

|  |  |  |        |
|--|--|--|--------|
| Документ подписан простой электронной подписью<br>Информация о владельце:<br>ФИО: Таскаев Сергей Валерьевич<br>Должность: Ректор<br>Дата подписания: 08.04.2026 16:43:51<br>Уникальный программный ключ:<br>04c19ed8bfb98f3b6cb77c486b9a8788b8323737 | МИНОВЕРНАУКИ РОССИИ<br>Федеральное государственное бюджетное образовательное учреждение высшего образования «Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ») | Рабочая программа дисциплины "Программные средства для задач искусственного интеллекта" по направлению подготовки (специальности) 01.04.02 "Прикладная математика и информатика" направленности (профилю) <u>Математическое моделирование и искусственный интеллект</u> ФГБОУ ВО «ЧелГУ» | стр. 1 |
|--|--|--|--------|

## **Рабочая программа дисциплины (модуля)\***

Программные средства для задач искусственного интеллекта

Направление подготовки (специальность)

01.04.02 Прикладная математика и информатика

Направленность (профиль)

Математическое моделирование и искусственный интеллект

Присваиваемая квалификация (степень)

магистр

Форма обучения

очная

Год набора 2026

\*Рабочая программа дисциплины (модуля) адаптирована для инклюзивного обучения инвалидов и лиц с ограниченными возможностями здоровья

Челябинск 2026 г.



## Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре ОПОП
3. Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля)
4. Объем дисциплины (модуля)
5. Структура и содержание дисциплины (модуля)
6. Фонд оценочных средств
  - 6.1. Перечень видов оценочных средств
  - 6.2. Типовые контрольные задания и иные материалы для текущей аттестации
  - 6.3. Типовые контрольные вопросы и задания для промежуточной аттестации
  - 6.4. Критерии оценивания
7. Учебно-методическое и информационное обеспечение дисциплины (модуля)
  - 7.1. Рекомендуемая литература
  - 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"
  - 7.3. Перечень информационных технологий
8. Материально-техническое обеспечение дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Специальные условия освоения дисциплины обучающимися с инвалидностью и ограниченными возможностями здоровья



### 1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью данной дисциплины является формирование профессиональных навыков и компетенций в области проектировании и/или разработки программного обеспечения систем искусственного интеллекта. К задачам дисциплины относятся: изучение средств и языков, используемых при проектировании и разработке систем искусственного интеллекта, знакомство с основными методами поиска решений, применяемых в системах искусственного интеллекта, формирование аналитических способностей, которые позволят делать обоснованный выбор изученных методов, средств и языков при решении задач искусственного интеллекта.

Результаты обучения по дисциплине направлены на достижение индикаторов соответствующих компетенций.

ПК-2

ПК-2.1. Исследует и разрабатывает архитектуры систем искусственного интеллекта для различных предметных областей.

### 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Цикл (раздел) ОПОП: Б1.В.02

**2.1 Требования к предварительной подготовке обучающегося:**

**2.2 Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:**

Современные нейросетевые технологии

Технологии искусственного интеллекта в задачах автоматизации производственных процессов

Методы и технологии машинного обучения

### 3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

**ПК-2: Способен исследовать и разрабатывать архитектуры систем искусственного интеллекта для различных предметных областей на основе комплексов методов и инструментальных средств систем искусственного интеллекта**

**Знать:**

Для достижения ПК-2.1.: знать инструменты инсталляции программного обеспечения для задач искусственного интеллекта.

**Уметь:**

Для достижения ПК-2.1.: выбирать инструментальные средства для решения задач машинного обучения в зависимости от типа задачи.

**Владеть:**

Для достижения ПК-2.1.: имеет практический опыт использования программных средств для реализации алгоритмов машинного обучения и алгоритмов построения искусственных нейронных сетей.

**В результате освоения дисциплины обучающийся должен**

**3.1 Знать:**

3.1.1 инструменты инсталляции и возможности современного программного обеспечения для решения задач искусственного интеллекта.

**3.2 Уметь:**

3.2.1 выбирать инструментальные средства для решения задач машинного обучения в зависимости от типа задачи.

