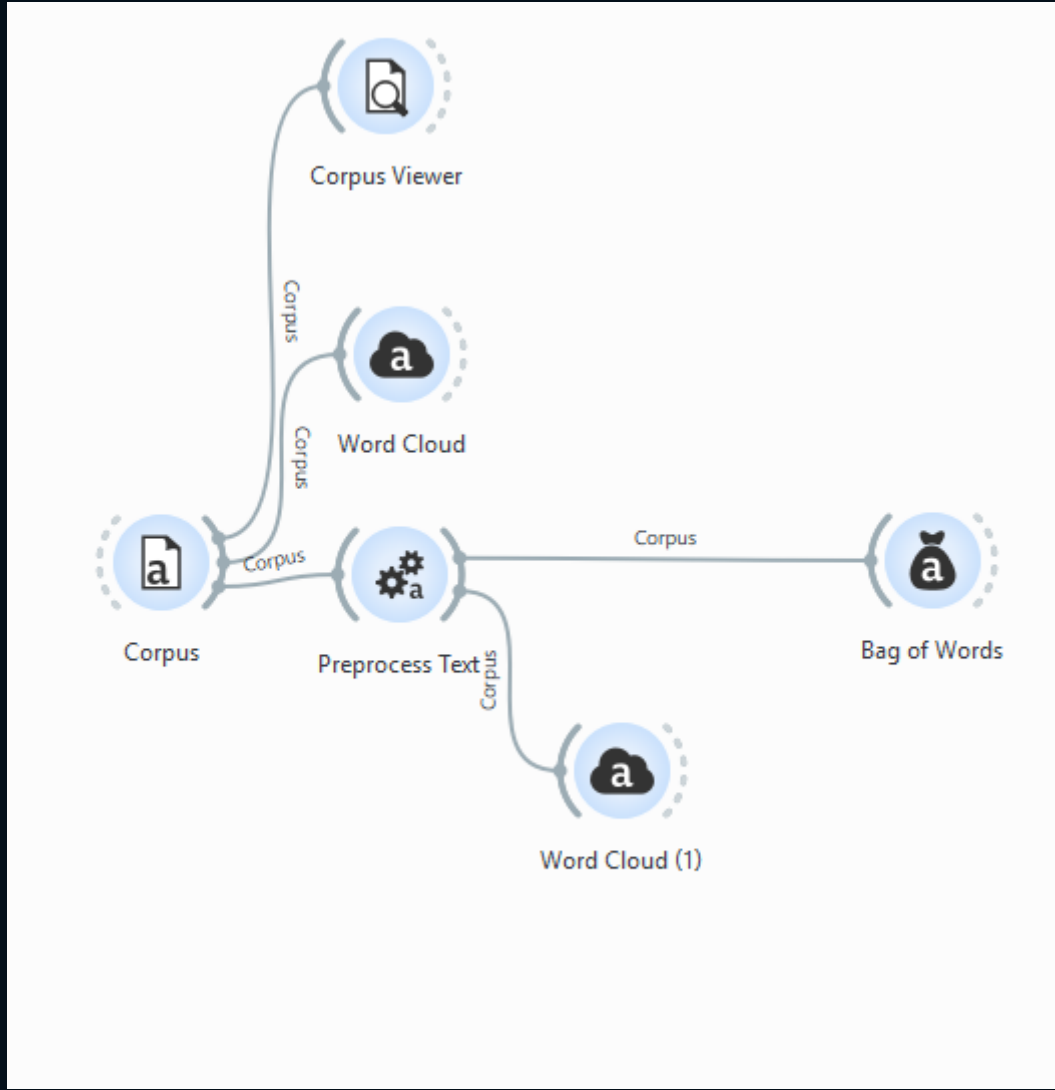
Сумка слов возвращается

текст превращается в таблицу чисел

- каждая сказка станет строкой
- каждое слово — признаком
- значение показывает частоту
- теперь можно считать расстояния



Bag of Words → Data Table



Title	Abstract	Content	ATU Numerical	ATU Type	{...}
True		True			
A Tale About th...	A simple boy w...	A certain father...	326.0	Supernatural A...	able=1, accord=1, actually=1, afraid=1, ago=1, air=1, ...
Brier Rose	An offended wi...	A king and que...	410.0	Supernatural or...	ale=3, alone=1, also=2, altogether=1, amiss=1, angry=...
Cat and Mouse ...	A mouse lives ...	A certain cat ha...	15.0	Wild Animals	absence=1, acquaintance=1, advice=1, agreed=1, alas...
Cinderella	The familiar sto...	The wife of a ri...	510A	Supernatural H...	_my_=1, afterwards=1, almost=1, also=1, altogether=1...
Hansel and Gretel	A poor woodcu...	Hard by a great...	327A	Supernatural A...	able=1, across=3, add=1, afar=1, afterwards=2, alighte...
Herr Korbes	A hen and a ro...	Once upon a ti...	210.0	Domestic Anim...	aboard=2, afterward=1, along=1, answered=2, arrived...
Jorinda and Jori...	A witch lures y...	There was once...	405.0	Supernatural or...	_jug=1, alas=2, almost=1, alone=1, already=1, always...
Little Red Ridin...	A girl known fo...	Once upon a ti...	333.0	Supernatural A...	able=1, act=1, afraid=1, afterwards=1, aged=1, alive=1...
Mother Holle	A widow spoils ...	Once upon a ti...	480.0	Supernatural Ta...	according=1, afraid=2, ago=2, agreed=1, alas=2, altho...
Old Sultan	A farmer decid...	A shepherd had...	101.0	Wild Animal an...	accordingly=1, advice=1, afterwards=1, air=2, along=...
Pack of Scound...	A rooster and a ...	The rooster sai...	210.0	Domestic Anim...	able=2, accept=1, across=2, ado=1, agreement=1, alre...
Rapunzel	The classic stor...	There were onc...	310.0	Supernatural A...	afraid=1, afterwards=1, agreed=1, aha=1, ails=1, alarm...
Rumpelstiltskin	A miller's daug...	By the side of a ...	500.0	Supernatural H...	alas=1, alone=2, among=1, arms=1, asked=2, astonish...
Snow White	The classic stor...	There was once...	426.0	Supernatural or...	account=1, accused=1, acquaintance=1, across=2, ad...
The Blue Light	A wounded sol...	There was once...	562.0	Supernatural H...	advice=1, aha=1, allowed=1, alone=1, already=1, anot...
The Bremen To...	A donkey, a do...	An honest farm...	130.0	Wild Animal an...	abode=1, according=1, added=1, afar=2, afterwards=...
The Crumbs on...	A man tells his ...	One day the ro...	236.0	Other Animals ...	anything=2, beat=1, beating=1, began=1, breadcrum...
The Dog and th...	A merchant run...	A shepherd's d...	248.0	Other Animals ...	aim=1, aimed=1, alas=3, alighted=1, almost=1, angril...
The Elves and t...	A poor shoema...	There was once...	503.0	Supernatural H...	always=1, amidst=1, asleep=1, away=3, backs=1, barg...
The Fisherman ...	A fisherman cat...	There was once...	555.0	Supernatural H...	alas=3, along=2, already=5, angry=2, anything=2, aros...
The Fox and th...	The fox is extre...	It happened tha...	105.0	Wild Animal an...	able=1, already=1, answer=1, arrogance=1, art=1, arts...
The Fox and th...	A hungry fox h...	The fox once ca...	227.0	Other Animals ...	allow=1, also=1, always=1, away=1, beautifully=1, beg...
The Fox and th...	A farmer will o...	A farmer had a ...	47A	Wild Animals	able=1, adrift=2, advice=1, avarice=1, away=1, back=2...
The Frog Prince	A princess loses...	One fine evenin...	440.0	Supernatural or...	abide=1, able=1, afterwards=1, air=1, alas=2, along=1,...
The Golden Bird	A young prince...	A certain king h...	550.0	Supernatural H...	account=1, advice=3, afraid=1, agreed=4, air=1, almo...
The Golden Go...	Unlike his cleve...	There was a ma...	511.0	Supernatural H...	able=1, across=1, afterwards=2, alone=1, also=2, answ...

Как понять, что две сказки похожи?

нужна мера расстояния

Сказка А

wolf: 7
king: 1
forest: 5

Сказка В

wolf: 3
king: 0
forest: 2

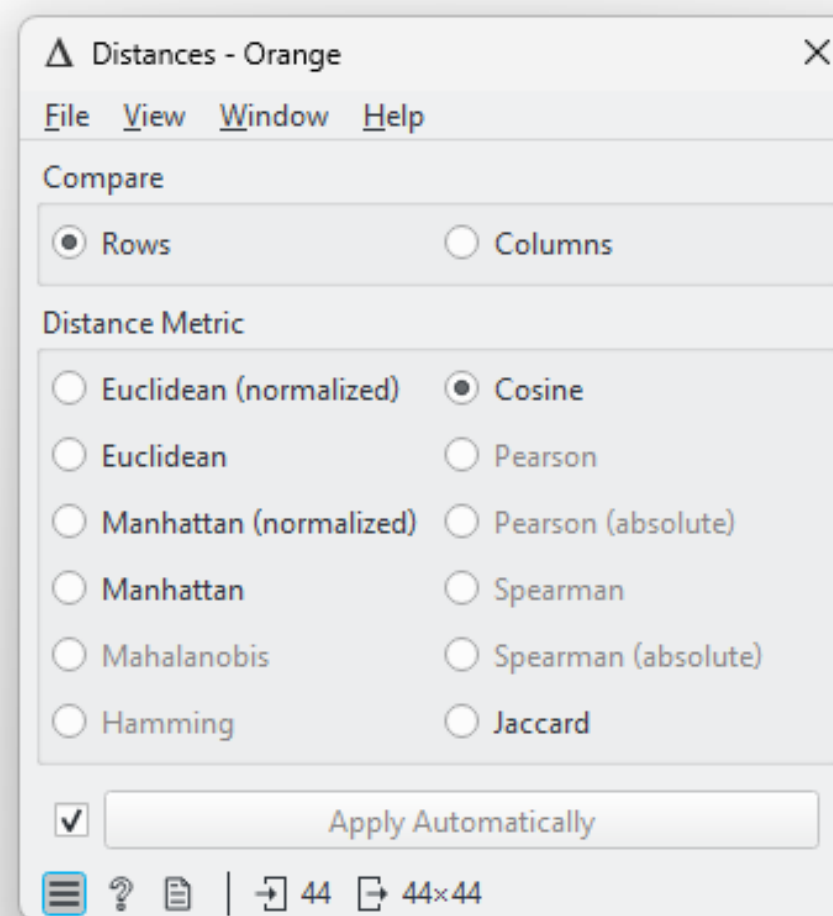
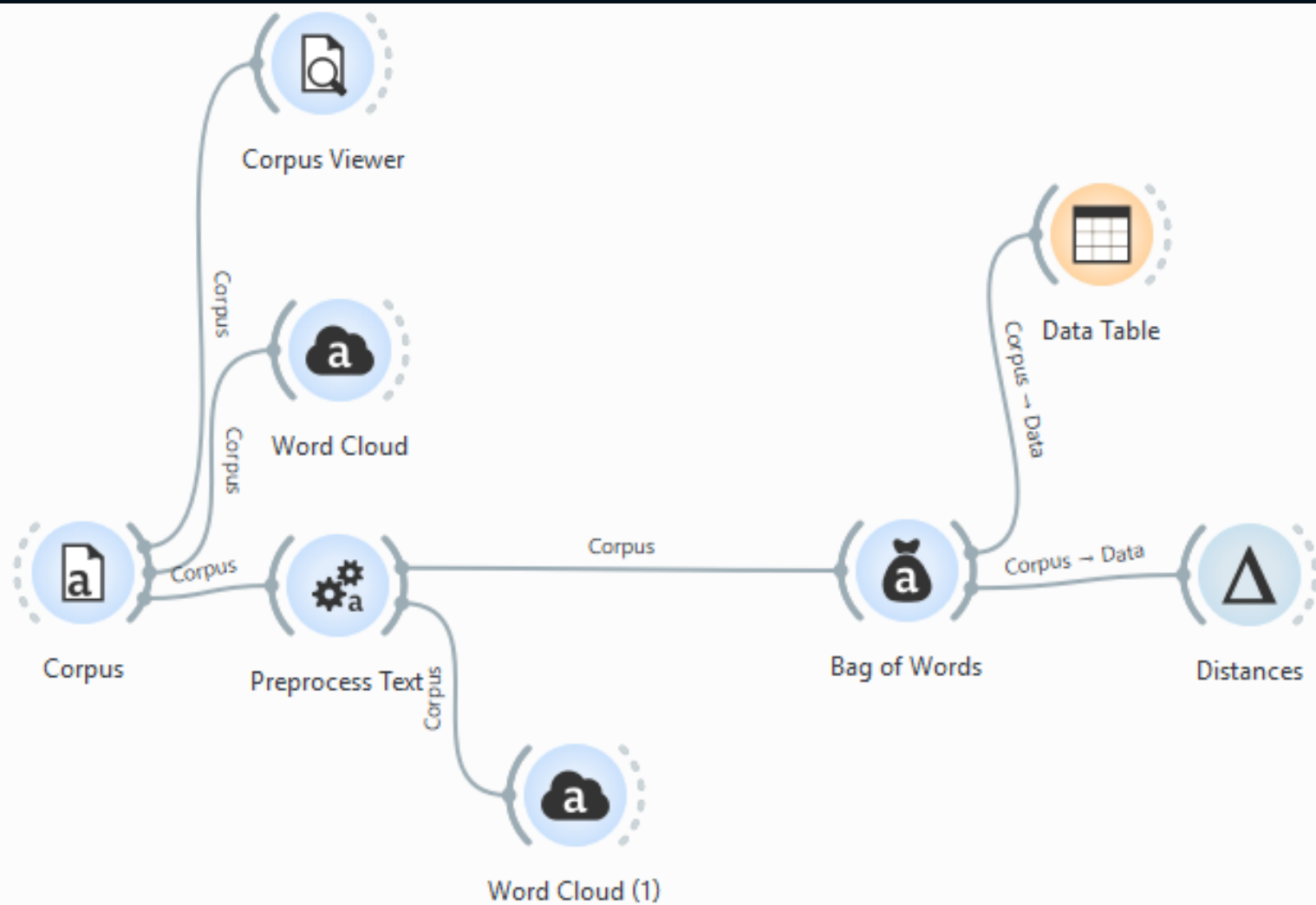
Сказка С

princess: 6
magic: 4
castle: 3

?

Что значит “близко”,
если признаков
тысячи?

Нужна метрика: правило, которое превращает
две строки таблицы в число расстояния.



Почему cosine distance?

для текста важен профиль слов, а не только длина

похожие направления

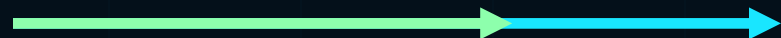
Сказка короткая

wolf: 3
forest: 2
king: 1

Сказка длинная

wolf: 30
forest: 20
king: 10

угол



Профиль похож.
Длина разная.

Distances → Cosine

последний шаг подготовки

- данные уже числовые
- считаем расстояния между всеми парами
- для текстов берём cosine
- теперь можно строить дерево

Hierarchical Clustering - example

данные готовы, можно приступать к кластеризации

почему метрика расстояний — косинус?

https://en.wikipedia.org/wiki/Grimms%27_Fairy_Tales

выбираем расстояние между сказками

Summary

10.2: Grimm — подготовка данных

Corpus

каждая запись — сказка

Preprocess

уменьшаем шум

Bag of Words

текст → числа

Data Table

строки и признаки

Cosine distance

сравниваем профиль слов

Готово

можно строить дерево

Алгоритм видит не сказки, а то представление, которое мы построили.

Вопрос

10.2 · проверка понимания

?

**Две сказки имеют почти одинаковый сюжет,
но написаны разными словами.
Bag of Words + cosine обязательно сочтут их похожими?**

Подумайте: что именно сравнивает наша текущая таблица?

Ответ

10.2 · ответ



Нет. Не обязательно.

Мы сравниваем не “истинный сюжет”, а числовое представление текста.

Если представление построено по словам, алгоритм ищет похожесть словарного профиля.



