МИНОБРНАУКИ РОССИИ

РГУ НЕФТИ И ГАЗА (НИУ) ИМЕНИ И.М. ГУБКИНА

Факультет **Автоматики и вычислительной техники**
Кафедра **Автоматизированных систем управления**

Оценка комиссии: _____ Рейтинг: _____
Подписи членов комиссии:

_____ _____
(подпись) (фамилия, имя, отчество)

_____ _____
(подпись) (фамилия, имя, отчество)

_____
(дата)

# КУРСОВАЯ РАБОТА

по дисциплине Методы и модели искусственного интеллекта в задачах нефтегазового комплекса

на тему Применение методов кластеризации для анализа мировой торговли нефтью и нефтепродуктами

«К ЗАЩИТЕ»

Профессор, д. э. н. Алетдинова А.А.
(должность, ученая степень; фамилия, и.о.)

_____
(подпись)

_____
(дата)

ВЫПОЛНИЛ:
Студент группы **АСМ-22-05**
(номер группы)

Матвеев Роман Вячеславович
(фамилия, имя, отчество)

_____
(подпись)

_____
(дата)

Москва, 20 22

МИНОБРНАУКИ РОССИИ

РГУ НЕФТИ И ГАЗА (НИУ) ИМЕНИ И.М. ГУБКИНА

Факультет **Автоматики и вычислительной техники**
Кафедра **Автоматизированных систем управления**

# ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

по дисциплине Методы и модели искусственного интеллекта в задачах нефтегазового комплекса

на тему Применение методов кластеризации для анализа мировой торговли нефтью и нефтепродуктами

ДАНО студенту **Матвееву Роману Вячеславовичу** (фамилия, имя, отчество в дательном падеже) группы **АСМ-22-05** (номер группы)

**Содержание работы:**

1. Изучить методы искусственного интеллекта (в соответствии с темой курсовой работы) и дать их описание.
2. Выделить особенности использования методов искусственного интеллекта (в соответствии с темой курсовой работы) в задачах НГК
3. Показать практическую реализацию одной из задач с помощью изученного метода искусственного интеллекта.

**Исходные данные для выполнения работы:**

1. Требования к оформлению работы
2. Датасет «BP Statistical Review of World Energy June 2022»
3.

**Рекомендуемая литература:**


1. Григорьев Л.И., Санжаров В.В., Тупысев А.М. Интеллектуальный анализ данных; примеры нефтегазовой отрасли: Учебное пособие – М.: Издат. центр РГУ нефти и газа имени И. М. Губкина, 2015. – 121 с.
2. Дюран Б., Оделл П. Кластерный анализ //М.: статистика. – 1977. – Т. 128. – С. 2.
3. Бураков М. В. Системы искусственного интеллекта: учебное пособие. – Москва: Изд-во ООО «Проспект», 2019. – 440 с.


**Графическая часть:**

1. не предусмотрена.

Руководитель: д.э.н. (уч.степень) профессор (должность) _____ (подпись) Алетдинова А.А. (фамилия, имя, отчество)

Задание принял к исполнению: студент _____ (подпись) Матвеев Р.В. (фамилия, имя, отчество)

**СОДЕРЖАНИЕ**

# ВВЕДЕНИЕ

Данная курсовая работа посвящена проблеме применения метода k-means кластерного анализа для решения задачи распределения мировых стран по результатам торговли нефтью и нефтепродуктами за 2021 год и дальнейшего анализа полученных результатов.

**Актуальность работы.**

На сегодняшний день мировой рынок энергоресурсов не может похвастаться своей стабильностью. Санкционная политика западных стран непосредственным образом влияет на текущее состояние мирового рынка энергоносителей. Экономисты и аналитики всех стран мира в условиях нестабильности и изменчивости ситуации проводят глубокие аналитические исследования для того, чтобы иметь возможность прогнозировать изменения на мировом рынке и предпринимать определенные действия с целью минимизации отрицательных последствий на экономику своих стран.

Кластерный анализ результатов торговли нефтью и нефтепродуктами за прошедший год позволит разделить страны по группам, каждая из которых будет иметь свои отличительные особенности, характеризующие общие тенденции и направления ведения внешней торговли энергоносителями. Используя результаты данного анализа, можно будет выдвинуть некоторые предположения о реакции мирового рынка нефти и нефтепродуктов на изменения текущей политической обстановки в мире.

**Целью работы** является проведение кластерного анализа международной торговли нефтью и нефтепродуктами за 2021 год.

В процессе реализации поставленной цели необходимо рассмотреть и решить следующие задачи:

1. Анализ литературных источников по выбранной предметной области.
2. Ознакомление с общими сведениями о кластерном анализе, его трактовках и определениях в различных источниках.
3. Рассмотрение наиболее популярных методов кластерного анализа и применяемых метрик.

4. Анализ алгоритма k-means.
5. Ознакомление с перечнем задач нефтегазовой отрасли, в которых может быть применим кластерный анализ.
6. Поиск и выбор исходных данных, связанных с нефтегазовым комплексом, для осуществления кластерного анализа методом k-means.
7. Реализация метода кластеризации k-means на выбранном наборе исходных данных.
8. Визуализация, анализ и интерпретация полученных результатов.

# ГЛАВА 1

# ТЕОРЕТИЧЕСКИЕ ОСНОВЫ

## 1.1 Особенности использования кластерного анализа

Кластер (cluster – «гроздь», «скопление») – группа объектов, имеющих общие свойства.

Кластеризация – разбиение входных данных, состоящих из объектов на кластеры или же группы, обладающие некоторой однородностью. Характеристикой кластера является внутренняя однородность и внешняя изолированность. Задача кластеризации сводится к определению так называемых «сгущений точек». Цель кластеризации – поиск существующих структур [1].

Кластерный анализ является одним из основных и наиболее популярных методов интеллектуального анализа данных. Задача кластеризации относится к группе методов обучения без учителя, а также методов «описательного моделирования». Методы кластерного анализа осуществляют поиск «естественных» групп данных, они применимы как в научных исследованиях, так и исследованиях прикладного характера. Целью или решением задачи кластерного анализа является поиск такого разбиения исходных данных, которое удовлетворяет некоторому критерию оптимальности.

Кластеризация – это объединение в кластеры или сегменты наборов, состоящих из n объектов, основанное на подходящем под ту или иную ситуацию определении понятия близости объектов. Объекты одного кластера должны обладать высоким сходством характеристик между собой, объекты разных кластеров наоборот не должны отличаться высоким сходством. Кластерный анализ может рассматриваться как отдельная задача, так и промежуточная задача более глубокого анализа данных. После разбиения объектов на кластеры могут быть использованы другие методы интеллектуального анализа данных для определения параметров,

характеризующих каждый из кластеров, при помощи которых аналитик может установить, почему данные были сгруппированы именно таким образом [1].

Кластерный анализ был разработан в середине прошлого века, однако особую популярность заработал не так давно. Это можно связать в первую очередь с развитием таких направлений, как Big Data (большие данные) и Data Mining (сбор и анализ данных).

Анализ данных на основе кластеризации обладает рядом преимуществ. Во-первых, он позволяет группировать объекты не только по одному показателю, а по целому набору различных показателей. Во-вторых, это отсутствие каких-либо ограничений на виды и типы исследуемых объектов. Также кластерный анализ позволяет сжимать обширные массивы различной информации и делать их более показательными и репрезентативными. Может быть использован для прогнозирования данных и детектирования аномалий.

В машинном обучении или Machine Learning решаются такие задачи, как классификация и кластеризация. Необходимо определить особенности, отличающие кластеризацию от классификации. При разбиении на кластеры сами кластеры и их количество не определено. Задача кластеризации решается с использованием обучения без учителя, классификацию же относят к группе методов обучения с учителем, так как изначально известны характеристики каждого из классов [1].

### 1.2 Методы кластерного анализа

Кластеризация характеризуется обширным набором разнообразных методов, которые в свою очередь можно объединить в категории по особенностям выполнения. Выбор подходящей совокупности и конкретного алгоритма кластеризации зависит от особенностей данных, которые необходимо исследовать, и цели анализа. Это можно считать одной из причин существования такого большого количества разнообразных методов деления объектов на кластеры. На данный момент не существует такого алгоритма, который бы наилучшим образом работал на любых входных данных.

Объясняется это тем, что для каждого набора данных, конкретной ситуации принимаются отдельные допущения для использования того или иного метода кластеризации.