**3.3 Владеть:**

3.3.1 решения практических задач с помощью алгоритмов машинного обучения и искусственных нейронных сетей.



#### 4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ)

|  |  |
|--|--|
| Общая трудоемкость   | <b>3 ЗЕТ</b>                               |
| Часов по учебному плану : 108<br>в том числе :<br>аудиторные занятия : 34<br>самостоятельная работа : 73,8<br>:<br>контактная работа: 34,2<br>ИКР: 0,2 | Виды контроля в семестрах:<br><br>зачеты 1 |

#### 5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

| Код занятия | Наименование разделов и тем /вид занятия/  | Семестр / Курс | Часов | Литература                     |
|-------------|--|----------------|-------|--------------------------------|
|             | <b>Раздел 1. Кластеризация</b>   |                |       |                                |
| 1.1         | Кластеризация /Лаб/  | 1              | 4     | Л1.1Л2.1<br>Э1 Э2 Э3           |
|             | <b>Раздел 2. Классификация</b>   |                |       |                                |
| 2.1         | Классификация. Алгоритм к ближайших соседей. Наивный байесовский классификатор. Деревья решений. /Лаб/ | 1              | 4     | Л1.1Л2.1<br>Э1 Э2 Э3           |
| 2.2         | Классификация. Логистическая регрессия. Машина опорных векторов. /Лаб/                                 | 1              | 4     | Л1.1Л2.1<br>Э1 Э2 Э3           |
|             | <b>Раздел 3. Регрессионный анализ</b>  |                |       |                                |
| 3.1         | Регрессионный анализ. /Лаб/  | 1              | 4     | Л1.1Л2.1<br>Э1 Э2 Э3           |
|             | <b>Раздел 4. Генетический алгоритм</b>   |                |       |                                |
| 4.1         | Генетический алгоритм /Лаб/  | 1              | 4     | Л1.1<br>Э1 Э2 Э3               |
|             | <b>Раздел 5. Искусственные нейронные сети</b>  |                |       |                                |
| 5.1         | Введение в искусственные нейронные сети /Лаб/  | 1              | 4     | Л1.2 Л1.3<br>Э1 Э2 Э3          |
| 5.2         | Сверточные нейронные сети /Лаб/  | 1              | 4     | Л1.2 Л1.3<br>Э1 Э2 Э3          |
|             | <b>Раздел 6. Ансамбли алгоритмов машинного обучения</b>  |                |       |                                |
| 6.1         | Ансамбли алгоритмов машинного обучения: случайные леса, градиентный бустинг /Лаб/                      | 1              | 6     | Л1.1Л2.1<br>Э1 Э2 Э3           |
|             | <b>Раздел 7. Самостоятельная работа и зачет</b>  |                |       |                                |
| 7.1         | Подготовка к зачету /Ср/   | 1              | 33,8  | Л1.1 Л1.2 Л1.3Л2.1<br>Э1 Э2 Э3 |
| 7.2         | Подготовка к лабораторным работам /Ср/   | 1              | 40    | Л1.1 Л1.2 Л1.3Л2.1<br>Э1 Э2 Э3 |
| 7.3         | Индивидуальные консультации, текущий контроль /ИКР/  | 1              | 0,2   | Л1.1 Л1.2 Л1.3Л2.1<br>Э1 Э2 Э3 |

#### 6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

##### 6.1. Перечень видов оценочных средств

Вопросы к зачету.  
Лабораторные работы.

##### 6.2. Типовые контрольные задания и иные материалы для текущей аттестации



Лабораторная работа №1. Кластеризация.  
Лабораторная работа №2. Классификация. Алгоритм к ближайших соседей. Наивный байесовский классификатор. Деревья решений.  
Лабораторная работа №3. Регрессионный анализ.  
Лабораторная работа №4. Классификация. Логистическая регрессия. Машина опорных векторов.  
Лабораторная работа №5. Генетический алгоритм.  
Лабораторная работа №6. Введение в искусственные нейронные сети.  
Лабораторная работа №7. Сверточные нейронные сети.  
Лабораторная работа №8. Ансамбли алгоритмов машинного обучения: случайные леса, градиентный бустинг.