Many Tales
One Human Story

Stories
Motifs
Families
Connections

COLLECT
COMPARE
CLUSTER
UNDERSTAND

GRIMM

MÄRCHEN

VOL. I

VOL. II

Different Paths
Same Wonders

Иерархическая кластеризация

выращиваем дерево сказок

Дендрограмма показывает не только группы, но и историю их объединения.

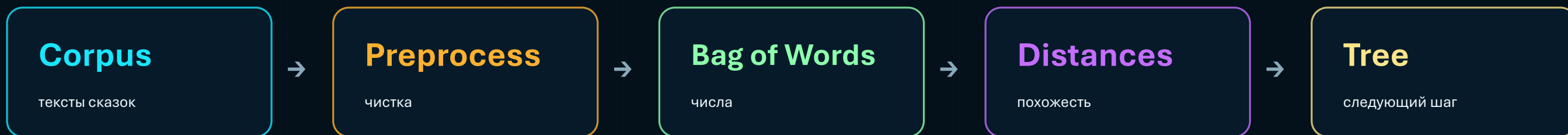
Доктор Игорь Клейнер



Где мы остановились

расстояния уже посчитаны

Corpus → Preprocess → Bag of Words → Distances → ?



Теперь из расстояний будем строить дерево отношений.

Идея: каждая сказка сначала сама по себе

в начале нет групп

Золушка

отдельный кластер

Рапунцель

отдельный кластер

Волк

отдельный кластер

Гензель

отдельный кластер

Король

отдельный кластер

Лиса

отдельный кластер

Гусь

отдельный кластер

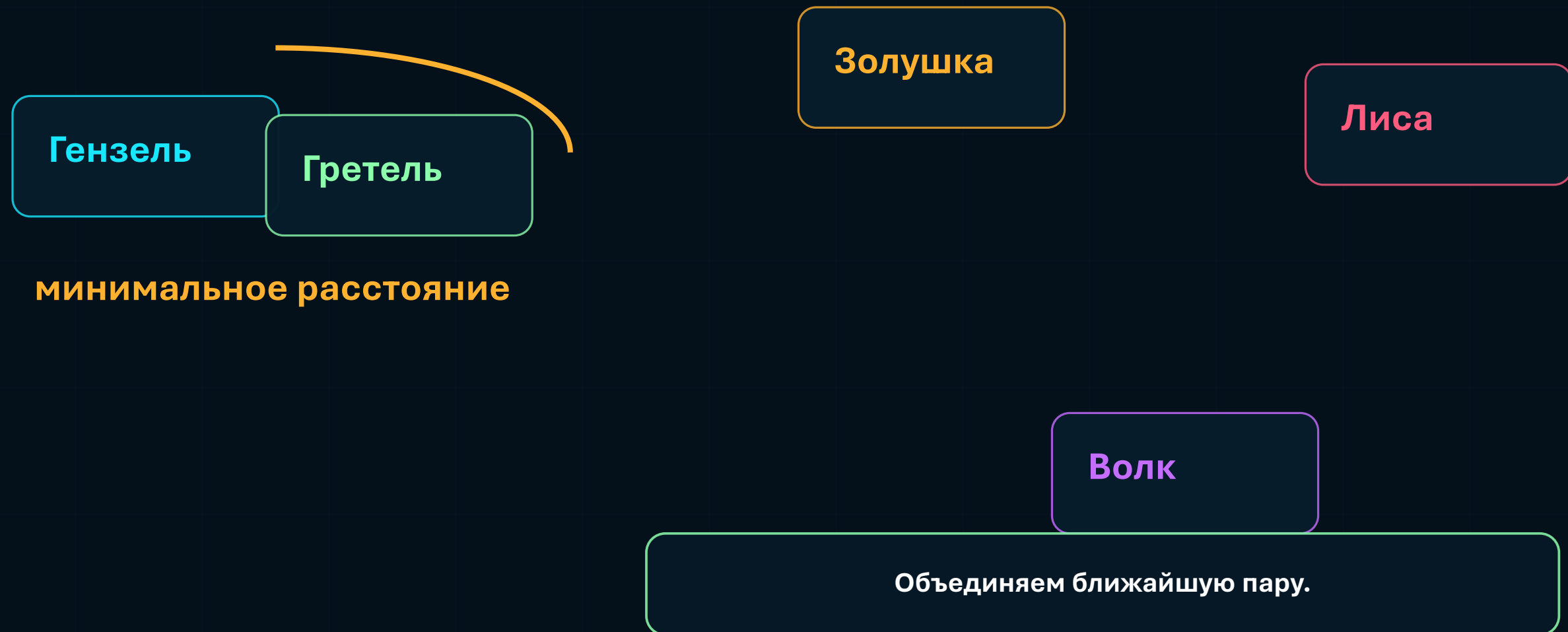
Мельник

отдельный кластер

Старт: 44 сказки → 44 маленьких кластеров

Шаг 1: нашли две ближайшие сказки

сначала объединяются самые похожие



Шаг 2: появился первый кластер

теперь у нас есть и объекты, и группа

Кластер 1

Гензель + Гретель

Золушка

Лиса

Волк

Теперь вопрос сложнее:

Как измерить расстояние
от сказки до группы?

Linkage: как сравнивать группы?

иерархический кластеринг — это семья методов

Single

смотрим на
самые близкие
точки

Complete

смотрим на
самые дальние
точки

Average

берём среднее
расстояние
между группами

Ward

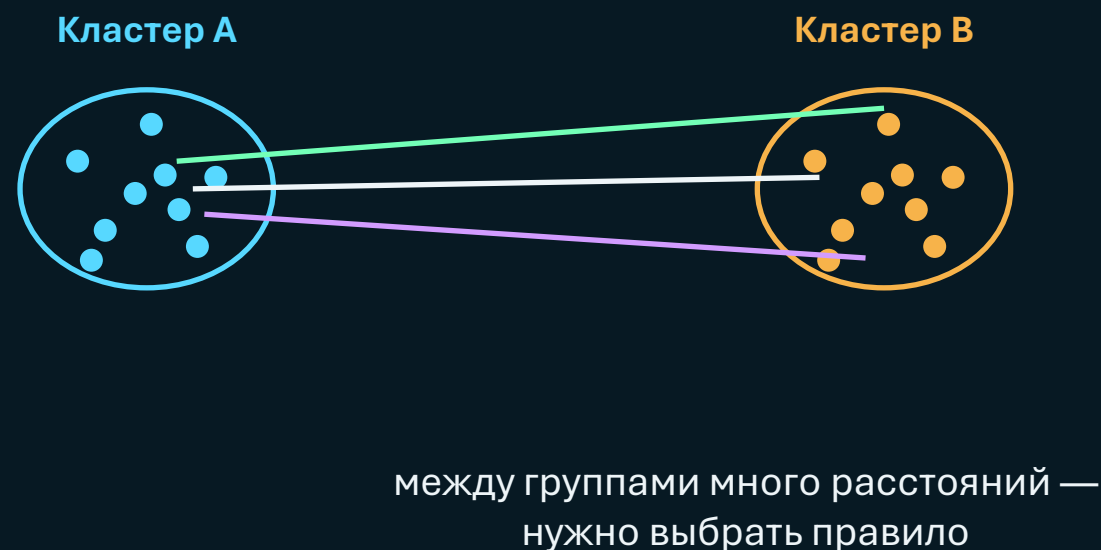
минимизируем рост
разброса внутри
групп

Главная мысль

поменяли linkage — дерево может измениться

Что такое linkage и зачем он нужен?

После первых объединений нужно измерять расстояние уже не между объектами, а между группами.



Ключевая идея

Linkage — это правило, которое говорит, как считать расстояние между двумя уже появившимися кластерами.

Почему это важно

На одних и тех же данных разные linkage могут построить разные дендрограммы.

Что запомнить

Метрика отвечает за похожесть объектов, linkage — за то, как дальше склеиваются группы.

Без linkage дерево не знает, как жить после первого объединения.

Какие linkage стоит знать в Orange?

В нашем маршруте важно упомянуть все основные варианты, но глубоко разбирать будем не каждый.

Single

ближайшая пара

может тянуть цепочки

Complete

самая дальняя пара

строже держит группы
отдельно

Average

среднее по всем
парам

часто спокойный
компромисс

Weighted

среднее между
группами

после объединения
группы получают
равный вес

Ward

минимальный рост
разброса

часто даёт
компактные кластеры

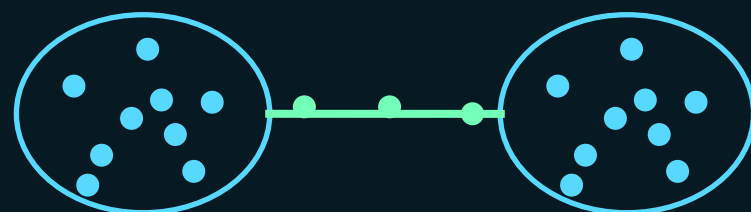
Исправление к прошлому варианту

Да, Weighted тоже нужно назвать: он есть среди вариантов linkage в Orange и не должен исчезнуть из карты методов.

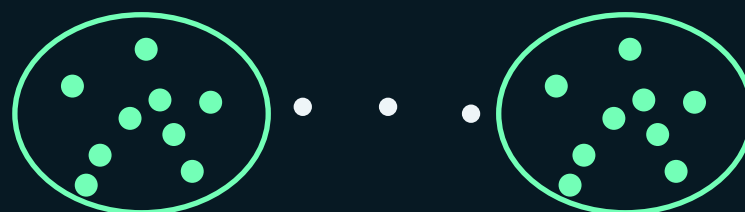
Сейчас нам важна карта вариантов; подробный смысл покажем на single linkage и Ward.

Почему выбор linkage меняет дерево?

Разная логика склеивания по-разному реагирует на мостики, выбросы и форму групп.



логика “соседей”
мостик быстро склеивает группы



логика “компактности”
группы дольше остаются отдельно

Данные те же

Меняется только правило, по которому дерево решает, какие группы считать близкими.

Главный вывод

Дендрограмма — не фотография природы, а результат выбранной логики объединения.

Что сейчас разберём

Сначала — почему single linkage может тянуть длинные цепочки.

Если меняем linkage, мы меняем историю объединений.

Почему single linkage тянет цепочки?

Он смотрит только на самую близкую пару точек между двумя группами.



Single linkage смотрит только на самую близкую пару.
Одна цепочка близких точек может шаг за шагом склеить длинную структуру.

Что происходит

Мостик из близких точек заставляет алгоритм склеивать группы звено за звеном.

Почему это может мешать

Вместо двух естественных кластеров появляется одна вытянутая “гусеница”.

Название эффекта

chaining effect — эффект цепочки.

Одной близкой пары иногда достаточно, чтобы начать длинное склеивание.

Complete, Average, Weighted: три спокойные альтернативы

Их можно понимать как разные способы не попасть в ловушку одной ближайшей пары.

Complete linkage

Смотрит на самую дальнюю пару между группами. Плюс: строже не даёт склеивать растянутые группы. Минус: может быть слишком осторожным.

Complete

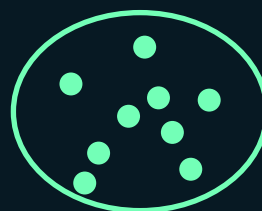


строгость

Average linkage

Смотрит на среднее расстояние по парам. Плюс: часто хороший компромисс. Минус: может сгладить важные крайние случаи.

Average

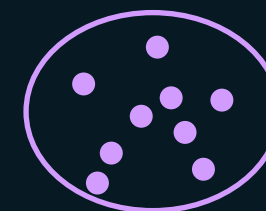


компромисс

Weighted linkage

Похож на average, но после объединения группы учитываются более равноправно. Плюс: меньше доминирования большой группы. Минус: менее интуитивен для новичка.

Weighted

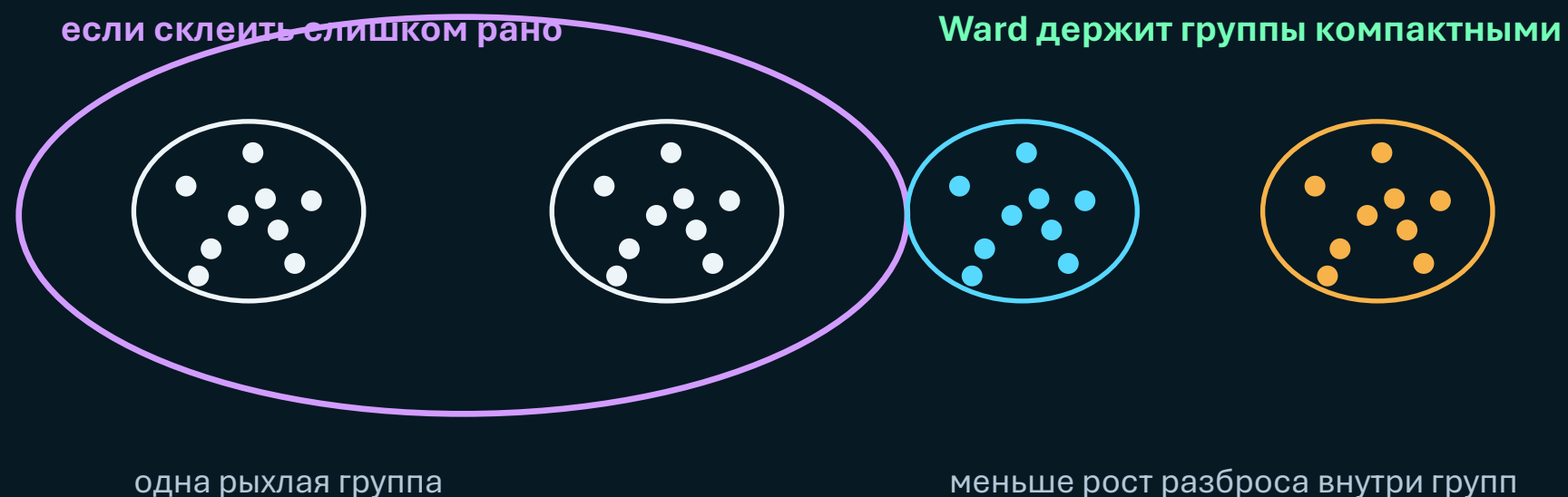


равный вес групп

Не обязательно любить все linkage одинаково; важно понимать, какую ошибку они склонны делать.

Зачем часто берут Ward?

Ward старается объединять группы так, чтобы разброс внутри кластеров рос как можно меньше.



Интуиция Ward

Не просто “есть ли близкая пара?”, а “не испортим ли мы компактность этим объединением?”.

Почему это нравится

Часто получаются более собранные и читаемые группы, без лишних цепочек.

Честный вывод

Ward — не истина, а выбор, который стоит проверять на данных.

Ward часто ближе к интуиции “хороших компактных групп”, но не превращает результат в истину.

Как выбирать linkage: короткая таблица

Финальный совет преподавателя: выбираем не “самый умный” linkage, а подходящую логику ошибки.