Методы кластерного анализа преимущественно подразделяются на следующие группы:

1. Методы разделения.

Входные данные состоят из n объектов. Методы данной группы подразделяют данные на k кластеров ($k \leq n$). При этом каждый кластер содержит один и более объектов, а каждый объект принадлежит строго одному кластеру. Алгоритмы выполняются следующим образом: осуществляется исходное разбиение на k кластеров, далее на каждой последующей итерации это разбиение улучшается, путем перераспределения объектов по кластерам. Оптимальность кластеризации достигается в одних методах путем определения каждого кластера как среднего значения параметров объектов, входящих в него, в других путем определения кластера объектом, расположенным наиболее близко к центру кластера.

Методы данной категории успешно осуществляют разделение объектов на кластеры в форме сфер на небольших и средних объемах входных данных. Для работы с данными большей размерности необходимо использование усовершенствованных методов разделения.

2. Иерархические методы.

Входные данные декомпозируются и создается некоторая иерархия. Данная группа методов подразделяется на подгруппы: агломеративные (агломерационные) и дивизионные.

Алгоритмы из первой подгруппы называются также методами снизу-вверх, то есть при первом разбиении каждый объект принадлежит отдельному кластеру, затем происходит их слияние, основанное на близости объектов. Такое объединение продолжается до тех пор, пока не будет получен один

кластер, являющийся верхним уровнем иерархии, либо пока не будет выполнено заданное условие остановки алгоритма.

Дивизионные методы основываются на подходе сверху-вниз, то есть изначально объекты принадлежат одному кластеру, далее итеративно происходит разбиение на отдельные группы, пока каждый объект не будет состоять в отдельном кластере или же пока не выполнено условие остановки алгоритма.

Отличительной особенностью данных методов является невозможность изменить выполненное разбиение или объединение. Такое ограничение способствует меньшим вычислительным нагрузкам, так как нет возможности осуществить уже выполненное разбиение или объединение вновь. Данные алгоритмы могут быть использованы в комплексе с другими методами кластеризации, что приводит к повышению качества выполнения задач [1].

3. Методы на основе плотности.

Методы данной группы могут формировать кластеры не только сферических форм, но и произвольных. Также такие методы не чувствительны ни к выбросам, ни к шуму.

В основе данных методов лежит не вычисление расстояния между объектами, как во многих других алгоритмах, а определение плотности. Под плотностью здесь подразумевается количество объектов в их окрестностях. То есть задается определенное пороговое значение для плотности каждого кластера. Кластер «разрастается», пока значения плотностей соседних кластеров превышает заданный порог [1].

4. Сеточные методы.

Данные алгоритмы основываются на квантовании пространства признаков гиперплоскостями или на конечное число клеток. Объекты, которые попадают в одну ячейку, объединяются в кластер. Данные методы характеризуются высокой скоростью обработки данных, так как она зависит только от числа ячеек в полученной сетке пространства [2].

5. Методы на основе моделей.

В этих методах изначально предлагается некоторая модель для каждого из кластеров. Обнаружение кластеров происходит при построении функции плотности, отражающей распределение объектов в пространстве. Данные методы не чувствительны к выбросам и шуму [1].

### 1.3 Используемые метрики

Метрика или мера расстояния во многих методах кластеризации используется для определения близости или сходства объектов. Функция расстояния (метрика) d(i, j) является неотрицательной и вещественнозначной функцией [3]:

- Расстояние является симметричной функцией;
- Расстояние не может быть отрицательным;
- Расстояние от объекта до самого себя равно нулю;
- Расстояние от одного объекта до другого не больше, чем расстояние от этого же объекта до другого через объект-посредник.

Наиболее популярными метриками являются Евклидово расстояние, Манхэттенское расстояние или City-Block, расстояние Минковского и расстояние Чебышева. Необходимо определить особенности каждой из этих метрик.

*Евклидово расстояние:*

$$d(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots + (x_{in} - x_{jn})^2}, \qquad (1)$$

где $i = (x_{i1}, x_{i2}, \dots, x_{in})$ и $j = (x_{j1}, x_{j2}, \dots, x_{jn})$ являются двумя объектами размерности n.

Данную метрику используют, если близость объектов для разбиения на кластеры есть близость геометрическая. Составляющие X отличаются своей однородностью как в физическом смысле, так и по важности для разделения на группы. Объекты X являются частью одной генеральной совокупности и взаимно независимы друг от друга.

*Манхэттенское расстояние или $L_1$ норма:*

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \cdots + |x_{in} - x_{jn}|, \quad (2)$$

Другими словами, данная метрика – это среднее разностей по координатам. Она менее чувствительная к выбросам, благодаря отсутствию возведения координат в квадрат.

*Расстояние Минковского или $L_p$ норма:*

$$d(i,j) = (|x_{i1} - x_{j1}|^p + |x_{i2} - x_{j2}|^p + \cdots + |x_{in} - x_{jn}|^p)^{1/p}, \quad (3)$$

где $p \in Z$, $p > 0$.

Данная метрика является обобщением Евклидова ($p = 2$) и Манхэттенского расстояния ($p = 1$).

*Расстояние Чебышева:*

$$d(X_i, X_j) = \max_{1 \le k \le p} |x_{ik} - x_{jk}|, \quad (4)$$

Данная метрика используется в том случае, если объекты, которые принадлежат различным кластерам, отличаются друг от друга по одной из составляющих X [1].

Также популярными метриками являются метрика несхожести Хеминга, «норма – верхняя граница», функция Махаланобиса, функция Джеффриса-Матуситы и «коэффициент дивергенции» [4].

### 1.4 Алгоритм k-means

Метод k-средних или метод Маккуина относят к группе методов разделения. В основе данного алгоритма лежит разбиение n объектов по k кластерам. В самом начале случайным образом выбираются k объектов, которые становятся начальными центрами кластеризации. Объекты объединяются в кластеры ближайшие по среднему значению. То есть задается некоторое предельное расстояние r, если расстояние от объекта до центра того или иного кластера меньше этого предельного значения, то объект добавляется в этот кластер, если же расстояние больше r, то объект образует новый кластер. При образовании нового кластера центры кластеров

пересчитываются. Кластеры, находящиеся друг от друга на расстоянии меньше заданного уровня, то такие кластеры объединяются в один. Алгоритм продолжается до выполнения условия сходимости [3].

Входными данными является исходное множество объектов ($x_1$, $x_2$, …, $x_n$), метод k-средних разбивает n объектов по k кластерам K = {$K_1$, $K_2$, …, $K_k$} таким образом, что сумма квадратов расстояний в каждом кластере стремилась к минимуму:

$$arg_K \min \sum_{i=1}^{k} \sum_{x_j \in K_i} (x_j - \mu_i)^2, \quad (5)$$

Данный алгоритм является NP-трудным, его сложность по времени составляет $O(n^{dk+1} \log (n))$.

Шаги алгоритма:

1. Аналитик задает предполагаемое число кластеров k.
2. Из исходных данных случайным образом выбираются k объектов, которые становятся центрами кластеров на начальном этапе.
3. Объекты распределяются по кластерам на основании Евклидова расстояния, объект попадает в тот кластер, центр которого ближе всего к нему.
4. Определяются центроиды или центры тяжести каждого кластера. Центроида представляется собой вектор, содержащий средние значения каждого параметра или признака кластера.
5. Центр кластера перемещается в центроид, таким образом центроид становится новым центром кластера.
6. Шаги 3 и 4 повторяются итеративно. Условием остановки алгоритма является отсутствие изменения координат центроид и границ кластеров.

К преимуществам данного метода относят простоту реализации и скорость его выполнения. Данный алгоритм является одним из или даже самым популярным методом кластеризации, следовательно, существует множество библиотек на различных языках программирования, содержащих его реализацию в качестве встроенных функций.

## 1.5 Применение алгоритмов кластеризации для задач нефтегазового комплекса

При поиске литературных источников по выбранной тематике были проанализированы несколько работ по кластерному анализу в нефтегазовом комплексе. Одна из таких работ была связана с индустриальным менеджментом, тема работы звучала следующим образом: «Кластерный анализ компаний нефтяной промышленности по параметрам налоговой нагрузки». При осуществлении кластерного анализа были использованы данные о компаниях нефтяной промышленности, которых нет в открытом доступе. Благодаря наличию такого набора исходных данных, было успешно произведено разделение компаний на три кластера по налоговой нагрузке на них [5].