Образец лабораторной работы приведен в приложении.

### 6.3. Типовые контрольные вопросы и задания для промежуточной аттестации

Вопросы к зачету

1. Метод k-средних.
2. Иерархическая агломеративная кластеризация.
3. Метод k ближайших соседей.
4. Наивный байесовский классификатор.
5. Деревья решений.
6. Парная линейная регрессия. Оценки коэффициентов регрессии.
7. Парная полиномиальная регрессия. Парная показательная регрессия. Сведение этих моделей к модели линейной регрессии.
8. Логистическая регрессия. Ее применение для бинарной классификации.
9. Модель мультиномиального логита.
10. Машина опорных векторов. Нелинейные ядра.
11. Генетический алгоритм. Основные определения.
12. Генетический алгоритм. Этапы генетического алгоритма.
13. Нейронные сети. Применение нейронных сетей для решения задачи классификации. Пример архитектуры нейронной сети.
14. Сверточные нейронные сети. Сверточные слои Conv2D и MaxPooling2D.
15. Алгоритм случайного леса (Random Forest).
16. Градиентный бустинг.

### 6.4. Критерии оценивания

В течение учебного семестра студенты за каждый вид работы получают баллы. Кроме этого на зачете максимально можно получить 20 баллов. Итоговая оценка складывается из суммы баллов, полученных за работу в семестре и за ответ на зачете. Затем полученная сумма баллов переводится в оценку. При этом допускается получение студентом автоматической оценки только по результатам работы в семестре.

Набранные баллы

Оценка

Менее 61  
61 и более

незачтено  
зачтено

Начисляемые баллы за выполнение плановых заданий

Выполнение лабораторной работы №1 - 10  
Выполнение лабораторной работы №2 - 10  
Выполнение лабораторной работы №3 - 10  
Выполнение лабораторной работы №4 - 10  
Выполнение лабораторной работы №5 - 10  
Выполнение лабораторной работы №6 - 10  
Выполнение лабораторной работы №7 - 10  
Выполнение лабораторной работы №8 - 10  
Выполнение заданий на зачете - 20

Порядок начисления баллов в лабораторных работах №1-8.

Написана программа, использующая инструментарий указанной программной библиотеки – 3 балла.

Выполнен анализ результатов, выводы логичны и оформлены в соответствии с указаниями – 3 балла.

Студент может пояснить порядок получения результатов, расчетов и графиков – 4 балла.

На зачете происходит оценивание учебной деятельности обучающихся по дисциплине на основе полученных оценок



за контрольно-рейтинговые мероприятия текущего контроля. Контрольное мероприятие промежуточной аттестации проводится во время зачета в виде устного опроса, если магистранту не хватило баллов для выставления зачета по текущему контролю. В этом случае, при условии выполнения всех лабораторных работ, студенту задаются 5 вопросов из разных тем курса, позволяющих оценить сформированность компетенций. Время на подготовку ответов не предусмотрено. Использование вспомогательных материалов при подготовке ответа не допускается. Правильный ответ на вопрос соответствует 4 баллам. Неправильный ответ на вопрос соответствует 0 баллов.

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

### 7.1. Рекомендуемая литература

#### 7.1.1. Основная литература

|      | Авторы, составители     | Заглавие   | Издательство, год        | Ресурс |
|------|-------------------------|--|--------------------------|--------|
| Л1.1 | Рашка С.                | Python и машинное обучение: крайне необходимое пособие по новейшей предсказательной аналитике, обязательное для более глубокого понимания методологии машинного обучения ( <a href="https://e.lanbook.com/book/100905">https://e.lanbook.com/book/100905</a> ) | Москва : ДМК Пресс, 2017 | ЭБС    |
| Л1.2 | Антонио Д., Суджит П.   | Библиотека Keras – инструмент глубокого обучения. Реализация нейронных сетей с помощью библиотек Theano и TensorFlow ( <a href="https://e.lanbook.com/book/111438">https://e.lanbook.com/book/111438</a> )   | Москва : ДМК Пресс, 2018 | ЭБС    |
| Л1.3 | Паттерсон Д., Гибсон А. | Глубокое обучение с точки зрения практика ( <a href="https://e.lanbook.com/book/116122">https://e.lanbook.com/book/116122</a> )  | Москва : ДМК Пресс, 2018 | ЭБС    |