Linkage	Сильная сторона	Подводный камень	Совет
Single	ловит мостики и вытянутые формы	цепочки могут склеить лишнее	редко как первый выбор
Complete	строго держит группы отдельно	может дробить данные	если опасны рыхлые кластеры
Average	спокойный компромисс	может сгладить крайности	хороший базовый вариант
Weighted	учитывает группы более равноправно	менее прозрачен новичкам	когда размеры групп мешают интуиции
Ward	держит кластеры компактными	любит геометрию разброса	если нужны читаемые компактные ветви

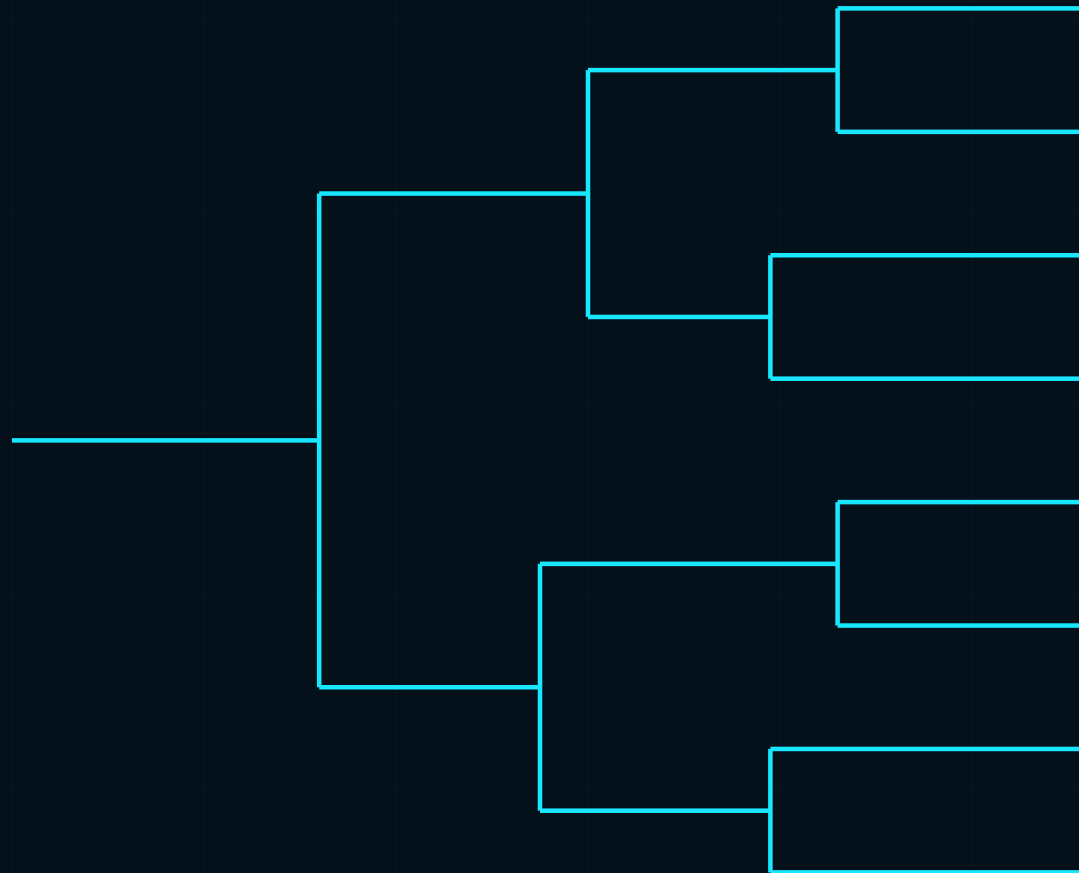
Совет от преподавателя

Сравните хотя бы два linkage на важных данных и спросите: “какая ошибка для моей задачи опаснее?”

Красивое дерево — начало разговора, а не финальный приговор.

Повторяем объединения

маленькие ветки сливаются в большие

**1**

ближайшие
пары

2

маленькие
группы

3

крупные
ветви

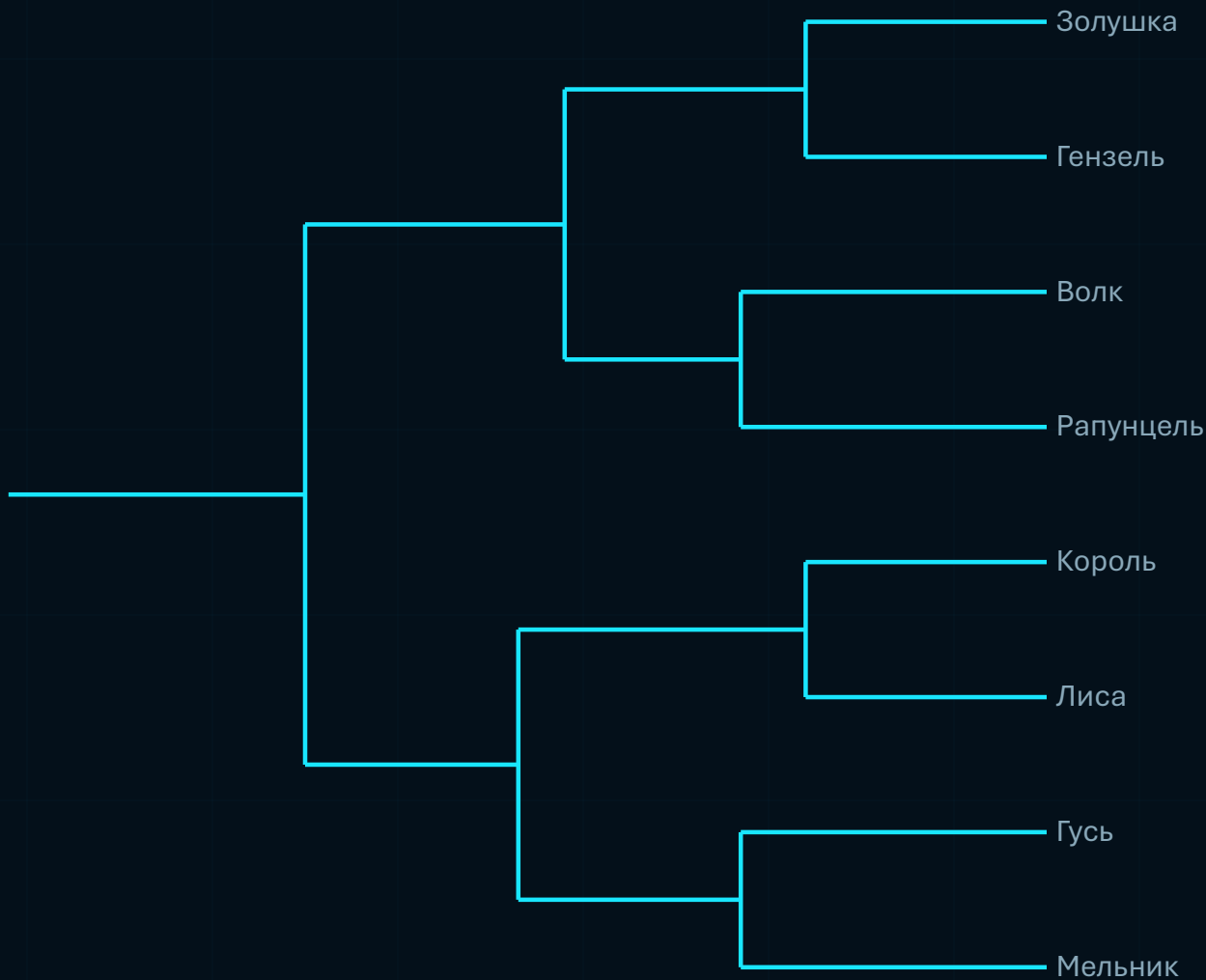
4

одно
дерево

Дерево хранит порядок объединений.

В конце всё объединится

но нас интересуют промежуточные уровни



Листья

отдельные сказки

Ветки

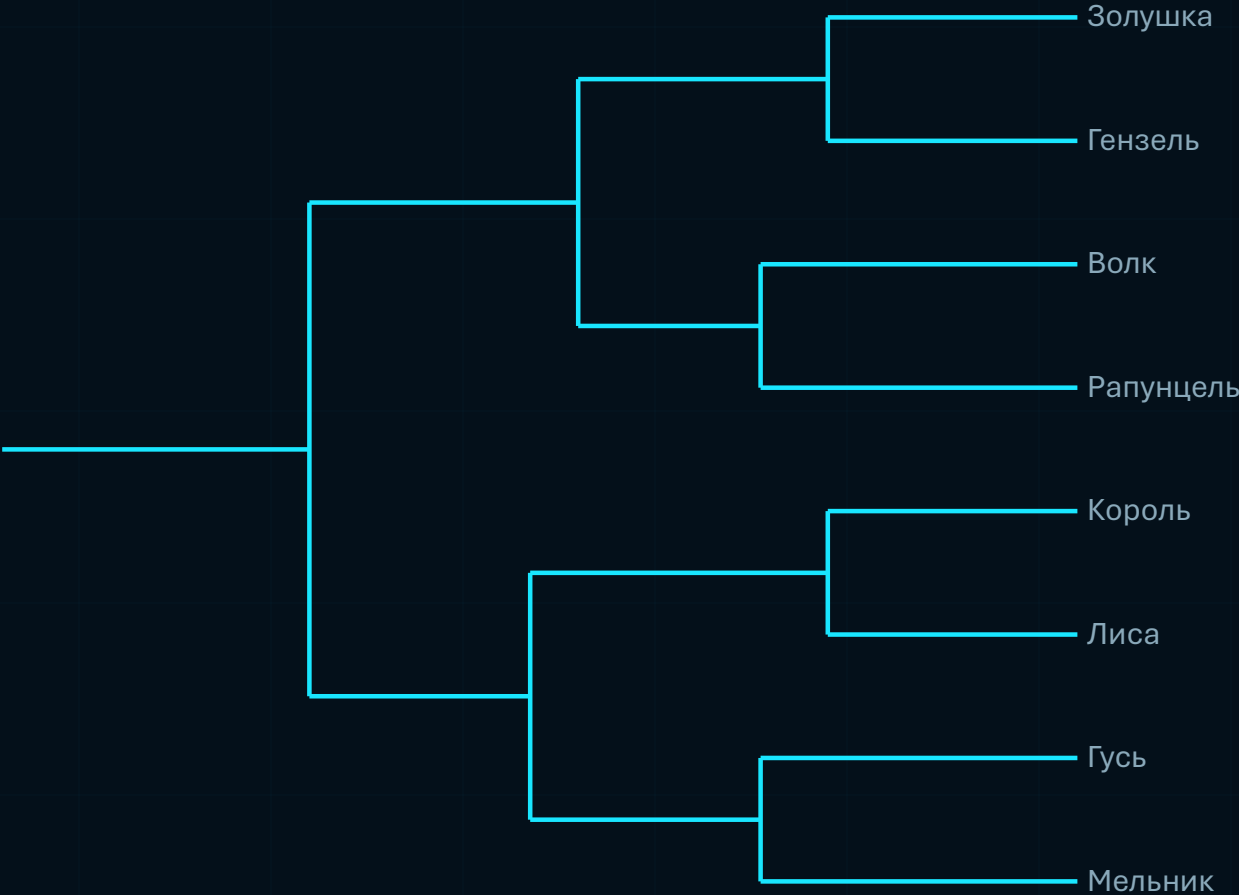
кто с кем сблизился

Корень

все сказки вместе

Как читать дендрограмму

три правила без формул



1. Листья
объекты: у нас это сказки

2. Раннее объединение
обычно означает большую похожесть

3. Длинный скачок
объединяем уже более разные группы

Одна дендрограмма — много разрезов

это ключевое преимущество метода

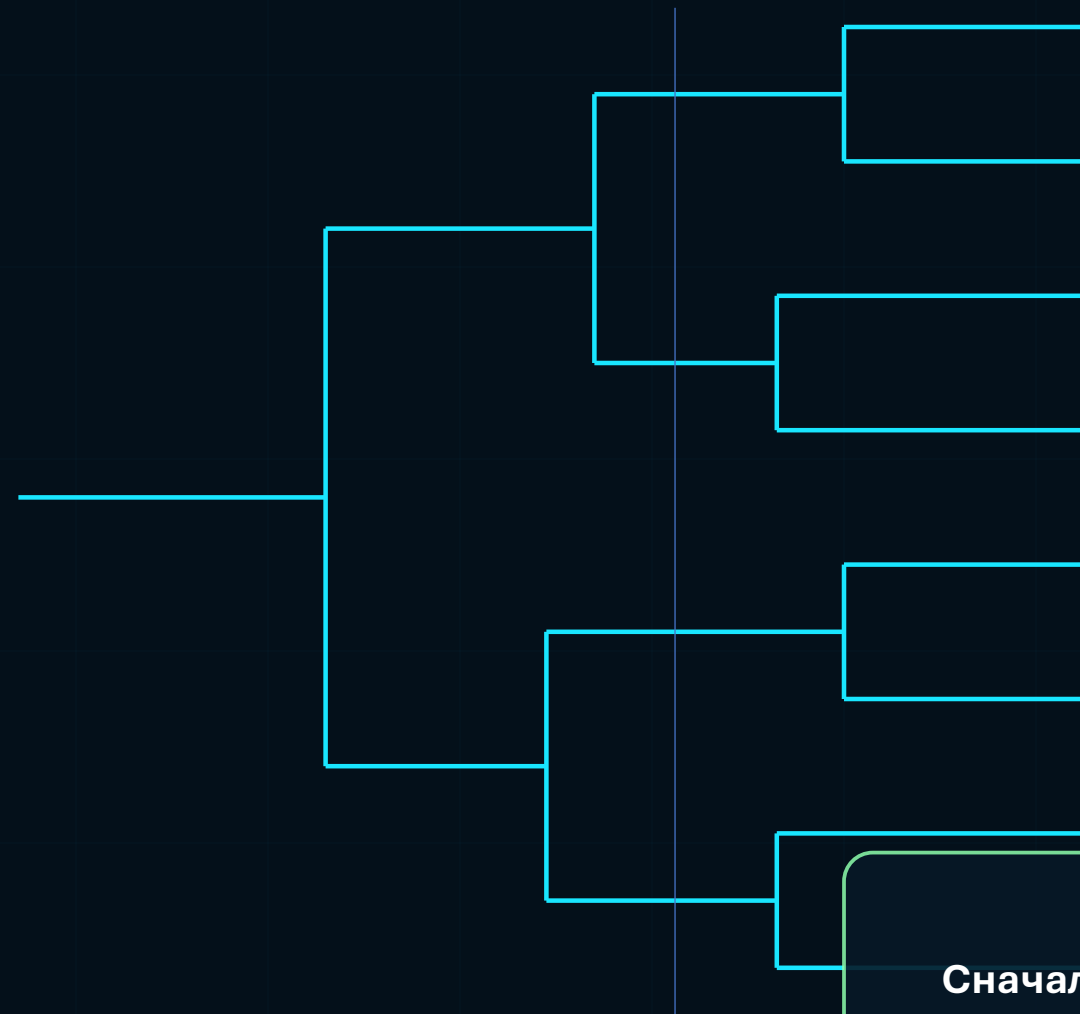


грубый разрез

Мы не обязаны заранее знать k .
Сначала видим дерево, потом выбираем уровень.

Одна дендрограмма — много разрезов

это ключевое преимущество метода



средний разрез

Мы не обязаны заранее знать k .
Сначала видим дерево, потом выбираем уровень.