В рамках данной работы кластеризация может быть произведена с целью разбиения объектов на группы и дальнейшего анализа результатов такого разделения. В перспективе возможен и более глубокий анализ с использованием различных методов и алгоритмов интеллектуального анализа данных или Data Mining.

В качестве исходных данных в свободном доступе в сети Интернет был найден отчет компании «BP» под названием «BP Statistical Review of World Energy June 2022» [6]. Данный файл состоит из нескольких десятков страниц, посвященных всевозможной информации об производстве и использовании различных энергетических ресурсов по всему миру. Для данной работы был выбран датасет под названием «Oil trade in 2020 and 2021 (Торговля нефтью в 2020 и 2021 годах)». Датасет состоит из данных об импорте и экспорте как сырой нефти, так и нефтепродуктов всех стран мира, значения которых отличны от нуля и имеют значимость в общемировой статистике (рис. 1).

**Oil trade in 2020 and 2021**

| | 2020 | | | | 2021 | | | |
|---|---|---|---|---|---|---|---|---|
| Million tonnes | Crude Imports | Product Imports | Crude Exports | Product Exports | Crude Imports | Product Imports | Crude Exports | Product Exports |
| Canada | 29,2 | 28,8 | 189,0 | 30,7 | 23,9 | 30,6 | 197,4 | 33,5 |
| Mexico | † | 54,5 | 56,7 | 5,5 | 0,0 | 59,0 | 52,9 | 8,2 |
| US | 293,7 | 95,4 | 159,5 | 236,6 | 304,7 | 112,9 | 138,5 | 244,4 |
| S. & Cent. America | 17,6 | 91,1 | 146,2 | 25,1 | 21,8 | 105,8 | 124,1 | 23,6 |
| Europe | 479,3 | 173,0 | 28,3 | 103,9 | 467,7 | 197,5 | 36,4 | 110,5 |
| Russia | † | 1,2 | 264,7 | 118,9 | † | 1,9 | 263,6 | 140,7 |
| Other CIS | 15,5 | 2,3 | 92,5 | 11,3 | 15,9 | 6,9 | 87,1 | 17,7 |
| Iraq | † | 6,3 | 177,4 | 17,3 | † | 8,3 | 176,1 | 12,3 |
| Kuwait | † | 0,7 | 96,9 | 16,6 | † | 0,9 | 88,4 | 24,3 |
| Saudi Arabia | 0,1 | 13,8 | 349,2 | 43,7 | † | 16,1 | 323,2 | 57,7 |
| United Arab Emirates | 11,0 | 30,4 | 138,7 | 63,4 | 3,2 | 31,8 | 146,1 | 86,7 |
| Other Middle East | 21,9 | 17,6 | 100,8 | 58,3 | 18,7 | 19,7 | 97,0 | 62,4 |
| North Africa | 12,1 | 28,2 | 50,9 | 39,6 | 9,3 | 30,8 | 85,4 | 45,4 |
| West Africa | 0,5 | 37,9 | 201,8 | 8,7 | 0,5 | 46,0 | 187,4 | 8,6 |
| East & S. Africa | 16,3 | 36,4 | 3,8 | 5,0 | 12,4 | 41,1 | 4,8 | 2,7 |
| Australasia | 18,6 | 22,3 | 8,7 | 5,8 | 14,9 | 26,2 | 9,2 | 5,4 |
| China | 557,2 | 85,1 | 0,8 | 58,3 | 526,0 | 103,4 | 1,6 | 60,6 |
| India | 196,9 | 45,8 | 0,1 | 53,3 | 213,7 | 49,4 | 0,1 | 69,3 |
| Japan | 123,5 | 40,1 | † | 10,1 | 122,1 | 43,0 | 0,4 | 11,0 |
| Singapore | 45,8 | 96,0 | 1,7 | 70,0 | 47,0 | 91,8 | 1,0 | 68,9 |
| Other Asia Pacific | 262,7 | 187,7 | 34,3 | 112,8 | 257,1 | 202,1 | 38,2 | 131,2 |
| **Total World** | **2101,8** | **1094,7** | **2101,8** | **1094,7** | **2058,9** | **1225,2** | **2058,9** | **1225,2** |

Рис. 1 Датасет с исходными данными для анализа

Выполним кластеризацию для данного датасета и проведем анализ полученных результатов.

# ГЛАВА 2

# РЕАЛИЗАЦИЯ ПРОГРАММЫ ДЛЯ КЛАСТЕРНОГО АНАЛИЗА

## 2.1 Выбор языка программирования и среды разработки

Языком программирования для создания программы был выбран Python. На сегодняшний день данный язык программирования является одним из самых популярных для решения задач, связанных с искусственным интеллектом, машинным обучением и нейронными сетями.

Python – это высокоуровневый язык программирования с динамической строгой типизацией и автоматическим управлением памятью, поддерживающий множество парадигм и оптимизированный для обеспечения высокой продуктивности программистов, читабельности кода и качества ПО. Данный ЯП пользуется большой популярностью у людей, связанных с IT-сферой, так как он сочетает в себе универсальность, относительную простоту и мощный инструментарий для решения многих нетривиальных задач.

К преимуществам использования Python относят: качество ПО, продуктивность труда, переносимость программ, поддержку библиотек и интеграции компонентов.

Python является популярным ЯП, следовательно, существует множество различных сред разработки и редакторов, поддерживающих его. Среда разработки должна отвечать следующим требованиям: комфортность, удобство и эффективность.

Google Colaboratory или Colab – бесплатная платформа для запуска блокнотов Python. Его функционала будет вполне достаточно для решения задачи, поставленной в рамках данной работы. А возможность разделения кода на отдельные блоки является несомненным преимуществом использования Colab.

## 2.2 Реализация алгоритма k-means в Google Colab

Для начала работы необходимо открыть веб-сайт Google Colaboratory: https://colab.research.google.com/. Затем необходимо пройти авторизацию в свой аккаунт Google и создать блокнот (рис. 2).

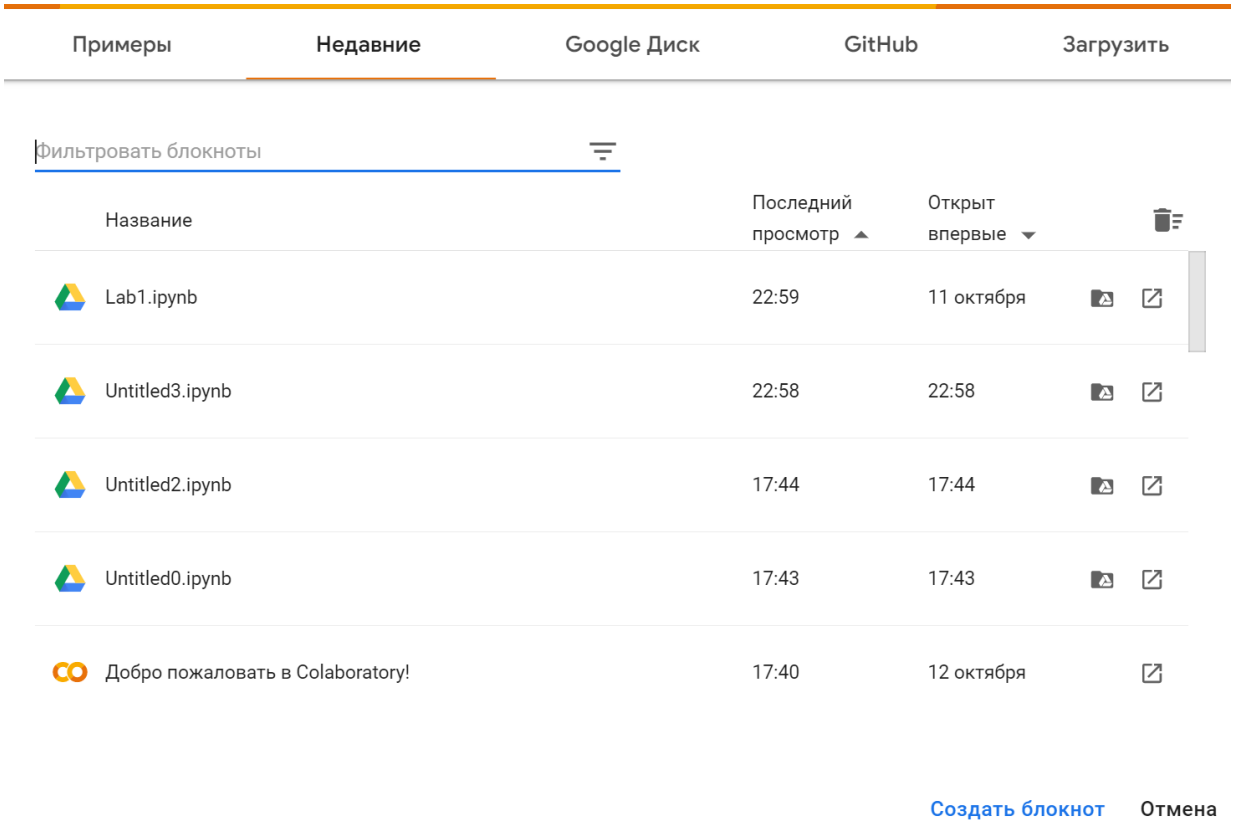


Рис. 2 Создание нового блокнота в Google Colab

Далее необходимо подключиться к удаленной среде разработки и начинать реализовывать алгоритм k-means для кластерного анализа (рис. 3).