#### 7.1.3. Методические разработки

|      | Авторы, составители            | Заглавие   | Издательство, год  | Ресурс |
|------|--------------------------------|--|--|--------|
| Л3.1 | Крутиков В. Н., Мешечкин В. В. | Анализ данных: учебное пособие ( <a href="https://biblioclub.ru/index.php?page=book&amp;id=278426">https://biblioclub.ru/index.php?page=book&amp;id=278426</a> ) | Кемерово : Кемеровский государственный университет, 2014 | ЭБС    |

### 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

|    |  |
|----|--|
| Э1 | Лань [Электронный ресурс] : электронно-библиотечная система (ЭБС) / издательство Лань. – URL: <a href="http://e.lanbook.com/">http://e.lanbook.com/</a> <a href="http://e.lanbook.com/">http://e.lanbook.com/</a>                                    |
| Э2 | Университетская библиотека онлайн [Электронный ресурс] : электронно-библиотечная система (ЭБС) / ООО Директмедиа Паблишинг. – URL: <a href="http://biblioclub.ru/">http://biblioclub.ru/</a> <a href="http://biblioclub.ru">http://biblioclub.ru</a> |
| Э3 | Znanium.com [Электронный ресурс] : электронно-библиотечная система (ЭБС) / Научно-издательский центр ИНФРА-М. – URL: <a href="http://znanium.com/">http://znanium.com/</a> <a href="http://znanium.com/">http://znanium.com/</a>                     |

### 7.3 Перечень информационных технологий

#### 7.3.1 Программное обеспечение

Adobe Connect Acrobat

LMS Moodle

Python

#### 7.3.2 Профессиональные базы данных и информационно-справочные системы

1. Научная электронная библиотека eLIBRARY.RU (<https://elibrary.ru/defaultx.asp?>) eLIBRARY.RU : научная электронная библиотека : сайт. – Москва, 2000 – . – URL: <https://elibrary.ru>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

2. Реферативная база по математике MathSciNet (<https://mathscinet.ams.org/mathscinet/>) Mathematical Reviews (MR) : реферативная база данных / American Mathematical Society. – URL: <http://www.ams.org/mathscinet/>. – Яз. рус., англ. – Режим доступа: для зарегистрир. пользователей ЧелГУ. – Текст : электронный

## 8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для реализации дисциплины используются компьютерные классы для проведения лабораторных работ, учебные аудитории для проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы.



Учебные аудитории укомплектованы специализированной мебелью (подразумевается наличие стандартных рабочих (посадочных) мест) и техническими средствами обучения (переносное и / или стационарное мультимедийное оборудование: экран, ноутбук, проектор).

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с подключением к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду университета

## 9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Учебным планом предусмотрена самостоятельная работа студентов. Самостоятельная работа проводится с целью углубления знаний по дисциплине и предусматривает:

- подготовку к лабораторным работам;
- подготовку к сдаче зачета.

При планировании времени на самостоятельную работу студентам необходимо предусмотреть регулярное повторение пройденного материала.

В случае применения при изучении дисциплины электронного обучения, дистанционных образовательных технологий общение обучающихся и преподавателя осуществляется в режиме реального или отложенного времени, при этом используются возможности системы дистанционного обучения Moodle и электронная почта.

Большую часть времени обучающиеся самостоятельно работают с учебно-методическими материалами. Студенты имеют возможность консультироваться с преподавателем по всем вопросам, возникающим в ходе самостоятельной работы, посредством электронной почты, сообщений системы дистанционного обучения Moodle.

Доступ обучающегося к учебным ресурсам в режиме отложенного времени, самостоятельной работы осуществляется через сеть Интернет в удобном для него месте, времени и темпе.

При обучении лиц с ограниченными возможностями здоровья электронное обучение, дистанционные образовательные технологии предусматривают возможность приема-передачи информации в доступных для них формах.