Одна дендрограмма — много разрезов

это ключевое преимущество метода



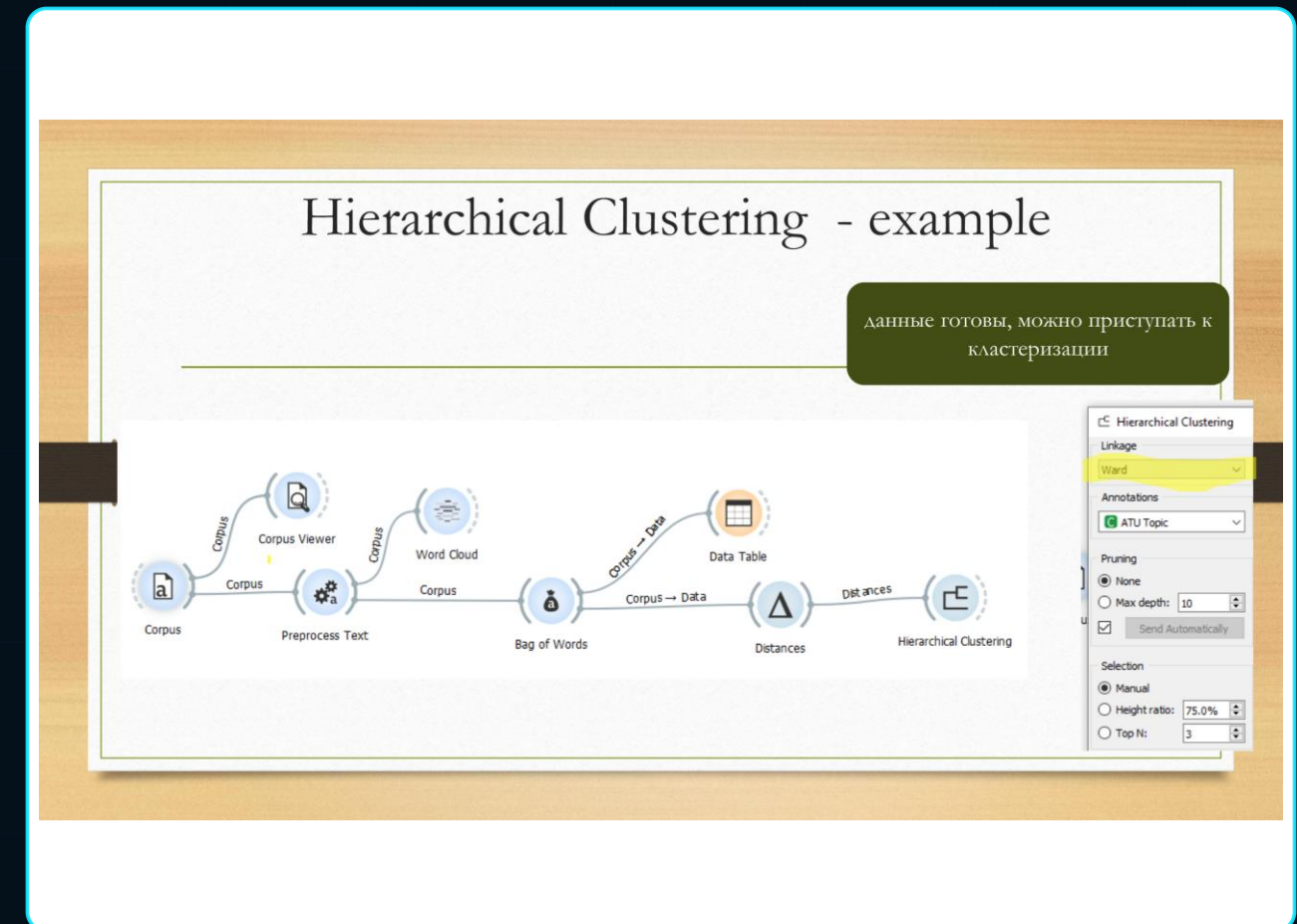
детальный разрез

Мы не обязаны заранее знать k .
Сначала видим дерево, потом выбираем уровень.

Подключаем Hierarchical Clustering

теперь Orange строит дерево по расстояниям

- на вход подаём расстояния
- выбираем linkage
- получаем дендрограмму
- не ищем “истину” одной кнопкой

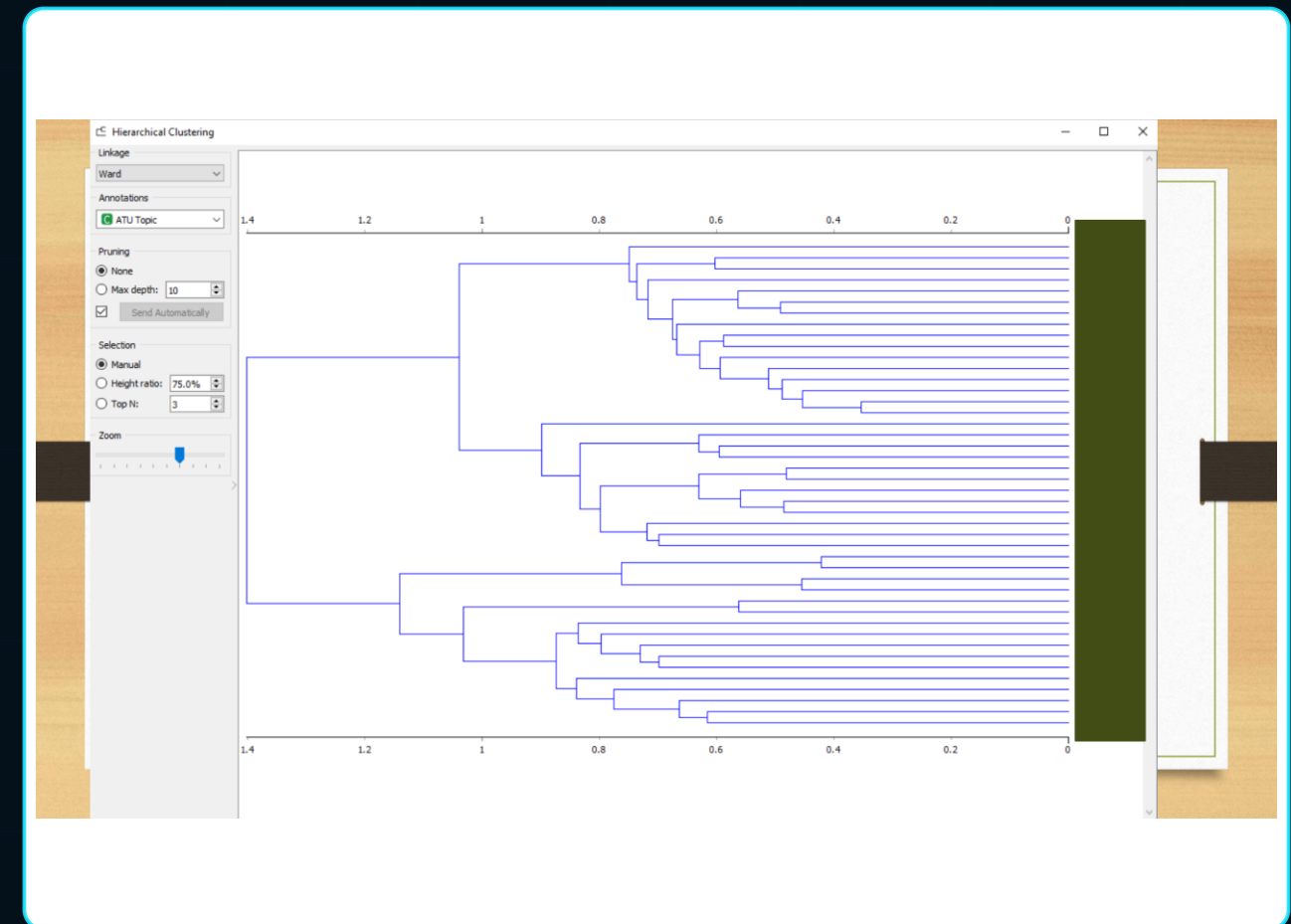


Distances → Hierarchical Clustering

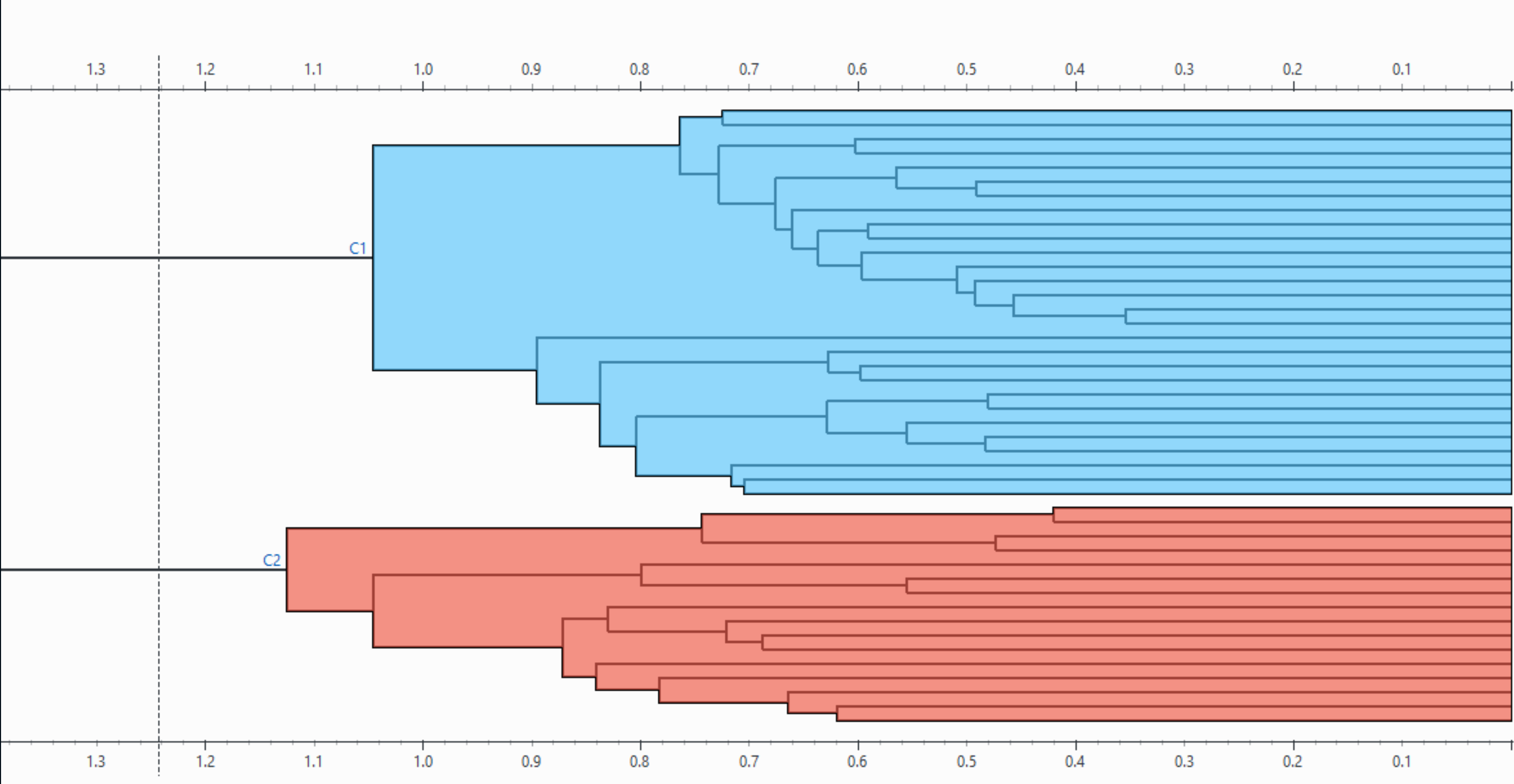
Настоящее дерево сказок

первый взгляд на дендрограмму

- каждый лист — сказка
- близкие ветви объединяются раньше
- длинные ветки показывают крупные различия
- смотрим на структуру, а не на красоту



дендрограмма корпуса Grimm



Важная проверка

метки можно открыть только после кластеризации

**Тип сказки не участвовал
в построении дерева.**

До кластеризации

алгоритм видит только
текстовые признаки и
расстояния

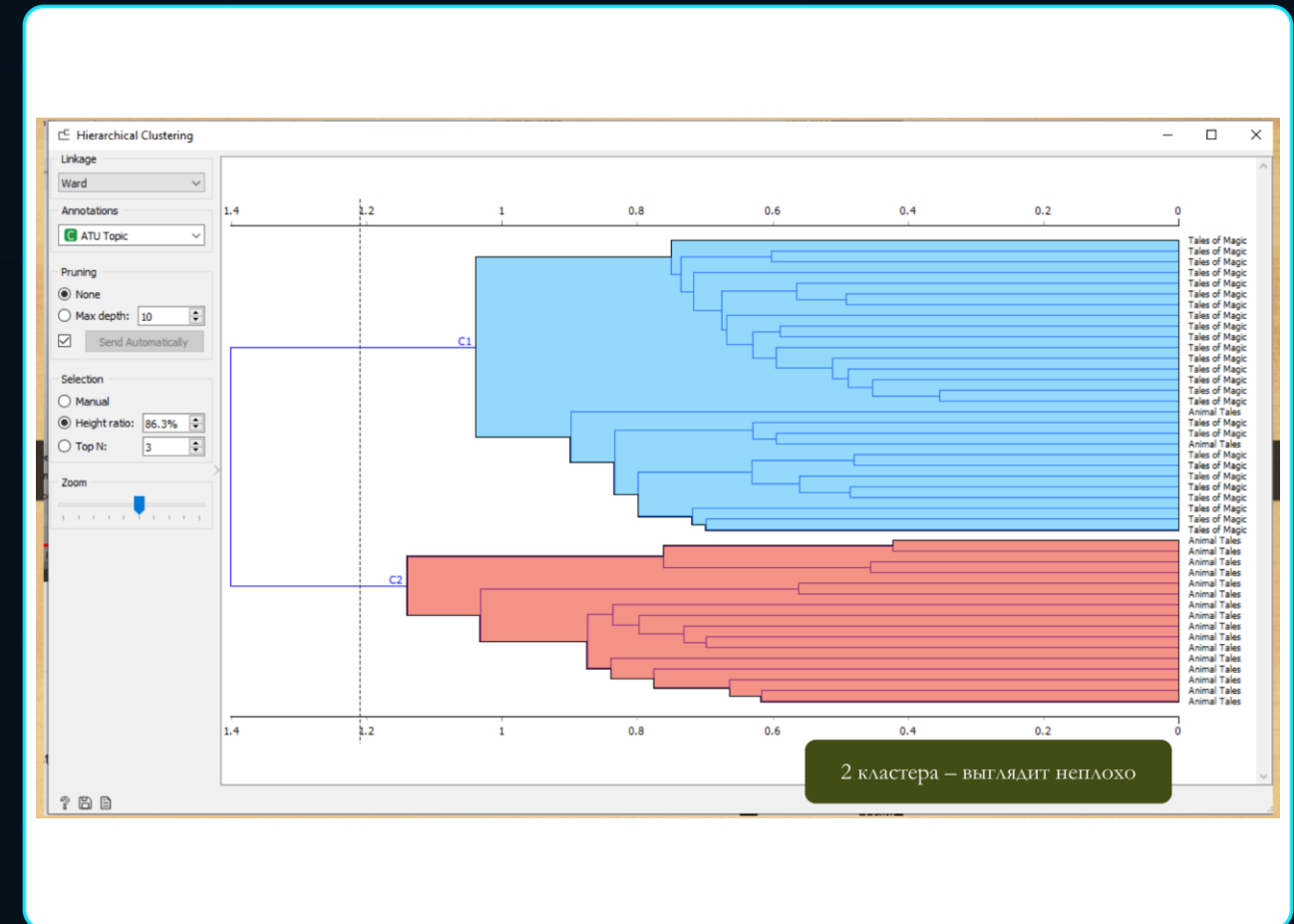
После результата

можно сравнить с
известными категориями

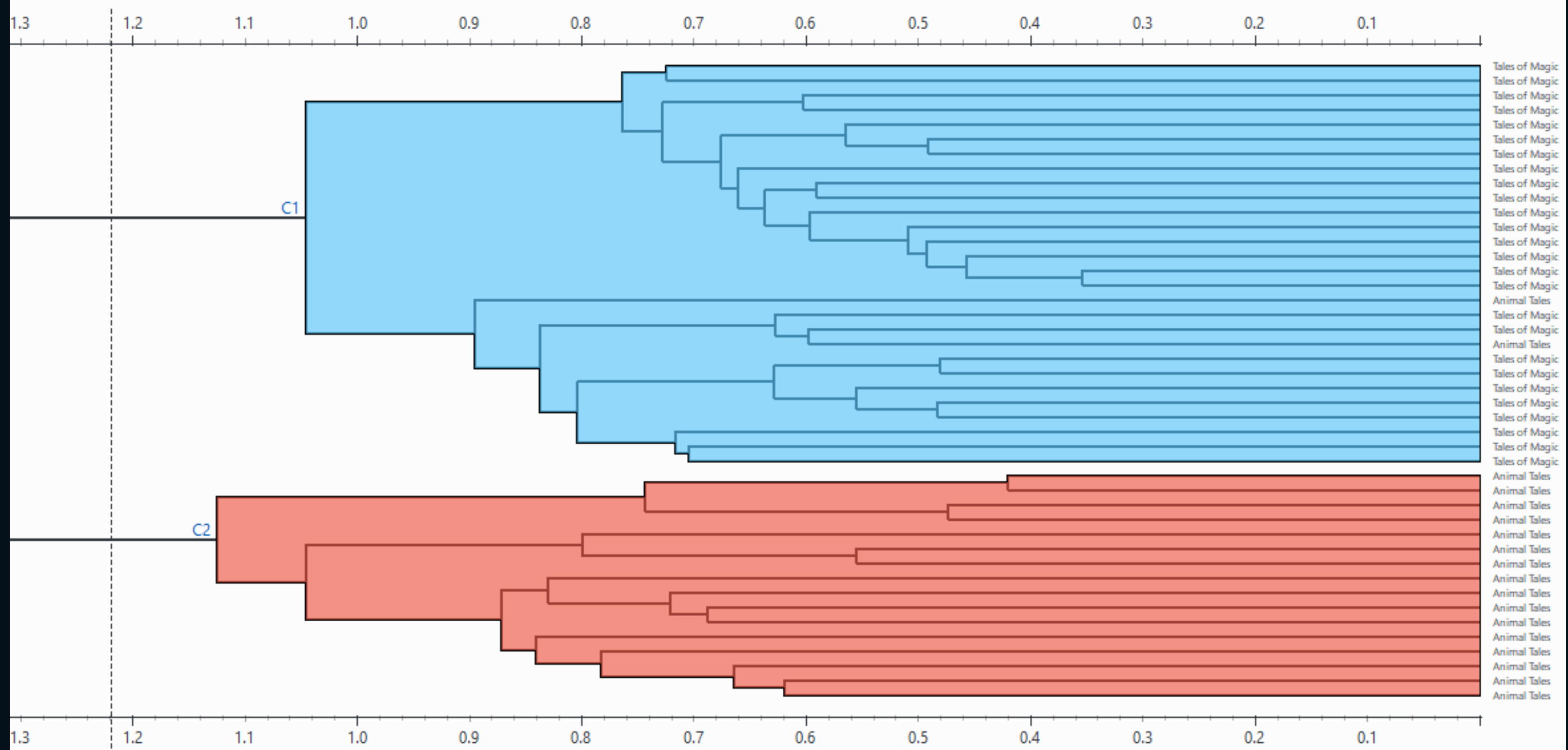
Разрез на 2 кластера

выглядит довольно разумно

- получаем крупную картину
- меньше деталей, больше устойчивости
- хорошо для первого обзора
- но можно потерять важные подгруппы