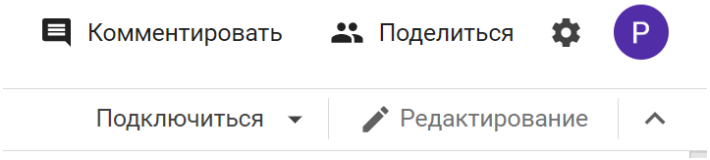


Рис. 3 Подключение к удаленной среде разработки

Подключаем все необходимые для работы библиотеки (рис. 4).

```
[154] import numpy as np
      from sklearn import datasets
      import pandas as pd
      import matplotlib.pyplot as plt
      from sklearn.cluster import KMeans
      from sklearn.metrics import silhouette_score
```

Рис. 4 Подключение библиотек

Загружаем выбранный датасет (рис. 5) и выведем его на экран (рис. 6).

```
[4]  df = pd.read_csv('oil trades.csv', sep=';')
     df
```

Рис. 5 Загрузка csv-файла

| | Countries | Crude Imports | Product Imports | Crude Exports | Product Exports |
|---|---|---|---|---|---|
| 0 | Canada | 23.909 | 30.636 | 197.439 | 33.542 |
| 1 | Mexico | 0.000 | 58.959 | 52.884 | 8.238 |
| 2 | US | 304.670 | 112.861 | 138.549 | 244.436 |
| 3 | S. & Cent. America | 21.826 | 105.767 | 124.147 | 23.639 |
| 4 | Europe | 467.741 | 197.500 | 36.379 | 110.523 |
| 5 | Russia | 0.024 | 1.875 | 263.565 | 140.670 |
| 6 | Other CIS | 15.941 | 6.889 | 87.121 | 17.713 |
| 7 | Iraq | 0.005 | 8.333 | 176.096 | 12.305 |
| 8 | Kuwait | 0.000 | 0.936 | 88.362 | 24.259 |
| 9 | Saudi Arabia | 0.013 | 16.142 | 323.215 | 57.653 |
| 10 | United Arab Emirates | 3.163 | 31.788 | 146.072 | 86.732 |
| 11 | Other Middle East | 18.665 | 19.664 | 96.967 | 62.436 |
| 12 | North Africa | 9.296 | 30.848 | 85.421 | 45.375 |
| 13 | West Africa | 0.471 | 46.032 | 187.365 | 8.555 |
| 14 | East & S. Africa | 12.368 | 41.093 | 4.827 | 2.747 |
| 15 | Australasia | 14.888 | 26.186 | 9.228 | 5.368 |
| 16 | China | 525.961 | 103.408 | 1.566 | 60.585 |
| 17 | India | 213.747 | 49.370 | 0.053 | 69.339 |
| 18 | Japan | 122.050 | 43.024 | 0.411 | 10.958 |
| 19 | Singapore | 47.017 | 91.829 | 1.009 | 68.932 |
| 20 | Other Asia Pacific | 257.100 | 202.057 | 38.180 | 131.191 |

Рис. 6 Исходные данные для анализа

Временно удаляем из массива столбец с названиями стран (рис. 7).

```
[5]  data = df.drop("Countries", axis=1)
     counries = df["Countries"].values
     data = data.astype(np.float)
     data.head()
```

| | Crude Imports | Product Imports | Crude Exports | Product Exports |
|---|---|---|---|---|
| 0 | 23.909 | 30.636 | 197.439 | 33.542 |
| 1 | 0.000 | 58.959 | 52.884 | 8.238 |
| 2 | 304.670 | 112.861 | 138.549 | 244.436 |
| 3 | 21.826 | 105.767 | 124.147 | 23.639 |
| 4 | 467.741 | 197.500 | 36.379 | 110.523 |

Рис. 7 Удаление поля "Название страны"

Запишем массив в переменную X, а также используем функцию nan_to_num из библиотеки NumPy для того, чтобы все значения NaN в датасете, если таковые имеются, были преобразованы в числа (рис. 8).

```
X = data.values[:,:]
X = np.nan_to_num(X)
```

Рис. 8 Функция np.nan_to_num()

Алгоритм k-means в качестве входных данных принимает сам датасет для кластеризации и количество кластеров, таким образом аналитик должен либо определить это число методом подбора, либо использовать метод «локтя» (The Elbow Method) или метод силуэта (The Silhouette Method). В рамках данной работы используем оба метода и сравним результаты.

Метод «локтя» – метод, позволяющий определить правильно количество кластеров k для разбиения датасета, тем самым повышая производительность реализуемой модели. Данный метод относят к эмпирическим, так как он для определения количества кластеров вычисляет сумму квадратов расстояний между объектами и вычисляется среднее значение. Затем вычисленные значения наносятся на график, по которому можно определить оптимальное количество кластеров.

Отдельной функции на Python для этого метода нет, поэтому необходимо выполнить последовательно кластеризацию по интервалу значений k от единицы до десяти. Метод KMeans из библиотеки sklearn возвращается значение суммы квадратов расстояний в виде атрибута inertia_, все эти значения запишем в вектор distortions (рис. 9).

```
distortions = []
K = range(1,10)
for k in K:
    kmeanModel = KMeans(n_clusters=k)
    kmeanModel.fit(X)
    distortions.append(kmeanModel.inertia_)
```

Рис. 9 Вычисление значений inertia_

Затем построим график зависимости суммы квадратов расстояний от количества кластеров, по которому и определим оптимальное количество кластеров (рис. 10).

```
plt.figure(figsize=(16,8))
plt.plot(K, distortions, 'bx-')
plt.xlabel('k')
plt.ylabel('Distortion')
plt.title('The Elbow Method showing the optimal k')
plt.show()
```

Рис. 10 Построение графика зависимости distortions(k)

На графике искомое значение k должно вызывать резкое изменение поведения кривой, в нашем случае четко определить оптимальное значение k не получается, оно может лежать в интервале [2; 4] (рис. 11).

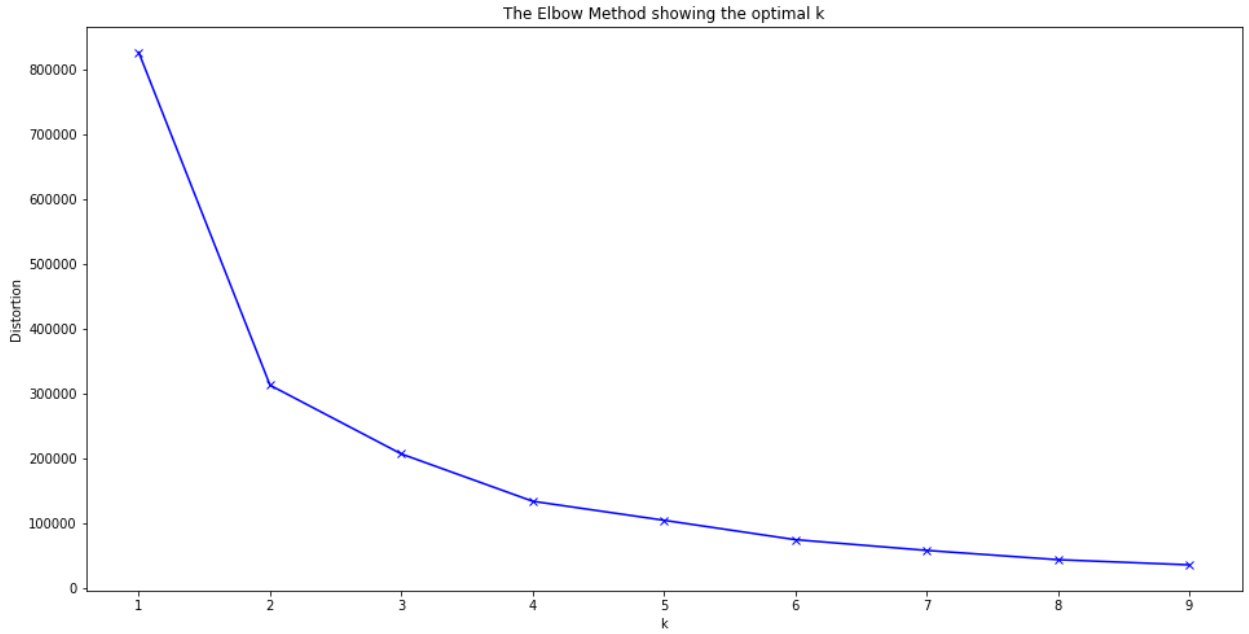


Рис. 11 График distortions(k)