Реализация дисциплины с применением электронного обучения, дистанционных образовательных технологий (далее – ЭО, ДОТ) осуществляется на основании «Положения о реализации основных и дополнительных образовательных программ с применением электронного обучения и дистанционных образовательных технологий в федеральном государственном бюджетном образовательном учреждении высшего образования «Челябинский государственный университет», «Положения о порядке зачета обучающимся по основным профессиональным образовательным программам высшего образования в ФГБОУ ВО «ЧелГУ» результатов освоения в организациях, осуществляющих образовательную деятельность, учебных предметов, курсов, дисциплин (модулей), практик, дополнительных образовательных программ» посредством электронной информационно-образовательной среды ФГБОУ ВО «ЧелГУ». В исключительных случаях (форс-мажор и т.п.) при реализации образовательной деятельности с применением ЭО, ДОТ могут применяться компоненты, не входящие в перечень электронной информационно-образовательной среды.

## 10. СПЕЦИАЛЬНЫЕ УСЛОВИЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ ОБУЧАЮЩИМИСЯ С ИНВАЛИДНОСТЬЮ И ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Освоение дисциплины инвалидами и лицами с ограниченными возможностями здоровья осуществляется с использованием специальных технических средств и информационных технологий, предоставляемых Ресурсным учебно-методическим центром по обучению инвалидов и лиц с ограниченными возможностями здоровья ЧелГУ по запросу обучающегося (мобильные специальные технические средства для лиц с нарушениями зрения и с нарушением слуха, ассистивные информационные технологии).

При необходимости для обучающихся с нарушениями зрения на рабочих местах для проведения практических или лабораторных занятий устанавливается специальное программное обеспечение (программа речевой навигации, речевые синтезаторы, экранные лупы).

В учебные аудитории обеспечивается беспрепятственный доступ для обучающихся с инвалидностью и с ограниченными возможностями здоровья. В каждой аудитории, где обучаются инвалиды и лица с ограниченными возможностями здоровья, предусматривается соответствующее количество мест для обучающихся с учетом нарушений их здоровья.

Для освоения дисциплины инвалидам и лицам с ограниченными возможностями здоровья предоставляется доступ к печатным источникам, имеющимся в научной библиотеке ЧелГУ, с помощью специальных технических средств; доступ с помощью специальных технических и программных средств к электронным источникам, представленным в форме электронного документа в фонде научной библиотеки ЧелГУ или электронно-библиотечных системах.

Учебно-методические материалы для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья предоставляются в формах, адаптированных к ограничениям их здоровья и особенностям восприятия



информации.

Для инвалидов и лиц с ограниченными возможностями здоровья освоение дисциплины может быть частично или полностью осуществлено с использованием дистанционных образовательных технологий.

При проведении промежуточной аттестации по дисциплине обучающимся с инвалидностью и с ограниченными возможностями здоровья обеспечивается по их заявлению предоставление в доступной форме в зависимости от их индивидуальных особенностей инструкции о порядке проведения промежуточной аттестации, оценочных средств и возможности ответов на задания (письменно на бумаге, набор ответов на компьютере, письменно шрифтом Брайля, с использованием услуг ассистента, устно).

При проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование предоставленных ЧелГУ или собственных технических средств, необходимых им в связи с их индивидуальными особенностями. При необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на задания, процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

## Лабораторная работа 1. Кластеризация.

Под **кластеризацией** понимается задача разбиения всей исходной совокупности элементов на отдельные группы однородных объектов, сходных между собой, так, чтобы эти группы имели друг от друга отчетливые отличия. В качестве меры близости между каждой парой изучаемых объектов может использоваться некоторая функция: евклидово или манхэттенское расстояние, коэффициент корреляции Пирсона и др.

### 1. Метод k-средних

Метод k-средних выполняет кластеризацию следующим образом:

1. Назначается число групп ( $k$ ), на которые должны быть разбиты данные. Случайно выбирается  $k$  объектов исходного набора как первоначальные центры кластеров (**центроиды**).