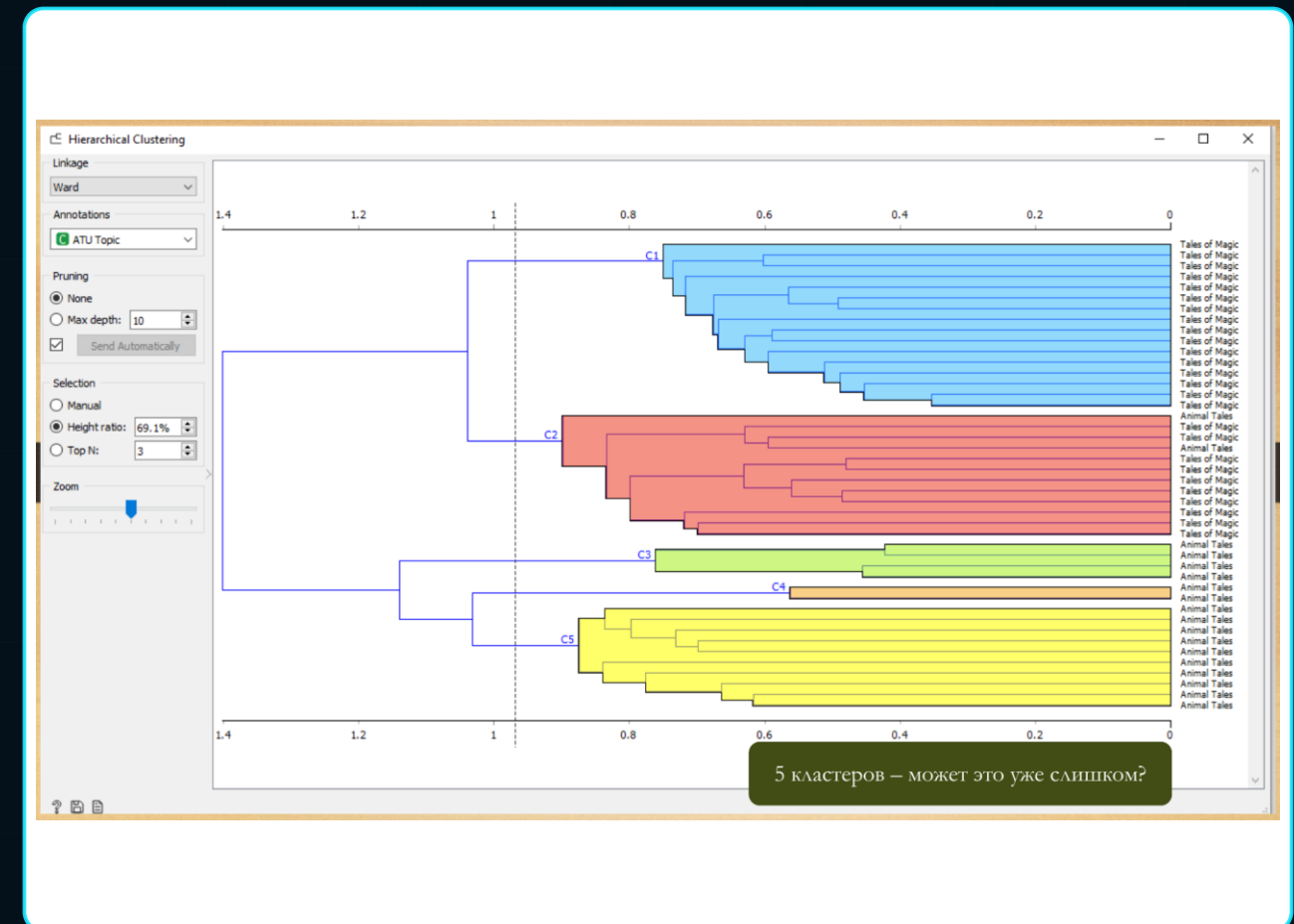
два крупных блока на дендрограмме



Разрез на 5 кластеров

уже детальнее — но не обязательно лучше

- детализация выше
- можно увидеть подгруппы
- растёт риск переинтерпретации
- “больше кластеров” не значит “глубже анализ”



много цветов, больше мелких групп

Вопрос

10.3 · выбор разреза

?

**Если разрез на 2 кластера выглядит красиво,
а разрез на 5 даёт больше деталей,
какой вариант правильный?**

Подсказка: в обучении без учителя нет учителя, который сразу скажет “правильно”.

Ответ

10.3 · ответ



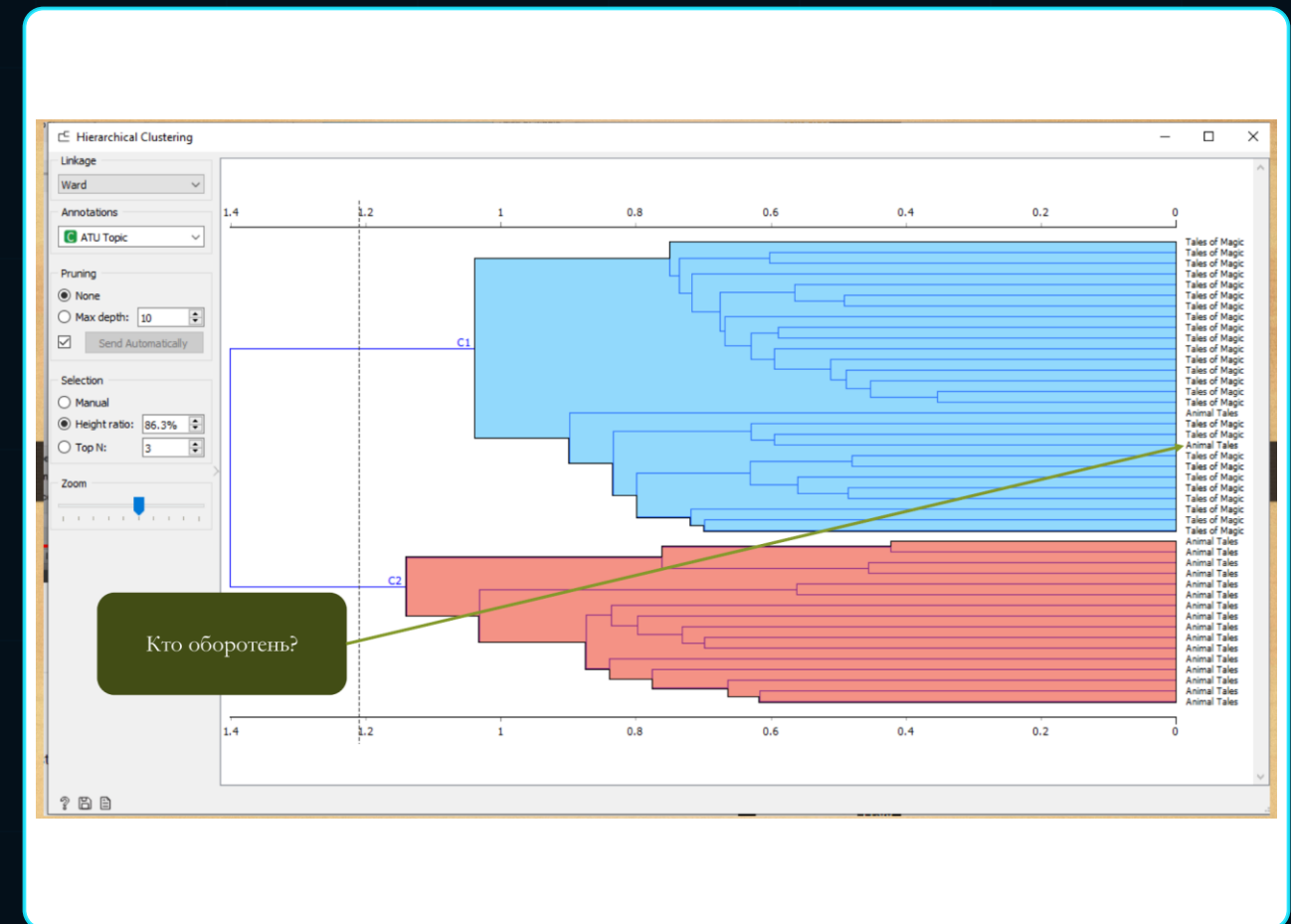
Зависит от задачи.

2 кластера могут быть хороши для общей картины.
5 кластеров — для более тонкого исследования.
Но оба ответа надо проверять содержательно.

Кто оборотень?

самые интересные места часто выглядят как ошибка

- не кричим “алгоритм сломался”
- сначала смотрим объект
- проверяем текст и признаки
- иногда странность — главный результат



одна сказка попала в неожиданный блок

Волк и семеро козлят

почему алгоритм решил именно так?

- открываем текст
- смотрим слова и профиль
- сравниваем с соседями
- объясняем решение через данные

The screenshot shows a data analysis workflow in Orange3. The workflow consists of the following steps: Corpus (input) → Preprocess Text → Corpus Viewer → Word Cloud → Corpus → Bag of Words → Data Table → Distances → Hierarchical Clustering → Selected Data → Corpus Viewer (1). A green box with the text "Почему-то алгоритм решил, что волк и семеро козлят волшебная сказка. Почему?" (Why did the algorithm decide that the wolf and seven kids are a magical story. Why?) is overlaid on the workflow. Below this box is a small image of the story's cover, "The Wolf and the Seven Little Goats". The Corpus Viewer (1) window is open, showing a list of documents. Document 2, "The Wolf and the Seven Young Kids", is selected. The window displays the story's title, abstract, and content. The content is a text description of the story, starting with "There was once upon a time an old goat who had seven little kids, and loved them with all the love of a mother for her children. One day she wanted to go into the forest and fetch some food. So she called all seven to her and said: 'Dear children, I have to go into the forest, be on your guard against the wolf. If he comes in, he will devour you all-skin, hair, and everything. The wretch often disguises himself, but you will know him at once by his rough voice and his black feet.' The kids said: 'Dear mother, we will take good care of ourselves; you may go away without any anxiety.' Then the old one bleated, and went on her way with an easy mind. It was not long before someone knocked at the house-door and called: 'Open the door, dear children; your mother is here, and has brought something back with her for each of you.' But the little kids knew that it was the wolf, by the rough voice. 'We will not open the door,' cried they, 'you are not our mother. She has a soft, pleasant voice, but your voice is rough; you are the wolf! Then the wolf went away to a shonkeen and

открываем сказку и проверяем содержание

Плюсы иерархической кластеризации

почему дерево полезно

Видно структуру

не только итоговые группы, но и близость между ними

Не нужно k сразу

сначала строим дерево, потом выбираем уровень

Хорошо для разведки

удобно изучать маленькие и средние наборы данных

Можно найти странности

необычные ветви и “пограничные” объекты

Легко объяснять

дендрограмма понятна как дерево отношений

Минусы и ограничения

дерево тоже не волшебное

Тяжело на больших данных

может требовать много
времени и памяти

Зависит от метрики

другое расстояние — другое
дерево

Зависит от linkage

правило объединения меняет
структуру

Разрез выбирает человек

дерево не выдаёт одну истину

Красиво ≠ верно

визуальная убедительность может обмануть

Совет от доктора Игоря

для работы с дендрограммой

**Сначала смотрите на дерево.
Потом выбирайте число кластеров.
Не наоборот.**

И обязательно спрашивайте: что означают найденные ветви?

Summary

10.3: Hierarchical Clustering

Старт

каждый объект отдельно

Объединение

ближайшие группы сливаются

Linkage

правило сравнения групп

Дендрограмма

история объединений

Разрез

выбираем уровень детализации

Проверка

смотрим смысл ветвей

Дерево показывает отношения, а не только готовые группы.

Если карта возможных ответов. коротко: иерархическая кластеризация — это не один ответ, а

K-means

ищем центры кластеров

**Быстрый и понятный метод, но число кластеров
нужно задать заранее.**

Доктор Игорь Клейнер



Зачем ещё один метод?

дерево красиво, но иногда тяжело

Hierarchical показывает структуру.
K-means быстро строит группы.

Особенно когда объектов много и мы готовы заранее выбрать k .

Дерево красиво рассказывает историю, но иногда нам нужен быстрый рабочий инструмент.

Главное отличие

k-means начинает с числа k

Сначала говорим алгоритму:
“Ищи k кластеров”.