Реализуем метод силуэта, он должен прояснить ситуацию и определить точное значение k. Суть данного метода заключается в вычислении среднего расстояния в текущем кластере и среднего расстояния от объекта до ближайшего кластера. На основании полученных данных вычисляется так называемая метрика силуэта. Ее значения лежат в интервале [-1; 1], где значение -1 является наихудшим, соответственно, 1 – наилучшим.

Вновь необходимо выполнить кластеризацию по интервалу значений k, затем использовать функцию silhouette_score из библиотеки sklearn (рис. 12).

```
K = range(3,5)
for k in K:
  km = KMeans(init = "random", n_clusters=k, n_init = 12)
  km.fit_predict(X)
  score = silhouette_score(X, km.labels_, metric='euclidean')

  print('Silhouetter Score for k = %d: %.3f' % (k, score))
```

Рис. 12 Реализация метода силуэт

Результат выполнения метода силуэт:

```
Silhouetter Score for k = 3: 0.381
Silhouetter Score for k = 4: 0.351
```

При помощи данного метода мы определили оптимальное значение k, которое равняется трём.

Далее выполним кластеризацию, реализованную в библиотеке sklearn, создав объект класса KMeans() и передав в конструктор следующие параметры: метод инициализации, k = 3 и количество раз выполнения алгоритма с различными начальными значениями центроид. Далее для созданного объекта k_means вызовем его метод fit(), в который в качестве параметра передадим данные. В переменную labels запишем результат кластеризации, а в centers – координаты центра каждого кластера (рис. 13).

```
clusterNum = 3
k_means = KMeans(init = "random", n_clusters=clusterNum, n_init = 12)
k_means.fit(X)
labels = k_means.labels_
centers = k_means.cluster_centers_
```

Рис. 13 Кластеризация Sklearn

Вернем в исходную таблицу столбец с названиями стран и добавим новый – с результатами кластеризации (рис. 14), выведем полученную таблицу (рис. 15).

```
data["Countries"] = counries
data["Clus_km"] = labels
data
```

Рис. 14 Работа со столбцами таблицы

| | Crude Imports | Product Imports | Crude Exports | Product Exports | Countries | Clus_km |
|---|---|---|---|---|---|---|
| **0** | 23.909 | 30.636 | 197.439 | 33.542 | Canada | 1 |
| **1** | 0.000 | 58.959 | 52.884 | 8.238 | Mexico | 2 |
| **2** | 304.670 | 112.861 | 138.549 | 244.436 | US | 0 |
| **3** | 21.826 | 105.767 | 124.147 | 23.639 | S. & Cent. America | 2 |
| **4** | 467.741 | 197.500 | 36.379 | 110.523 | Europe | 0 |
| **5** | 0.024 | 1.875 | 263.565 | 140.670 | Russia | 1 |
| **6** | 15.941 | 6.889 | 87.121 | 17.713 | Other CIS | 2 |
| **7** | 0.005 | 8.333 | 176.096 | 12.305 | Iraq | 1 |
| **8** | 0.000 | 0.936 | 88.362 | 24.259 | Kuwait | 2 |
| **9** | 0.013 | 16.142 | 323.215 | 57.653 | Saudi Arabia | 1 |
| **10** | 3.163 | 31.788 | 146.072 | 86.732 | United Arab Emirates | 1 |
| **11** | 18.665 | 19.664 | 96.967 | 62.436 | Other Middle East | 2 |
| **12** | 9.296 | 30.848 | 85.421 | 45.375 | North Africa | 2 |
| **13** | 0.471 | 46.032 | 187.365 | 8.555 | West Africa | 1 |
| **14** | 12.368 | 41.093 | 4.827 | 2.747 | East & S. Africa | 2 |
| **15** | 14.888 | 26.186 | 9.228 | 5.368 | Australasia | 2 |
| **16** | 525.961 | 103.408 | 1.566 | 60.585 | China | 0 |
| **17** | 213.747 | 49.370 | 0.053 | 69.339 | India | 2 |
| **18** | 122.050 | 43.024 | 0.411 | 10.958 | Japan | 2 |
| **19** | 47.017 | 91.829 | 1.009 | 68.932 | Singapore | 2 |
| **20** | 257.100 | 202.057 | 38.180 | 131.191 | Other Asia Pacific | 0 |

Рис. 15 Полученная таблица

Выпишем названия стран в соответствии с распределением по кластерам (рис. 16).

```
cluster_0 = []
cluster_1 = []
cluster_2 = []

for i in data.values:
  n = i[len(data.values[0,:])-1]
  if n == 0:
    cluster_0.append(i[-2])
  elif n == 1:
    cluster_1.append(i[-2])
  elif n == 2:
    cluster_2.append(i[-2])
print("Cluster 0 ", cluster_0, "\n","Cluster 1 ",cluster_1,"\n","Cluster 2 ", cluster_2, "\n")
```

```
Cluster 0  ['US', 'Europe ', 'China', 'Other Asia Pacific']
 Cluster 1  ['Canada', 'Russia', 'Iraq', 'Saudi Arabia', 'United Arab Emirates', 'West Africa']
 Cluster 2  ['Mexico', 'S. & Cent. America', 'Other CIS', 'Kuwait', 'Other Middle East', 'North Africa', 'East & S. Africa', 'Australasia', 'India', 'Japan', 'Singapore']
```

Рис. 16 Результат кластеризации

Визуализируем полученный результат при помощи точечной диаграммы в двумерном и трехмерном видах (рис. 17-18).

```
plt.figure(figsize=(8,6))
plt.scatter(X[:, 2], X[:, 3], c=labels.astype(np.float), alpha=1)
plt.xlabel("Crude Exports")
plt.ylabel("Product Exports")
plt.scatter(centers[:, 2], centers[:, 3], marker='x', c="r")
plt.show()
```