2. Каждому наблюдению присваивается номер группы по самому близкому центроиду, т.е. на основании наименьшего расстояния между точкой и центроидом  $\mu_k$ .

3. Затем центроиды вычисляются как центры масс для каждого кластера, полученного на предыдущем шаге. Минимизируется общий внутригрупповой разброс  $W_{total} = \sum_k W(C_k)$ , где  $W(C_k) = \sum_{x_i \in C_k} (x_i - \mu_k)^2$ ,  $\mu_k$  - координаты центроидов. Для это шаги 2 и 3 повторяются многократно, пока назначения групп не прекращают изменяться или не достигнуто заданное число итераций.

#### 1.1. Определение количества кластеров «методом локтя»

При кластеризации методом k-средних количество кластеров чаще всего оценивают с помощью «метода локтя» (Elbow method). Это интуитивная, довольно грубая эвристика. На графике вдоль оси Ох откладывается количество кластеров, а вдоль оси Оу – некоторая величина, характеризующая качество кластеризации, например,  $W_{total}$  для разного количества кластеров (например, в методе KMeans на языке Python – это значение атрибута `inertia_`). Оптимальное количество кластеров соответствует значению  $k$ , после которого величина  $W_{total}$  перестает резко падать (см. рис. 1).

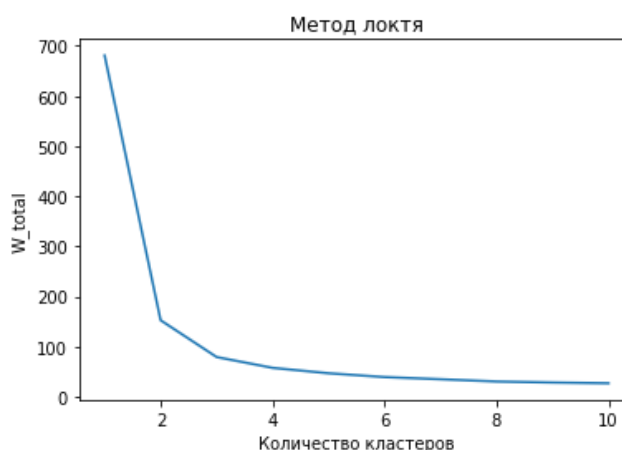


Рис. 1. Визуализация «метода локтя».

## 1.2. Класс Kmeans из библиотеки scikit-learn

В Python для реализации вычислений по методу k-средних необходимо импортировать из класса sklearn.cluster (библиотека scikit-learn) класс KMeans:

```
from sklearn.cluster import KMeans
```

### Параметры класса KMeans()

|            |  |
|------------|--|
| n_clusters | Количество кластеров   |
| init       | Метод определения начального положения центроидов (k-means++ - в качестве начальных центроидов выбираются точки наблюдений, далеко отстоящие друг от друга; random – начальные центроиды определяются случайным образом) |
| max_iter   | Максимальное количество итераций   |

### Атрибуты класса KMeans()

|                  |   |
|------------------|---|
| cluster_centers_ | Массив центроидов   |
| inertia_         | Сумма квадратов расстояний выборок до их ближайших центроидов |
| labels_          | Метки кластеров для каждой точки.                             |

### Методы класса KMeans()

|                |   |
|----------------|---|
| fit(X)         | Выполняет кластеризацию методом k-средних                 |
| fit_predict(X) | Находит значение кластера для каждого наблюдения данных X |

См. более подробно здесь

[sklearn.cluster.KMeans — scikit-learn 1.1.2 documentation](#)

## 1.3. Пример

```
import numpy as np
```

```
from sklearn.cluster import KMeans
```

```
X = np.array([[5,3],
```

```
    [10,15],
```

```
    [15,12],
```

```
    [24,10],
```

```
    [30,30],
```

```
    [85,70],
```

```
    [71,80],
```

```
    [60,78],
```

```
    [70,55],
```

```
    [80,91],])
```

```
cluster = KMeans(init='k-means++', n_clusters=3, max_iter=100)
```

```
cluster.fit(X)
```

```
print(cluster.labels_)
```

## 2. Иерархическая кластеризация

Иерархические методы кластеризации - совокупность алгоритмов кластеризации, направленных на создание иерархии (дерева) вложенных кластеров.