Hierarchical

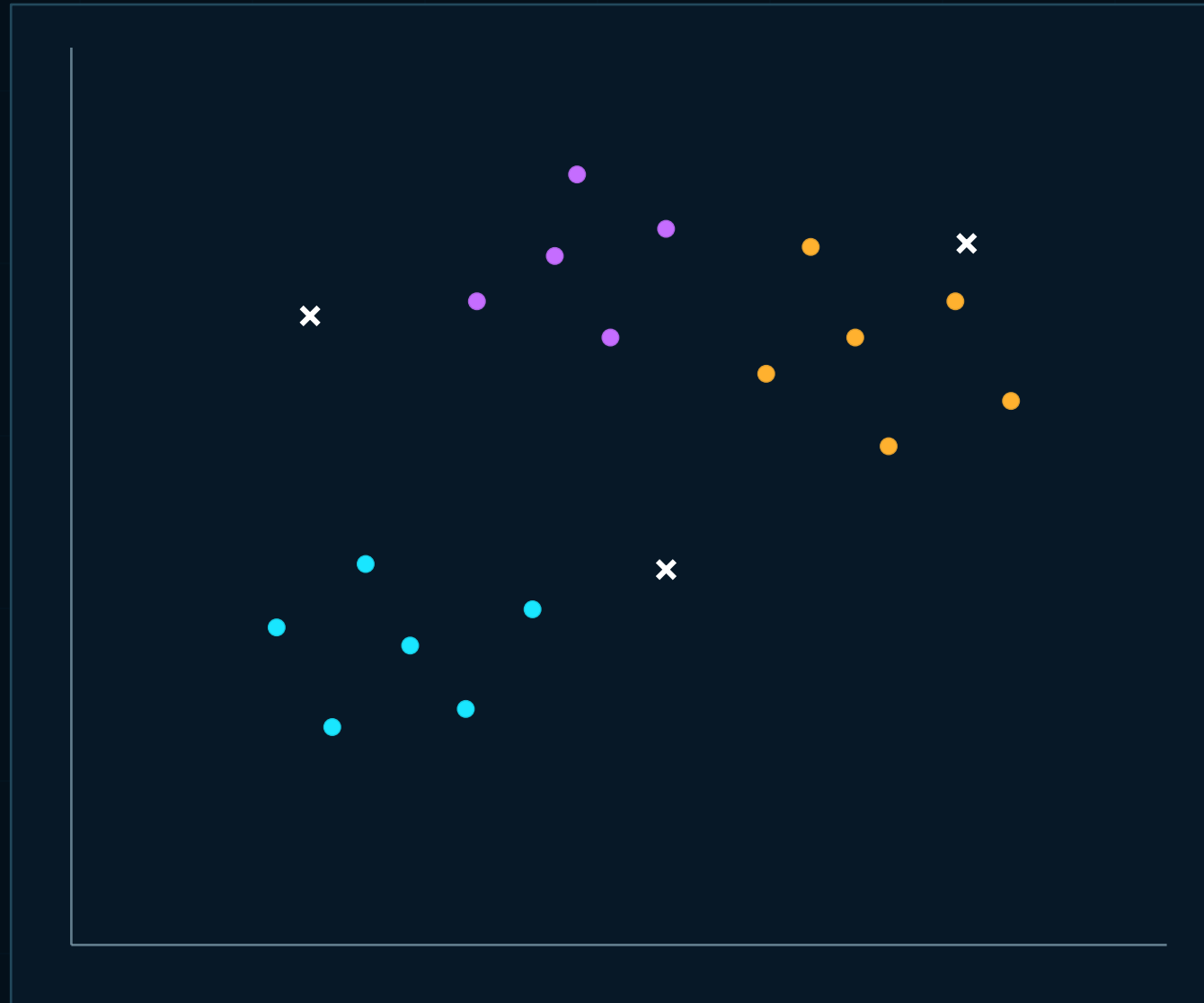
сначала дерево, потом разрез

K-means

сначала k, потом группы

Шаг 1: выбираем центры

обычно старт случайный



k = 3

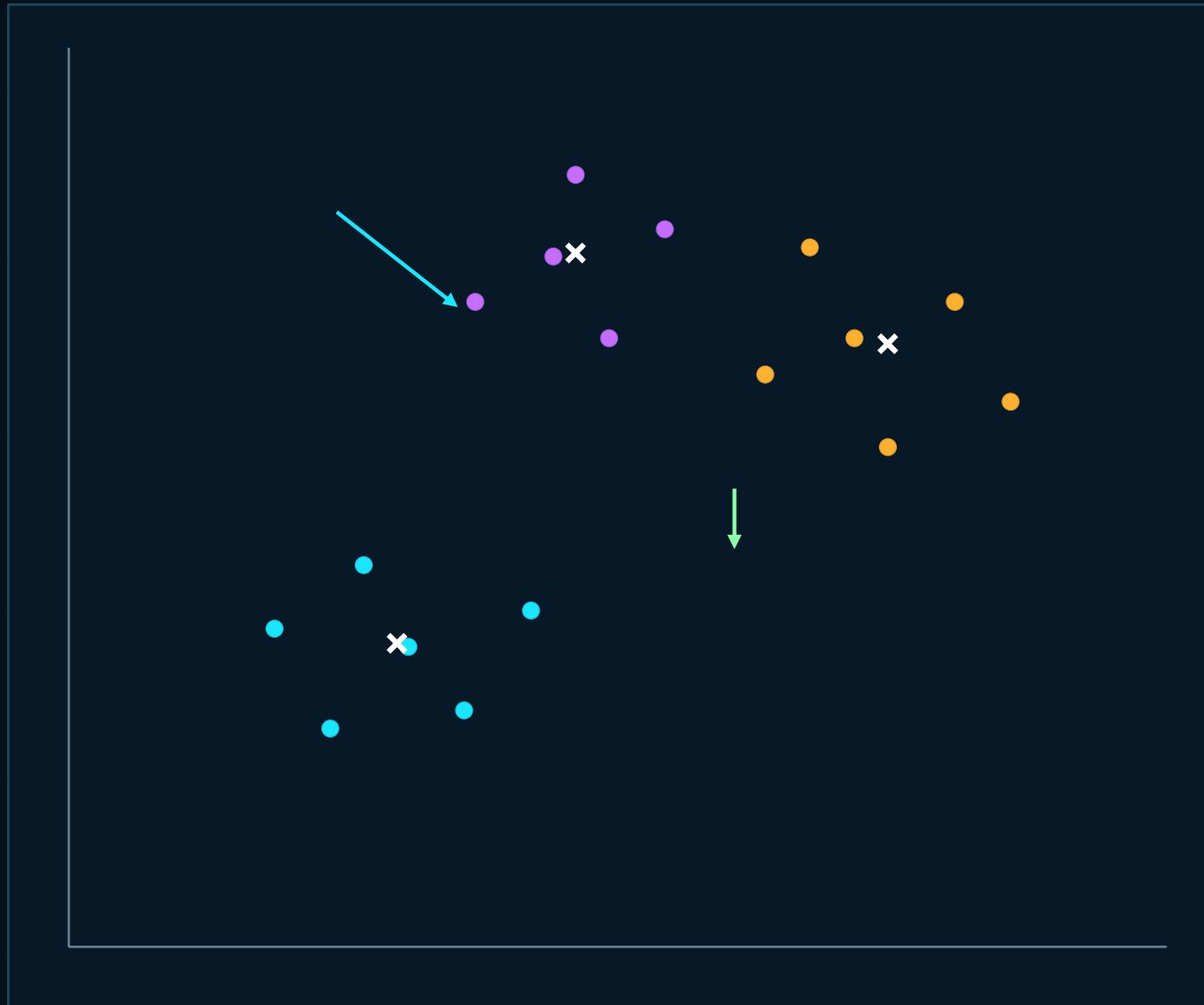
ставим три центра

Важно

начальный выбор может повлиять на результат

Шаг 2: назначаем точки центрам

каждый объект идёт к ближайшему центру



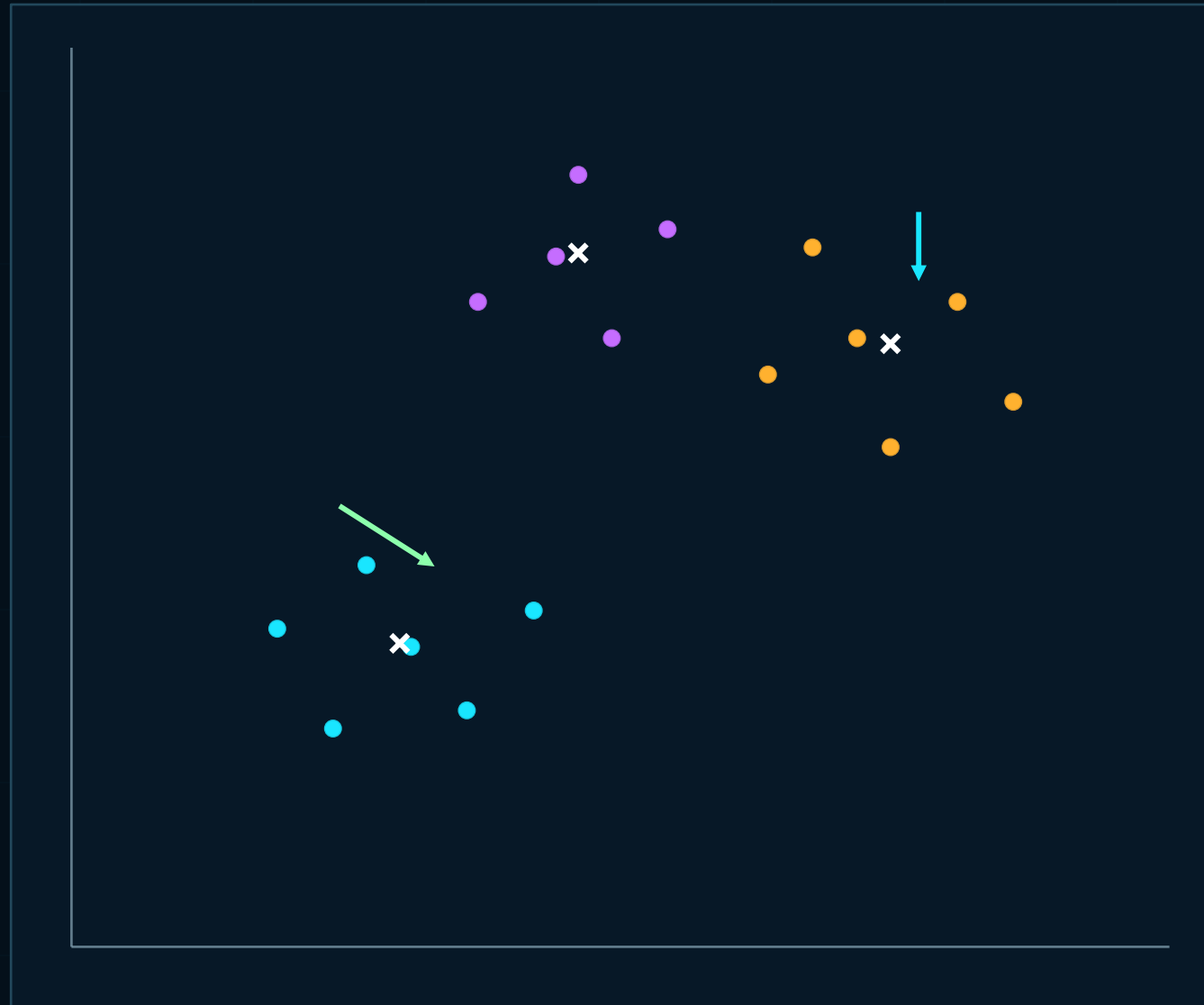
Правило

точка получает цвет ближайшего центра

Пока это только черновое разбиение.

Шаг 3: центры переезжают

means = средние



means

центр кластера — среднее положение его точек

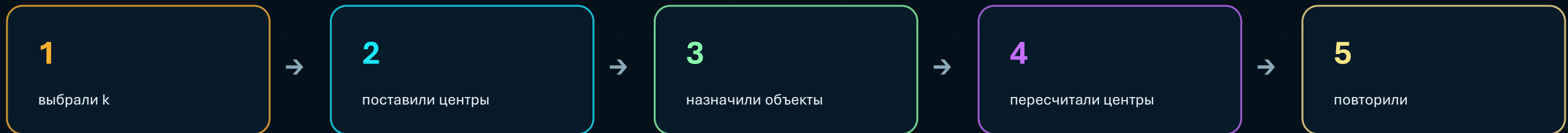
Переезд

после назначения точек центры сдвигаются

K-means одним слайдом

повторяем до стабилизации

центры → назначение → новые центры → назначение → ...

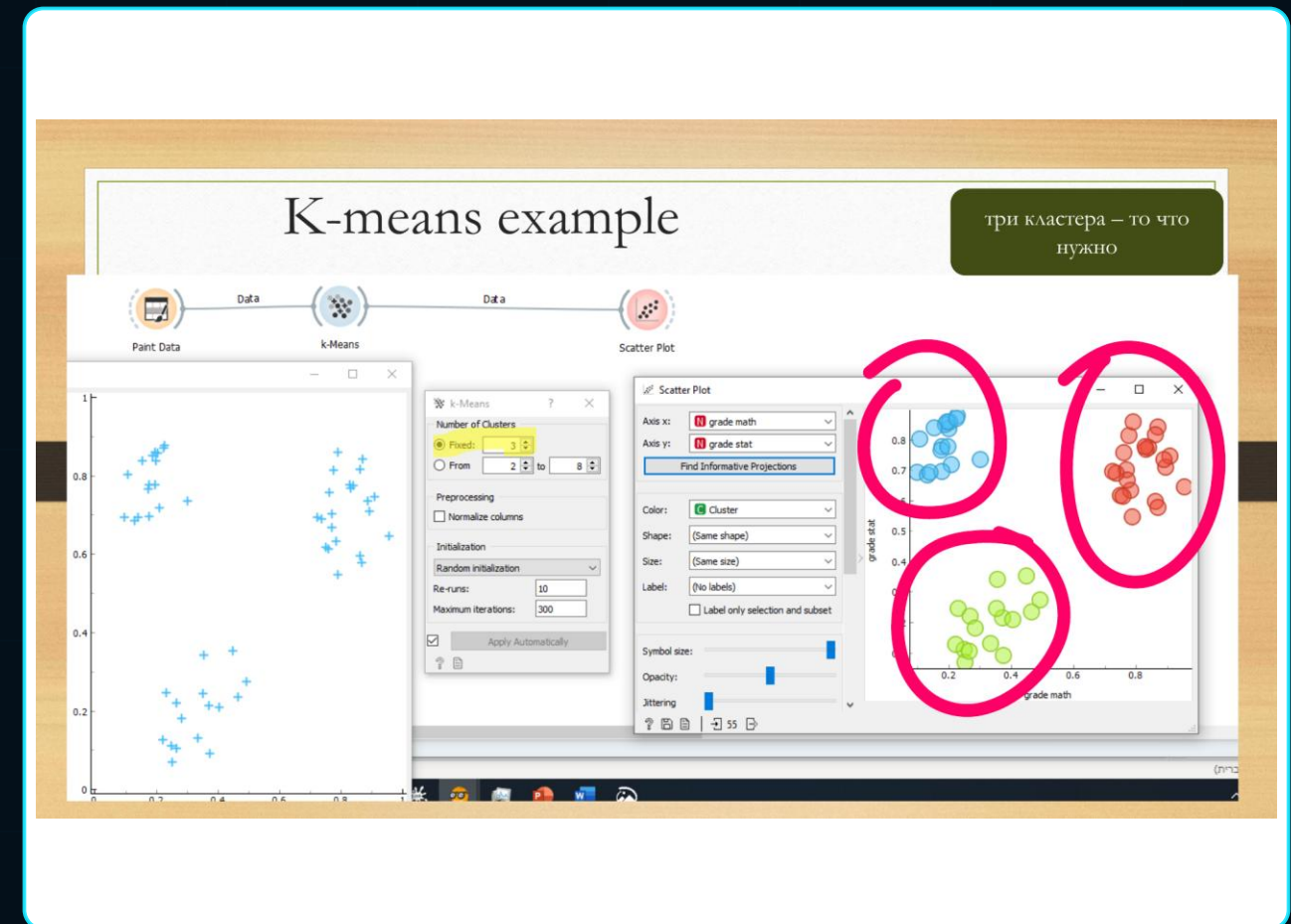


Когда почти ничего не меняется — алгоритм остановился.