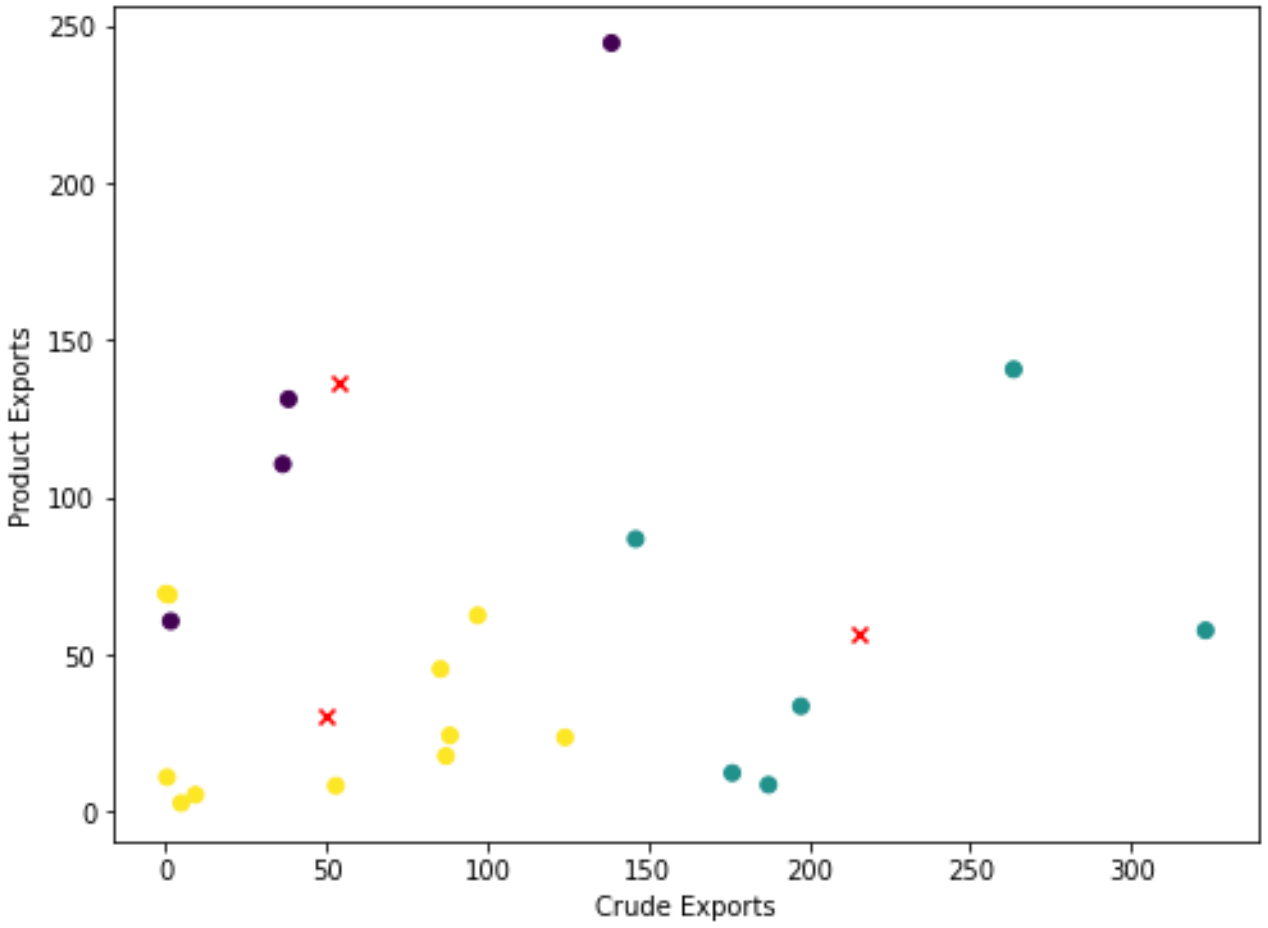


Рис. 17 Визуализация в 2D

```
from mpl_toolkits.mplot3d import Axes3D
fig = plt.figure(1, figsize=(8, 6))
plt.clf
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=134)

ax.set_xlabel("Crude Imports")
ax.set_ylabel("Crude Exports")
ax.set_zlabel("Product Exports")

ax.scatter(X[:, 0], X[:, 2], X[:, 1], c=labels.astype(np.float))
```

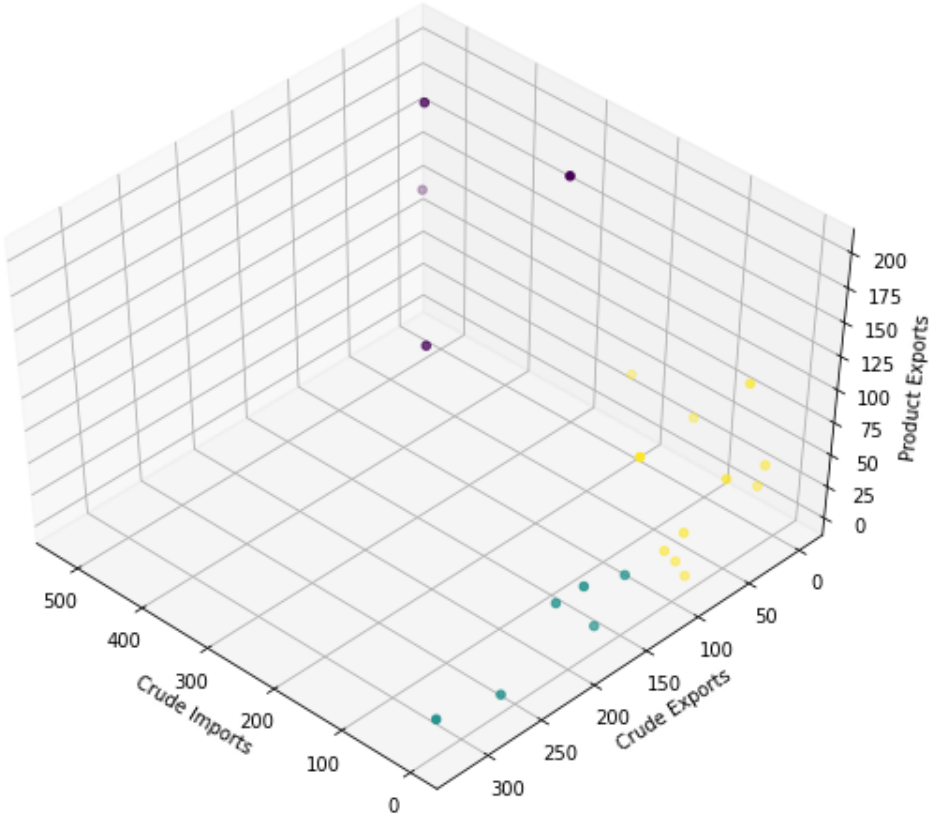


Рис. 18 Визуализация в 3D

Вычислим средние значения импорта и экспорта нефти и нефтепродуктов по каждому кластеру (рис. 19).

```
data.groupby('Clus_km').mean()
```

| Clus_km | Crude Imports | Product Imports | Crude Exports | Product Exports |
|---|---|---|---|---|
| 0 | 388.868000 | 153.956500 | 53.668500 | 136.683750 |
| 1 | 4.597500 | 22.467667 | 215.625333 | 56.576167 |
| 2 | 43.254364 | 43.142273 | 50.039091 | 30.818545 |

Рис. 19 Средние значения по кластерам

Визуализируем значения, характеризующие каждый из кластеров, при помощи графика, по оси абсцисс которого заданы параметры кластеров

(значения импорта и экспорта нефти и нефтепродуктов), по оси ординат значения в натуральном выражении (в миллионах тонн). Кривые, отражающие значения показателей кластеров, представлены разными цветами для наглядности (рис. 20).

```
plt.figure(figsize=(10, 10))
plt.ylabel("Million tonnes")
clusters_arr = np.arange(0, clusterNum, 1)
for i in clusters_arr:
  plt.plot(data.groupby('Clus_km').mean().iloc[i, :], label=i)
plt.grid(True)
plt.legend(loc='best', fontsize=12)
```