Иерархические алгоритмы делятся на:

а) **агломеративные** (восходящие), характеризующиеся последовательным объединением мелких кластеров в более крупные (число кластеров уменьшается);

б) **дивизимные** (нисходящие), характеризующиеся последовательным разделением кластеров, начиная с одного кластера, который содержит все объекты.

В обоих случаях результат работы алгоритма представляет собой древовидную структуру, или дендрограмму.

### 2.1. Агломеративная кластеризация

Ниже приведены шаги, связанные с **агломеративной кластеризацией**:

1. Вначале рассмотрим каждую точку данных как один кластер. Таким образом, число кластеров вначале будет равно  $K$ , а  $K$ -целое число, представляющее количество точек данных.

2. Сформируем кластер, объединив в один кластер две ближайшие точки. В результате чего получатся  $K-1$  кластер.

3. Далее, объединим два ближайших кластера. Таким образом, получим  $K-2$  кластера.

4. Повторяйте описанные выше три шага, пока не образуется один большой кластер.

Существуют различные способы определения расстояния между кластерами. Ниже приведены некоторые из вариантов измерения расстояния между двумя кластерами:

1. Увеличение суммы квадратов ошибок после объединения двух кластеров в один кластер.

2. Расстояние между самыми дальними точками двух кластеров.

3. Расстояние между центроидами двух кластеров.

4. Среднее значение расстояний между всеми возможными парами точек, лежащими в разных кластерах.

### 2.2. Класс `AgglomerativeClustering` из библиотеки `scikit-learn`

В Python для проведения агломеративной кластеризации необходимо импортировать из класса `sklearn.cluster` (библиотека `scikit-learn`) класс `AgglomerativeClustering`:

```
from sklearn.cluster import AgglomerativeClustering
```

#### Параметры класса `AgglomerativeClustering()`

|                         |                      |
|-------------------------|----------------------|
| <code>n_clusters</code> | Количество кластеров |
|-------------------------|----------------------|

|  |  |
|--|--|
| affinity   | Метрика (по умолчанию евклидова)   |
| linkage  | Критерий связи, который определяет, как вычислять расстояние между парами кластеров. |
| <b>Атрибуты класса AgglomerativeClustering()</b> |  |
| labels_  | Метки кластеров для каждой точки.  |

См. более подробно здесь

[sklearn.cluster.AgglomerativeClustering — scikit-learn 1.1.2 documentation](http://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html)

### 2.3. Пример

```
import numpy as np
from sklearn.cluster import AgglomerativeClustering
X = np.array([[5,3],
              [10,15],
              [15,12],
              [24,10],
              [30,30],
              [85,70],
              [71,80],
              [60,78],
              [70,55],
              [80,91],])
cluster = AgglomerativeClustering(n_clusters=3, affinity='euclidean', linkage='ward')
cluster.fit_predict(X)
print(cluster.labels_)

#нарисуем дендрограмму для рассматриваемого примера
import scipy.cluster.hierarchy as shc
plt.figure(figsize=(10, 7))
plt.title("Дендрограмма")
dend = shc.dendrogram(shc.linkage(X, method='ward'))
```

## 3. Задания

### 3.1. Метод k-средних.

1. Провести кластеризацию данных из файла iris.data (исключая 5-ый столбец) методом k-средних при k=2, 3, 4.
2. Визуализировать результаты: нарисовать двухмерные проекции, в которых точки, принадлежащие разным кластерам, будут выделены разным цветом.
3. Определите наилучшее количество кластеров методом локтя.

### 3.2. Иерархическая кластеризация.

1. Провести иерархическую кластеризацию на тех же данных, что и в задании 3.1.

2. Визуализировать результаты: нарисовать двухмерные проекции, в которых точки, принадлежащие разным кластерам, будут выделены разным цветом.
3. Нарисовать дендрограмму до 6 уровня (см. truncate\_mode здесь <https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.dendrogram.html> ).
4. Провести кластеризацию при каждом значении параметра linkage. Сравнить полученные результаты.