K-means в Orange

на искусственных данных видно, что происходит

- выбираем число кластеров
- получаем цвета групп
- смотрим scatter plot
- проверяем, имеет ли разбиение смысл

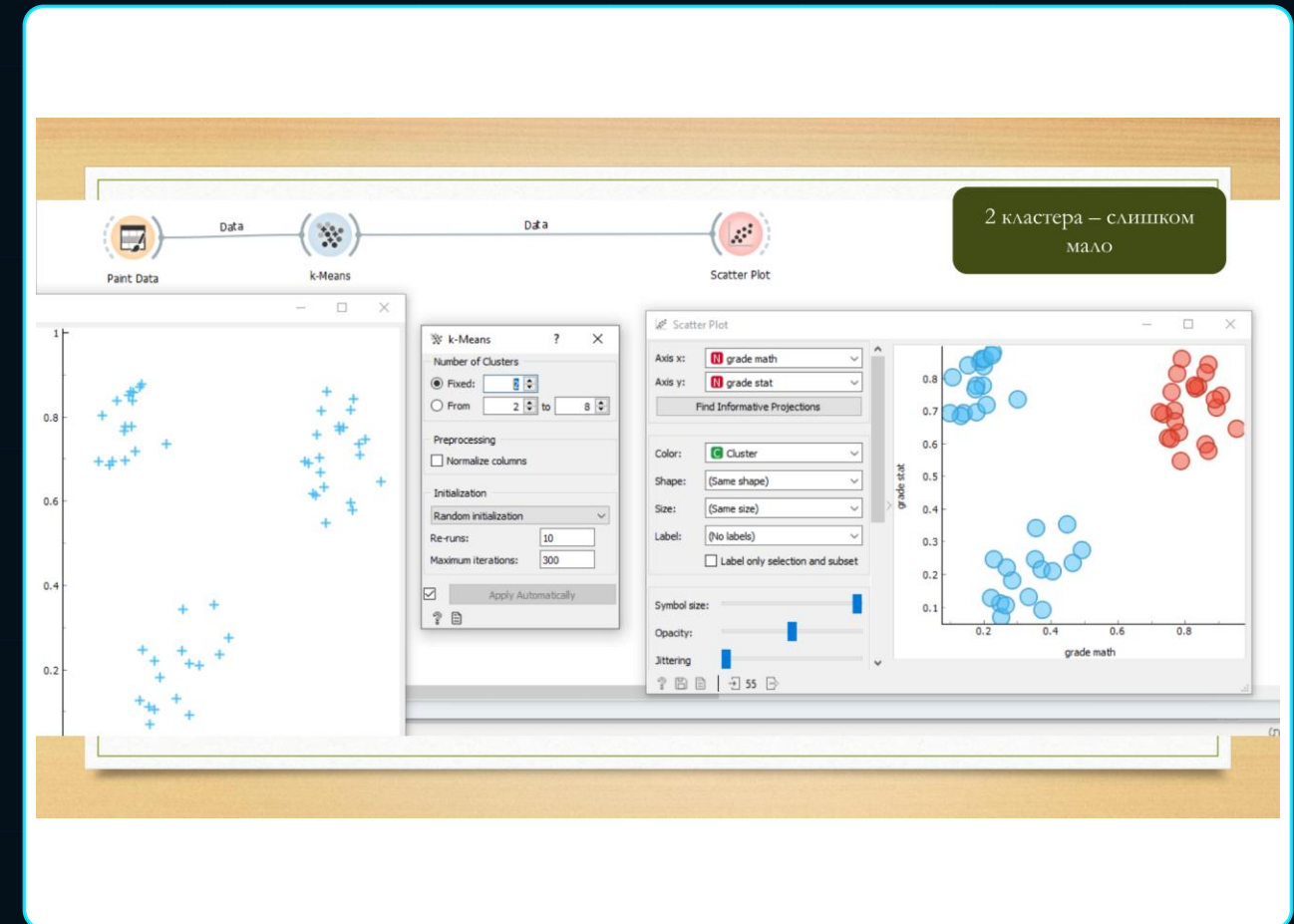


$k = 3$ выглядит естественно

$k = 2$: слишком грубо?

часть структуры исчезает

- меньше групп
- проще объяснить
- но два естественных облака могут склеиться
- вопрос: подходит ли это задаче?

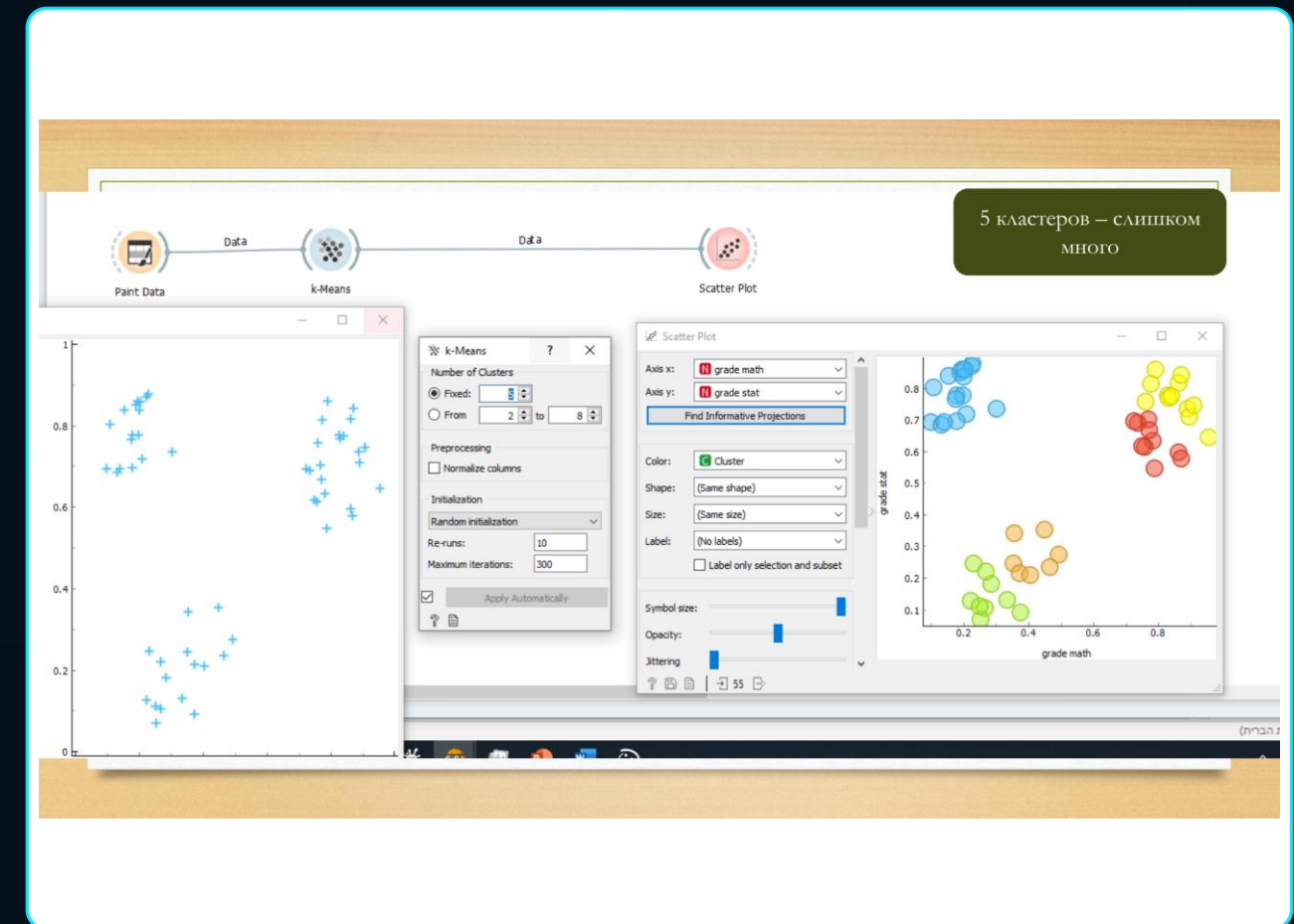


два кластера на тех же данных

$k = 5$: слишком мелко?

данные можно раздробить почти всегда

- больше цветов
- больше деталей
- выглядит “умнее”
- но может быть искусственным дроблением



пять кластеров на тех же данных

Вопрос

10.4 · главный практический вопрос

?

**Если k-means требует заранее выбрать k ,
откуда нам взять это число?**

Иногда помогает область задачи. Иногда — графики. Иногда — метрики. Но автоматической истины нет.

Почему результат может меняться?

случайный старт — не мелочь

Старт А

центры попали удачно
→ группы выглядят хорошо

Старт В

центры попали плохо
→ результат хуже

Практика

делают несколько запусков
и выбирают стабильный
вариант

Вывод

одинаковые данные и одинаковое k не гарантируют
полностью одинаковый результат

K-means любит компактные группы

сложные формы могут сломать интуицию

- метод ищет центры
- границы получаются вокруг центров
- изогнутые формы делятся плохо
- красивые цвета могут быть неверной историей

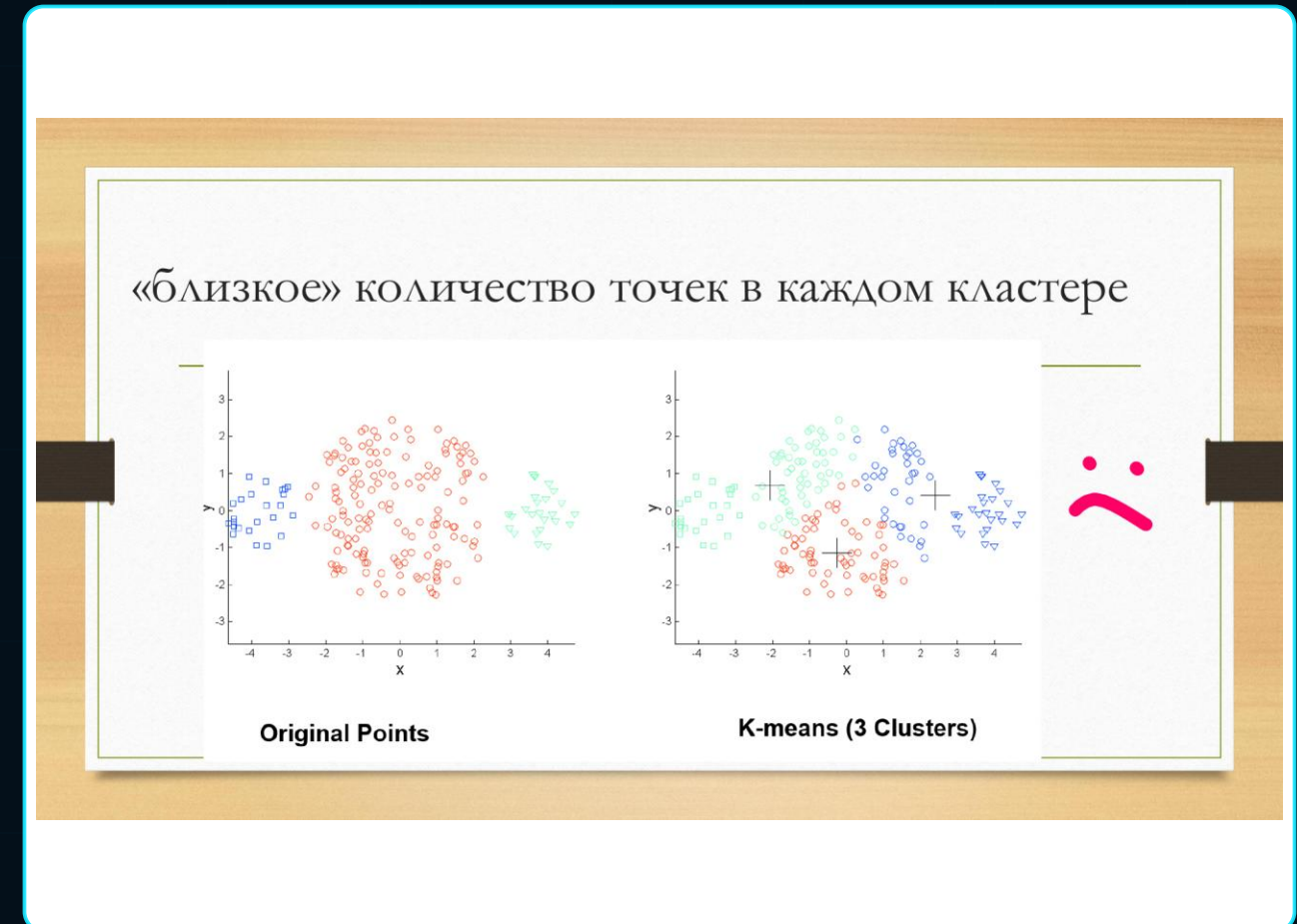


пример с изогнутыми кластерами

Неравные размеры и плотности — сложный случай

k-means оптимизирует расстояния до центров, а не “естественность” групп

- крупный кластер может раздробиться
- малый кластер может поглотиться
- разная плотность усложняет границы
- нужна проверка результата



большая или разреженная группа может быть разделена неожиданно

Ещё один подводный камень: масштаб

расстояние чувствительно к единицам измерения

Признак А

доход:
0–200 000

Признак В

оценка:
1–5

Что будет?

расстояние почти полностью
определит большой масштаб

Правило

перед кластеризацией числовые признаки часто
нормализуют

K-means: плюсы и минусы

короткая честная характеристика

Плюсы

- простой для объяснения
- быстрый на больших данных
- понятная идея центров
- хорошо работает для компактных групп

Минусы

- нужно заранее выбрать k
- старт может менять результат
- любит “круглые” группы
- чувствителен к масштабу и выбросам

Главная мысль

k-means — быстрый рабочий инструмент, но не оракул.

Совет от преподавателя

для красивых цветных графиков

**Если k-means нарисовал красивые цвета,
это ещё не значит,
что он нашёл разумные группы.**

Красивое разбиение нужно объяснить и проверить.

Summary

10.4: K-means

Выбираем k

число кластеров заранее

Ставим центры

часто случайно

Назначаем точки

к ближайшему центру

Двигаем центры

в среднее группы

Повторяем

до стабилизации

Проверяем

форма, масштаб, смысл

K-means быстро делит данные вокруг центров, но требует осторожности.

Вопрос

10.4 · проверка понимания

?

Если запустить k-means дважды на одних данных с одинаковым k, обязаны ли результаты совпасть?

Подумайте о первом шаге алгоритма.

Ответ

10.4 · ответ



Нет. Не обязаны.

K-means часто начинает со случайных центров.
Другой старт может привести к другому локальному решению, поэтому
результат проверяют и повторяют запуск.

Hierarchical vs K-means

как выбрать метод и можно ли доверять кластерам

Метод выбирают под задачу. Смысл кластеров проверяет человек.

Доктор Игорь Клейнер



Что даёт каждый метод?

два разных ответа на одну задачу

Hierarchical

строит дерево отношений

мы видим, кто с кем сближался и на каком уровне можно разрезать дерево

K-means

строит k групп вокруг центров

мы сразу получаем фиксированное разбиение на заданное число кластеров

Не победитель и проигравший

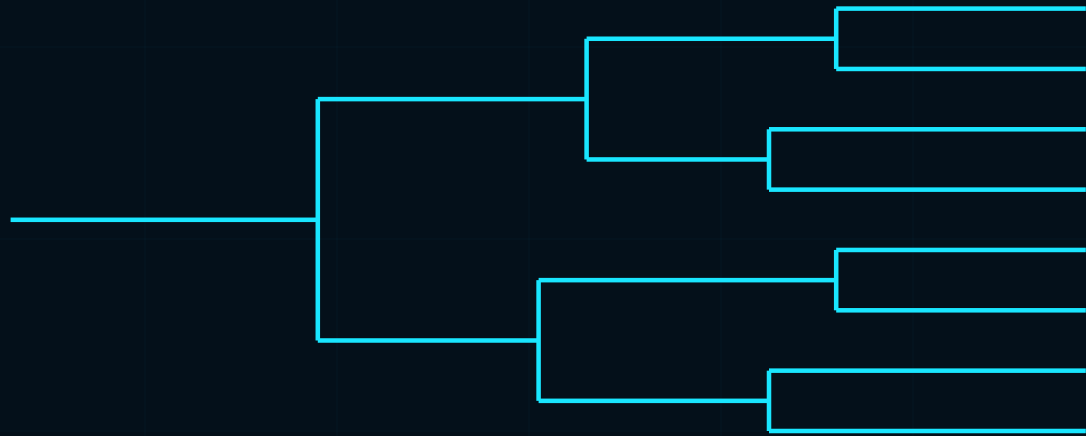
это разные инструменты для разных ситуаций

Главное отличие визуально

сначала структура или сначала k?

Hierarchical

сначала строим дерево



потом выбираем разрез

K-means

сначала выбираем k



потом получаем группы

Когда удобнее Hierarchical

дерево как инструмент исследования

Данных немного

можно позволить более тяжёлый метод

Не знаем к

хотим сначала увидеть структуру

Нужны отношения

важно кто ближе к кому

Нужна объяснимость

дерево легче показать аудитории

Ищем странные ветки

пограничные случаи могут быть ценными

Когда удобнее K-means

быстрый метод для заданного k

Данных много

важна скорость работы

k примерно известно

например, нужна сегментация
на 3–5 групп

Группы компактны

кластеры похожи на облака
вокруг центров

Нужен простой baseline

полезно начать с понятного метода

Нужны центры

центр помогает описать типичного
представителя

Сколько кластеров нужно?