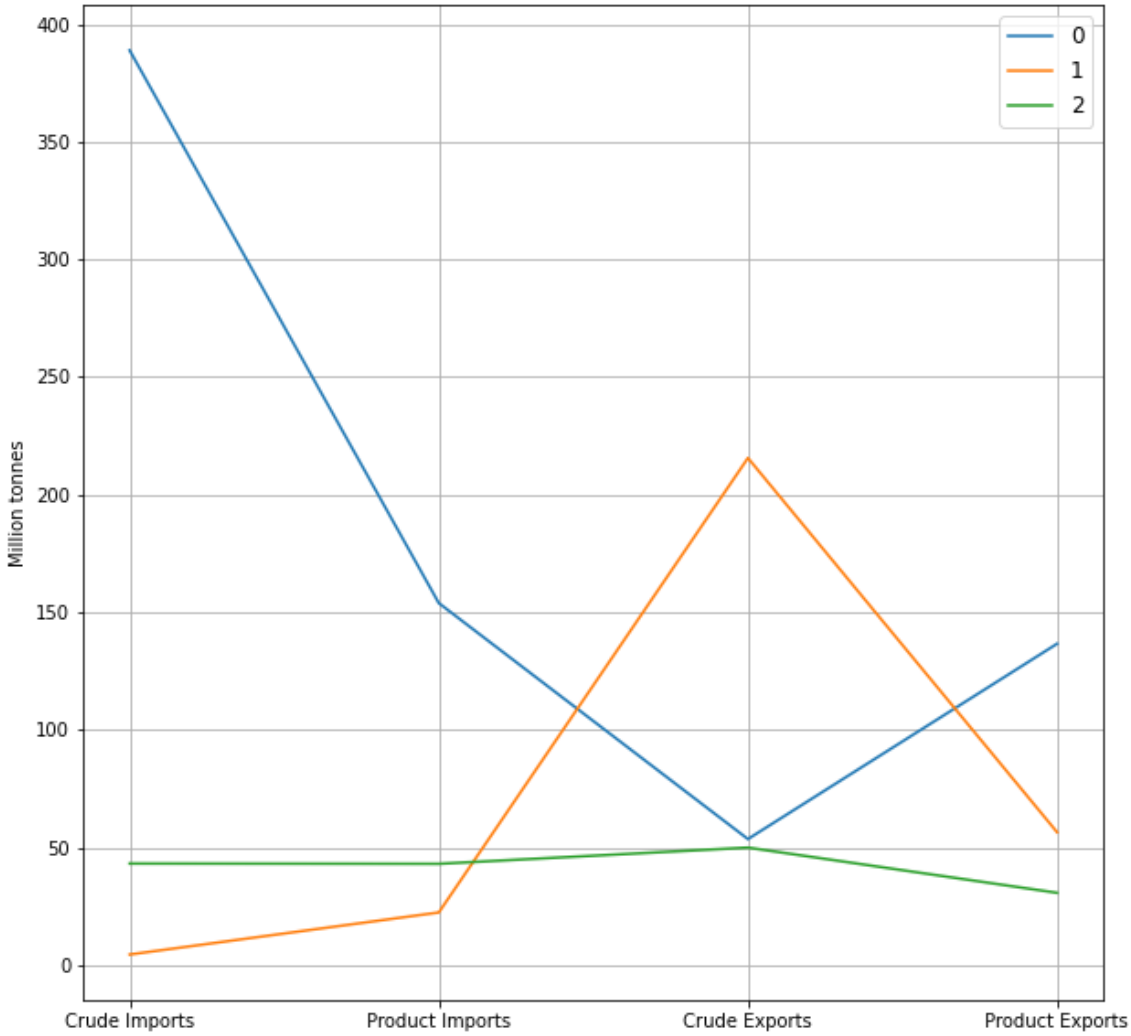


Рис. 20 Визуализация средних значений параметров по кластерам

Также метод k-means, реализован в библиотеке SciPy, выполним кластеризацию при помощи него и сравним результаты разбиения.

Импортируем из библиотеки SciPy функции kmeans и vq (рис. 21).

```
from scipy.cluster.vq import kmeans,vq
```

Рис. 21 Импорт SciPy

Передадим в метод kmeans() исходные данные и число кластеров, получим координаты центров кластеров и расстояния от каждого объекта до центра ближайшего кластера. Затем передадим в функцию vq() данные и координаты центров для присвоения каждой стране своего номера кластера (рис. 22).

```
clusterNum = 3
centers_scipy, distortion = kmeans(X, k_or_guess=clusterNum)
labels_scipy,_ = vq(X,centers_scipy)
```

Рис. 22 Кластеризация SciPy

Выпишем названия стран в соответствии с распределением по кластерам (рис. 23).

```
cluster_0 = []
cluster_1 = []
cluster_2 = []

for i in data.values:
  n = i[len(data.values[0,:])-1]
  if n == 0:
    cluster_0.append(i[-2])
  elif n == 1:
    cluster_1.append(i[-2])
  elif n == 2:
    cluster_2.append(i[-2])
print(" Cluster 0 ", cluster_0, "\n","Cluster 1 ",cluster_1,"\n","Cluster 2 ", cluster_2, "\n")
```

```
Cluster 0  ['US', 'Europe ', 'China', 'India', 'Other Asia Pacific']
Cluster 1  ['Canada', 'Russia', 'Iraq', 'Saudi Arabia', 'United Arab Emirates', 'West Africa']
Cluster 2  ['Mexico', 'S. & Cent. America', 'Other CIS', 'Kuwait', 'Other Middle East', 'North Africa', 'East & S. Africa', 'Australasia', 'Japan', 'Singapore']
```

Рис. 23 Результат кластеризации

Если сравнивать кластеризацию, выполненную при помощи двух различных библиотек, можно заметить лишь одно отличие: Индия оказалась в кластере №0, в то время как sklearn отнес данную страну в кластер №2. Если оценить показатели Индии, то они окажутся близкими по значению к показателям Японии, которая распределена в кластер №2. На основании этого примем за истину кластеризацию от sklearn и проанализируем именно этот результат.

### 2.3 Анализ и интерпретация полученных результатов

В результате проведения кластерного анализа было получено три кластера, в кластер №0 вошли такие страны, как US, Europe, China, Other Asia

Pacific, в кластер №1 – Canada, Russia, Iraq, Saudi Arabia, United Arab Emirates, West Africa, в кластер №2 – Mexico, S. & Cent. America, Other CIS, Kuwait, Other Middle East, North Africa, East & S. Africa, Australasia, India, Japan, Singapore.

Кластер №0 характеризуется следующими значениями параметров: крайне высокий импорт сырой нефти, высокий импорт нефтепродуктов, низкий экспорт сырой нефти и высокий экспорт нефтепродуктов.

Кластер №1 – крайне низкие показатели импорта нефти, чуть большие, но все же крайне низкие показатели импорта нефтепродуктов, крайне высокие показатели экспорта сырой нефти и низкие показатели экспорта нефтепродуктов.

Кластер № 2 – низкие показатели импорта сырой нефти, практически аналогично низкие показатели импорта нефтепродуктов, чуть большие, но все же низкие показатели экспорта сырой нефти и крайне низкие показатели экспорта нефтепродуктов. Низкие значения всех параметров могут говорить о том, что эти страны потребляют столько энергоресурсов, сколько и производят.

Страны кластера №1, в числе которых и Россия, являются главными экспортерами сырой нефти, главными же потребителями – страны кластера №0, среди которых Китай, США, страны Европы и Азии. При этом страны кластера №0 можно назвать главными импортерами и экспортерами нефтепродуктов (рис. 24).

```
data.groupby('Clus_km').mean()
```

| Clus_km | Crude Imports | Product Imports | Crude Exports | Product Exports |
|---|---|---|---|---|
| 0 | 388.868000 | 153.956500 | 53.668500 | 136.683750 |
| 1 | 4.597500 | 22.467667 | 215.625333 | 56.576167 |
| 2 | 43.254364 | 43.142273 | 50.039091 | 30.818545 |

Рис. 24 Таблица усредненный значений параметров каждого кластера

При помощи графика (рис. 25) можно наглядно убедиться в истинности сделанных выше выводов.

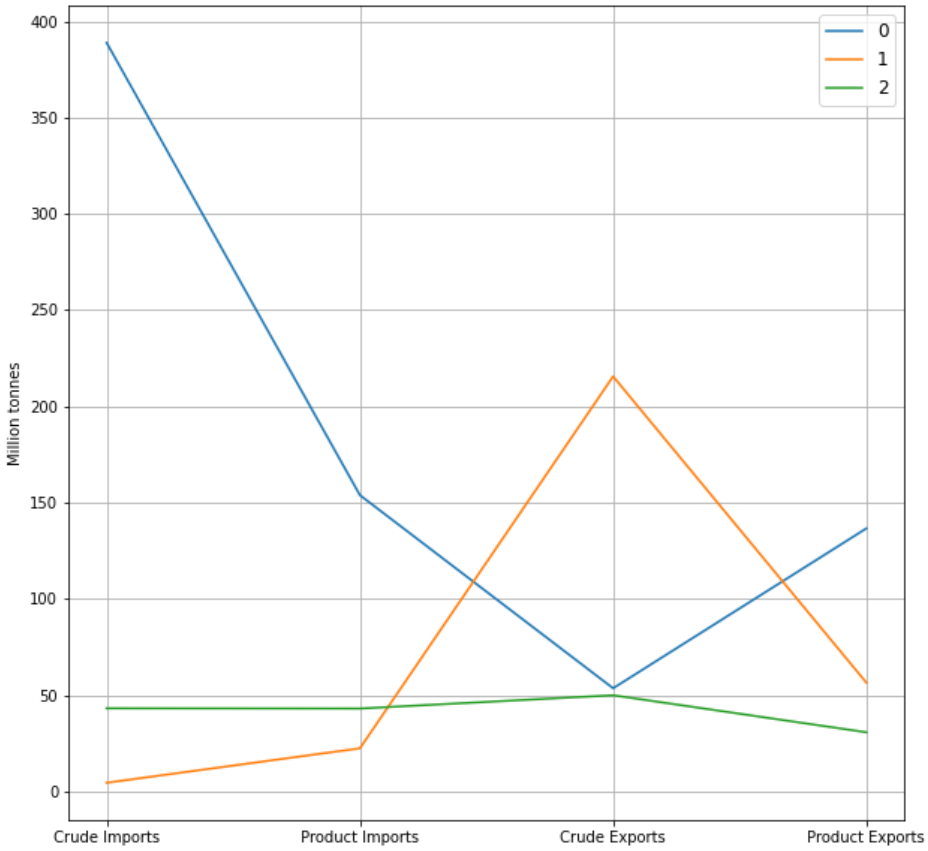


Рис. 25 График средних значений параметров по кластерам

На сегодняшний день несколько стран Европейского союза, США, Канада и Япония или неформально страны «большой семерки» в рамках санкционной политики ввели предельную цену («потолок» цен) на нефть, которая может быть импортирована из России. Правительство Российской Федерации рассматривает несколько вариантов ответного механизма на данные ограничения, одним из которых является полный запрет экспорта нефти в страны, поддержавшие «потолок» цен на нефть.

В результате выполнения кластерного анализа было определено, что как раз Россия является одной из главных стран-экспортёров сырой нефти, а страны «большой семерки» главными импортерами нефти. Если Россия в ответ на введенные санкции прекратит поставки нефти в страны G7, то необходимо будет в кратчайшие сроки определить новых потребителей экспортируемого сырья, иначе такое резкое сокращение экспорта нанесет

непоправимый урон российской экономике. Новыми потребителями могут стать страны, объединенные в нашей модели в кластер №2, а также Китай и страны Азии, которые не поддержат введение ограничений на цены российской нефти.

Результаты кластерного анализа результатов мировой торговли нефтью и нефтепродуктами могут быть использованы для более глубокого исследования с использованием иных инструментов интеллектуального анализа данных.

# ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы цель, которая заключалась в выделении особенностей кластерного анализа, рассмотрении его методов и метрик, применении алгоритма k-means для разделения стран на кластеры по данным о торговле нефтью и нефтепродуктами за 2021 год и обработке полученных результатов, была достигнута. Для достижения поставленной цели были решены следующие задачи:

1. Произведен анализ литературных источников по выбранной предметной области.
2. Ознакомление с общими сведениями о кластерном анализе, его трактовках и определениях в различных источниках.
3. Рассмотрение наиболее популярных методов кластерного анализа и применяемых метрик.
4. Выполнен анализ алгоритма k-means.
5. Ознакомление с перечнем задач нефтегазовой отрасли, в которых может быть применим кластерных анализ.
6. Осуществлены поиск и выбор исходных данных, связанных с нефтегазовым комплексом, для осуществления кластерного анализа методом k-means.
7. Осуществлена реализация метода кластеризации k-means на выбранном наборе исходных данных.
8. Выполнены визуализация, анализ и интерпретация полученных результатов.

Кластерный анализ результатов торговли нефтью и нефтепродуктами за прошедший год позволил разделить страны по группам, каждая из которых имеет свои отличительные особенности, характеризующие общие тенденции и направления ведения внешней торговли энергоносителями. Используя результаты данного анализа, были выдвинуты некоторые предположения о

реакции мирового рынка нефти и нефтепродуктов на введение ограничений на цены российской нефти и возможные ответные меры со стороны России.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ


1. Григорьев Л.И., Санжаров В.В., Тупысев А.М. Интеллектуальный анализ данных; примеры нефтегазовой отрасли: Учебное пособие – М.: Издательский центр РГУ нефти и газа имени И. М. Губкина, 2015. – 121 с.
2. Sarmah, S. A grid-density based technique for finding clusters in satellite image / S. Sarmah, D.K. Bhattacharyya // Pattern Recognition Letters. – 2012. – Vol. 33. – No. 5. – P. 589-604.
3. Дюран Б., Оделл П. Кластерный анализ //М.: статистика. – 1977. – Т. 128. – С. 2.
4. Гитис Л.Х. Статистическая классификация и кластерный анализ. – М.: Издательство Московского государственного горного университета, 2003. – 157 с.
5. Филимонова И.В., Эдер Л.В., Проворная И.В., Комарова А.В. Кластерный анализ компаний нефтяной промышленности по параметрам налоговой нагрузки. Экономика промышленности / Russian Journal of Industrial Economics. 2018;11(4):377-386. https://doi.org/10.17073/2072-1633-2018-4-377-386
6. «BP Statistical Review of World Energy June 2022» [Электронный ресурс] — URL: https://www.bp.com/en/global/corporate/energy-economics/statistical-review-of-world-energy.html (дата обращения 12.12.2022).

# ПРИЛОЖЕНИЕ 1

# ЛИСТИНГ ИСПОЛЬЗУЕМЫХ ФАЙЛОВ

```
import numpy as np
from sklearn import datasets
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score

df = pd.read_csv('oil trades.csv', sep=';')
df

data = df.drop("Countries", axis=1)
counries = df["Countries"].values
data = data.astype(np.float)
data.head()

print(counries.transpose())

X = data.values[:,:]
X = np.nan_to_num(X)
X

distortions = []
K = range(1,10)
for k in K:
    kmeanModel = KMeans(n_clusters=k)
    kmeanModel.fit(X)
    distortions.append(kmeanModel.inertia_)

plt.figure(figsize=(16,8))
plt.plot(K, distortions, 'bx-')
plt.xlabel('k')
plt.ylabel('Distortion')
plt.title('The Elbow Method showing the optimal k')
plt.show()

K = range(3,5)
for k in K:
  km = KMeans(init = "random", n_clusters=k, n_init = 12)
  km.fit_predict(X)
  score = silhouette_score(X, km.labels_, metric='euclidean')

  print('Silhouetter Score for k = %d: %.3f' % (k, score))

clusterNum = 3
k_means = KMeans(init = "random", n_clusters=clusterNum, n_init = 12)
k_means.fit(X)
labels = k_means.labels_
centers = k_means.cluster_centers_
print(centers, labels)

data["Countries"] = counries
data["Clus_km"] = labels
data

```

```
cluster_0 = []
cluster_1 = []
cluster_2 = []

for i in data.values:
  n = i[len(data.values[0,:])-1]
  if n == 0:
    cluster_0.append(i[-2])
  elif n == 1:
    cluster_1.append(i[-2])
  elif n == 2:
    cluster_2.append(i[-2])
print("Cluster 0 ", cluster_0, "\n","Cluster 1 ",cluster_1,"\n","Cluster 2 ", cluster_2, "\n")

plt.figure(figsize=(8,6))
plt.scatter(X[:, 2], X[:, 3], c=labels.astype(np.float), alpha=1)
plt.xlabel("Crude Exports")
plt.ylabel("Product Exports")
plt.scatter(centers[:, 2], centers[:, 3], marker='x', c="r")
plt.show()

plt.figure(figsize=(8,6))
plt.scatter(X[:, 0], X[:, 1], c=labels.astype(np.float), alpha=1)
plt.xlabel("Crude Imports")
plt.ylabel("Product Imports")
plt.scatter(centers[:, 0], centers[:, 1], marker='x', c="r")
plt.show()

plt.figure(figsize=(8,6))
plt.scatter(X[:, 0], X[:, 2], c=labels.astype(np.float), alpha=1)
plt.xlabel("Crude Imports")
plt.ylabel("Crude Exports")
plt.scatter(centers[:, 2], centers[:, 3], marker='x', c="r")
plt.show()

plt.figure(figsize=(8,6))
plt.scatter(X[:, 0], X[:, 3], c=labels.astype(np.float), alpha=1)
plt.xlabel("Product Imports")
plt.ylabel("Product Exports")
plt.scatter(centers[:, 0], centers[:, 3], marker='x', c="r")
plt.show()

from mpl_toolkits.mplot3d import Axes3D
fig = plt.figure(1, figsize=(8, 6))
plt.clf
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=134)

ax.set_xlabel("Crude Imports")
ax.set_ylabel("Crude Exports")
ax.set_zlabel("Product Exports")

ax.scatter(X[:, 0], X[:, 2], X[:, 1], c=labels.astype(np.float))

data.groupby('Clus_km').mean()

plt.figure(figsize=(10, 10))
plt.ylabel("Million tonnes")
clusters_arr = np.arange(0, clusterNum, 1)
for i in clusters_arr:
  plt.plot(data.groupby('Clus_km').mean().iloc[i, :], label=i)
plt.grid(True)
plt.legend(loc='best', fontsize=12)

from scipy.cluster.vq import kmeans,vq

clusterNum = 3
centers_scipy, distortion = kmeans(X, k_or_guess=clusterNum)
labels_scipy,_ = vq(X,centers_scipy)
```