иногда ответ даёт область задачи, иногда нет

Одежда

S / M / L / XL

число групп задано смыслом задачи

Футболисты

вратари / защита / атака

грубое предметное деление

Сказки

а вот здесь
число групп может быть
спорным

Вывод

“правильное k ” — не всегда математический факт

Silhouette: помощник, не оракул

оцениваем, насколько объект “сидит” в своей группе

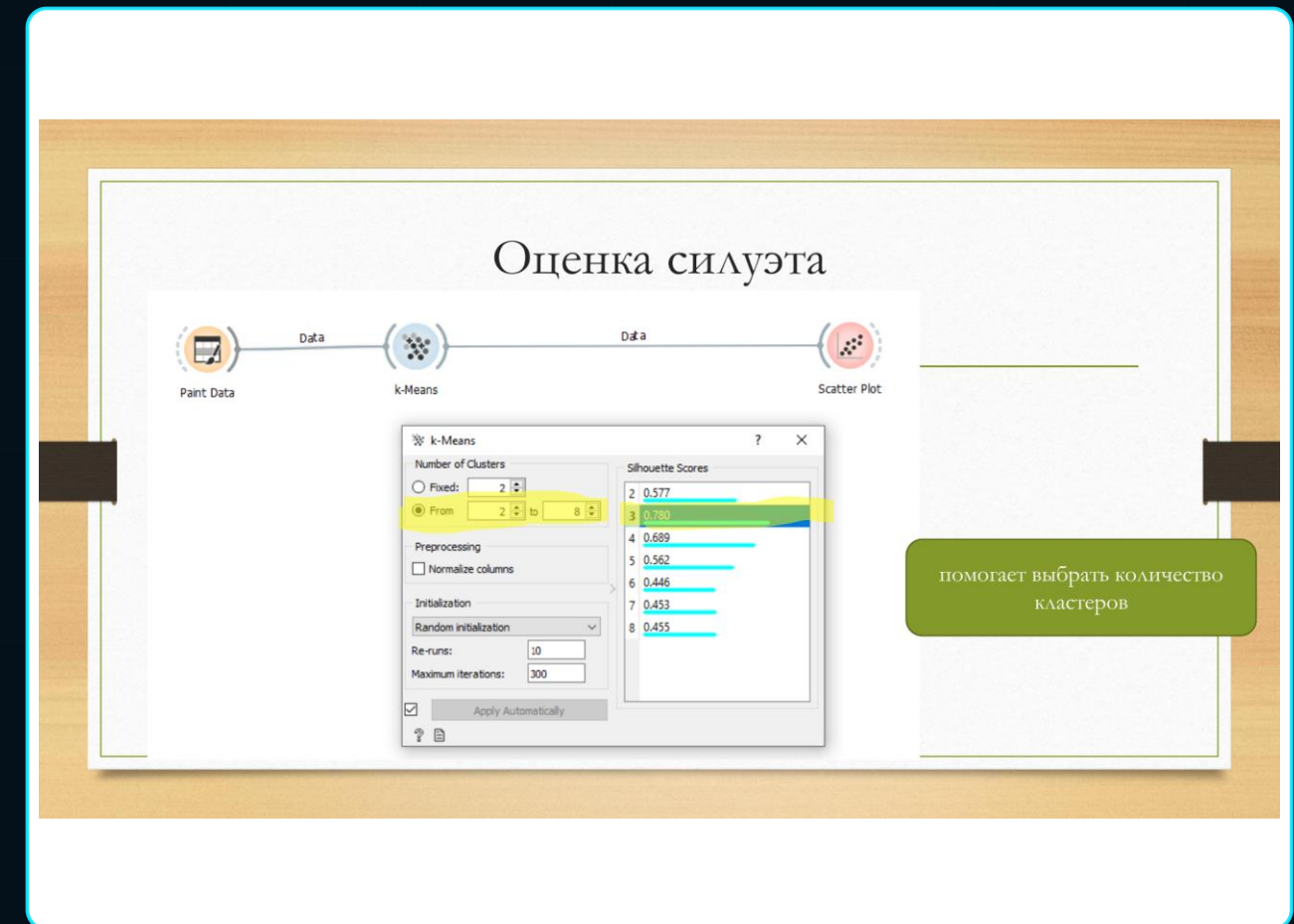
**Хороший объект близок к своему кластеру
и далёк от соседнего.**

Silhouette помогает сравнить k , но не доказывает “истинное” число кластеров.

Silhouette Score в Orange

сравниваем несколько k

- запускаем k-means для разных k
- смотрим средний silhouette
- лучшее число — кандидат
- но итог проверяем смыслом задачи



оценка силуэта помогает выбрать число кластеров

Silhouette plot: ищем проблемные объекты

не только среднее число

- видно качество внутри кластеров
- пограничные объекты выделяются
- можно вернуться к данным
- иногда “ошибка” — самая интересная сказка

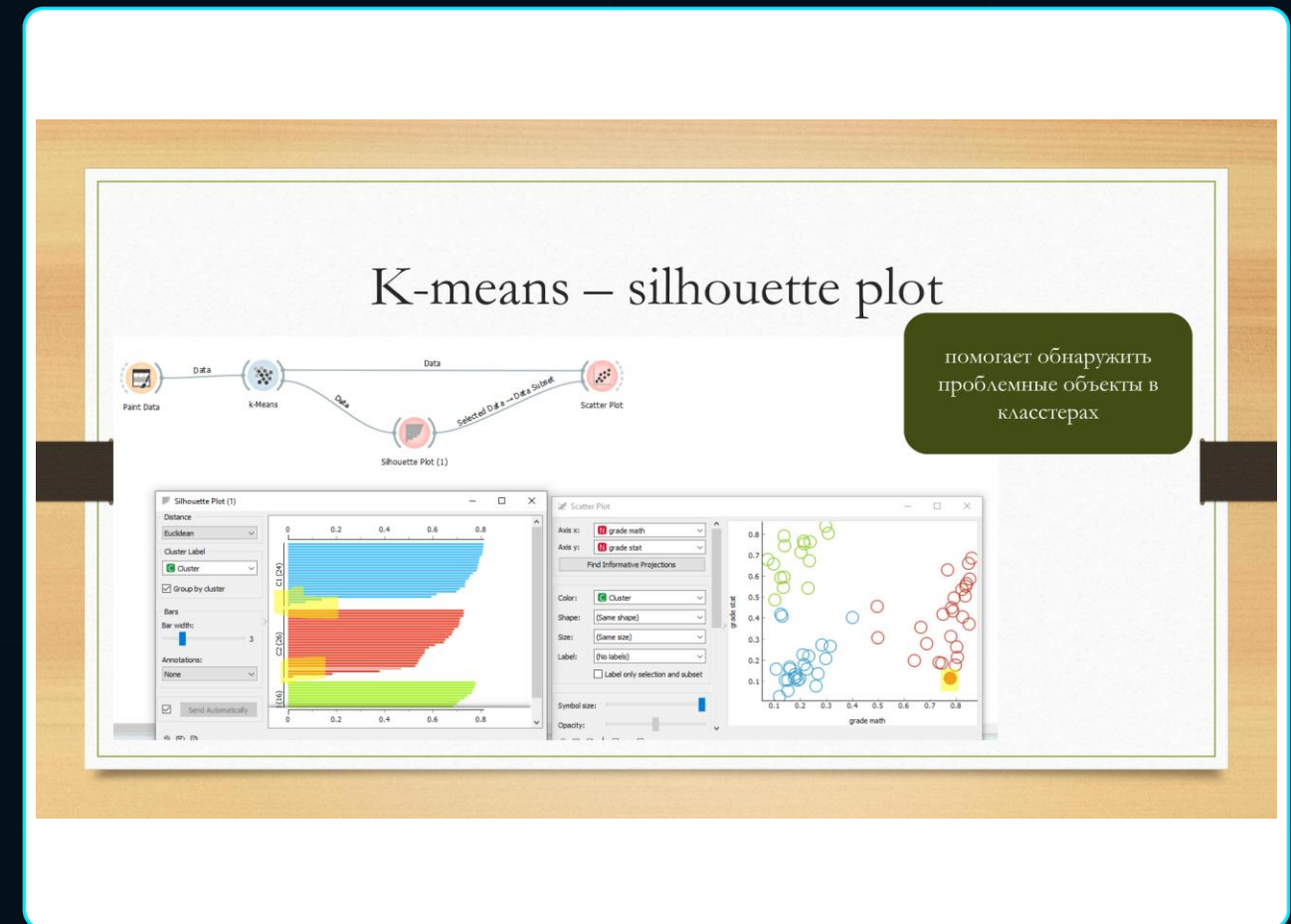


график силуэта показывает слабые места разбиения

Старый итог — теперь осмысленно

две строки сравнения уже не выглядят сухо

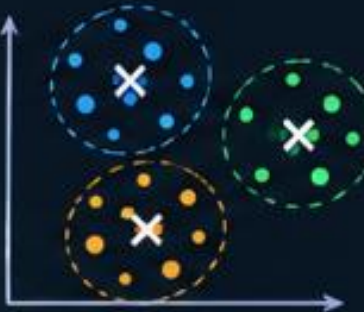
- hierarchical: дерево и структура
- k-means: скорость и центры
- оба зависят от признаков
- оба требуют интерпретации

Итоги			
	Used For	Pros	Cons
Hierarchical Clustering	поиск кластеров	визуальное представление результата в виде дерева, не нужно знать заранее количество кластеров	кубическое время работы
k-means Clustering		скорость работы	случайный алгоритм, нужно знать сколько кластеров искать, круглые кластеры

старый слайд сравнения методов

Иерархическая кластеризация vs K-means

Когда какой метод выбирать

Метод	Идея	Плюсы	Минусы	Сложность	Когда подходит
Иерархическая кластеризация 	Строит дерево близости объектов (дендрограмму) и позволяет выбрать уровень разреза позже.	- Видно структуру данных и отношения между объектами - Не нужно заранее задавать число кластеров - Удобна для разведочного анализа	- Медленнее на больших данных - Чувствительна к выбору метрики и linkage - После склеивания назад уже не откатывается	Время: обычно $O(n^2) - O(n^3)$ Память: $O(n^2)$ (зависит от реализации)	- Небольшие и средние наборы данных - Когда важна дендрограмма - Когда хотим сначала изучить структуру
K-means 	Ищет k центров кластеров и многократно перераспределяет объекты к ближайшему центру.	- Быстрый и простой - Хорошо масштабируется - Понятный рабочий алгоритм для практики	- Нужно заранее выбрать k - Чувствителен к старту, масштабу и выбросам - Лучше работает для компактных, примерно круглых кластеров	Время: примерно $O(n \cdot k \cdot i \cdot d)$ Память: $O(n + k \cdot d)$	- Большие данные - Когда нужен быстрый базовый метод - Когда ожидаем компактные кластеры



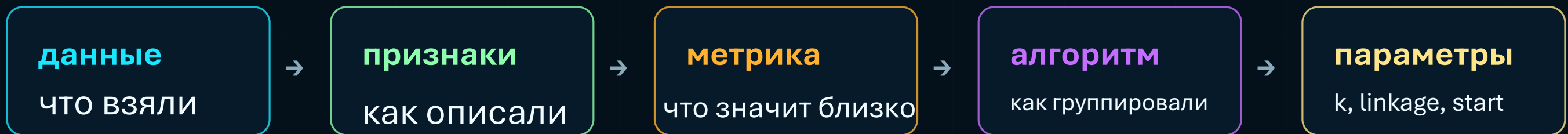
Совет преподавателя

Если вы хотите сначала понять структуру данных — начните с иерархической кластеризации. Если нужен быстрый рабочий результат на больших данных — часто разумно попробовать K-means.

Почему методы могут дать разные ответы?

не только алгоритм решает судьбу кластеров

Результат кластеризации = много решений до алгоритма



Если поменять одно звено, группы могут измениться.

Подводные места финала

что чаще всего ломает интерпретацию

Красивые цвета

не гарантируют содержательный смысл

Высокий silhouette

не доказывает истинные классы

Разные методы

не обязаны соглашаться

Плохие признаки

дают плохие кластеры даже хорошим алгоритмом

Слишком сильная история

мы можем придумать смысл там, где его нет

Совет от учителя

главное правило недели

**Не спрашивайте:
“какой алгоритм лучший?”**

**Спросите:
“почему эти объекты должны считаться похожими?”**

Похожесть — это решение аналитика.

Summary

Финальный Summary недели 10

Target нет

ищем структуру

Текст → числа

Bag of Words

Похожесть

метрика расстояния

Hierarchical

дерево отношений

K-means

группы вокруг центров

Silhouette

проверка посадки

Смысл найденных групп всё равно проверяет человек.

Вопрос

финальный вопрос преподавателя

?

Мы кластеризовали сказки и получили 4 красивые группы с высоким silhouette. Можно ли утверждать, что существуют ровно 4 объективных типа сказок?

Ответ должен быть не только “да/нет”, но и почему.

Ответ

финальный ответ



Нет.

Мы можем сказать только это:
при выбранных признаках, предобработке, метрике, алгоритме и
параметрах данные хорошо разделились на 4 группы.

А смысл этих групп надо читать, проверять и обсуждать.

DBSCAN и спектральная кластеризация

ещё два способа искать кластеры — без лишней магии

Кластеризация не заканчивается деревом и центрами.

Доктор Игорь Клейнер



DBSCAN: ищем плотные области

смотрим, где точки стоят плотно, а где почти пусто

Идея

если у точки много близких соседей, она живёт внутри плотной области

Не требует к

**кластеры рождаются из
плотных областей сами**

Где особенно хорош

изогнутые кластеры, шумные
выборки и данные с выбросами

Шум виден явно

одиночки не притворяются кластером, а
помечаются как noise

Где бывает трудно

если одна группа плотная, а другая разреженная,
один набор параметров может не угодить обеим

DBSCAN одним слайдом

как из плотности рождаются кластеры

ядра → соседи → рост кластера → шум

1

задали eps

→

2

нашли ядра

→

3

собрали соседей

→

4

нарастили
кластеры

→

5

отделили шум

Точки с достаточным числом близких соседей создают кластеры, остальные могут остаться шумом.



DBSCAN не красит всё подряд, он сначала спрашивает, где вообще есть жизнь

DBSCAN: плюсы и минусы

что полезно помнить

Плюсы

- не нужно заранее задавать число кластеров
- умеет находить шум и выбросы
- ловит кластеры сложной формы
- полезен для разведочного анализа

Минусы

- нужно подбирать `eps` и `min_samples`
- разные плотности могут сбить метод
- на больших данных может быть тяжёлым
- результат зависит от расстояния

Главная мысль

DBSCAN хорош, когда в данных есть плотные группы и настоящие “одиночки”.

Спектральная кластеризация: идея на пальцах

сначала строим карту связей, а потом ищем скрытую структуру

Строим граф

мы соединяем похожие объекты рёбрами и получаем граф похожести

Сильная сторона

после такого перехода легче увидеть хитрую форму кластеров

Где особенно хороша

сложные границы, графовые данные и случаи, где “круги” не работают

Что происходит внутри

мы переводим данные в новое пространство, где разделение видно лучше

Где бывает трудно

качество сильно зависит от графа похожести, а вычислительная цена выше

Спектральная кластеризация одним слайдом

как из графа похожести получают группы

граф → матрица → новое пространство → кластеры

1

построили граф

→

2

собрали матрицу

→

3

перешли в новое
пространство

→

4

сделали
кластеризацию

→

5

прочитали группы

Смысл в том, что в новом представлении скрытая структура данных часто становится заметнее.

граф похожести



новое пространство

Спектральная кластеризация: плюсы и минусы

что полезно помнить

Плюсы

- хорошо работает со сложной формой
- использует граф похожести
- помогает в трудных геометриях
- полезна, когда центры мешают

Минусы

- часто всё равно нужно выбрать k
- чувствительна к графу похожести
- требует больше памяти и вычислений
- объяснять сложнее, чем k -means

Главная мысль

Спектральная кластеризация полезна, когда важны связи между объектами, а не только центры.

Summary

короткая карта алгоритмов кластеризации

Hierarchical

дерево отношений

K-means

центры и заданное k

DBSCAN

плотность и шум

Spectral

граф похожести

Метрика

задаёт смысл близости

Проверка

человек проверяет смысл

Нет универсального победителя: метод выбирают под данные, метрику и цель анализа.