ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ

Лекция 1

Введение в ИАД. Series и DataFrame

Оспан Ә.Ғ., PhD, доцент-исследователь

КазНУ, кафедра ИИ и Big Data

2026-2027 учебный год

Цель и содержание лекции

Цель и результаты обучения

- Определять наблюдение, признак и допустимый аналитический вопрос.
- Создавать Series и DataFrame, различать метку и позицию.
- Читать CSV и проверять схему, размер и ограничения набора.

Содержание лекции

1. **Задача ИАД**
2. **Признаки**
3. **Series**
4. **DataFrame**
5. **Чтение и проверка**

Как вопрос превращается в проверяемый расчет

1

Вопрос

Назвать показатель, сравниваемые группы и границы вывода.

2

Наблюдение

Определить, что означает одна строка и каждый признак.

3

Расчет

Проверить данные и выбрать содержательно допустимую операцию.

4

Вывод

Объяснить число, ограничения и способ повторить результат.

- Форма таблицы не определяет смысл строки: его задает источник.
- Воспроизводимый результат включает данные, код и интерпретацию.

Тип хранения и смысл признака

Поле	Смысл	Допустимый расчет
Age	Возраст	Среднее, медиана, диапазон
Position	Категория позиции	Частоты и сравнение групп
SprintSpeed	Рейтинг характеристики	Сравнение рейтингов
Name	Имя из источника	Показ записей; уникальность не обещана
Unnamed: 0	Старый индекс	Техническая сверка, не аналитический признак

Сначала определяем смысл и единицы; затем выбираем тип и операцию.

Series: значения связаны с метками

Части объекта

- Значения: данные одной переменной.
- Индекс: метки, связанные с элементами.
- name и dtype: подпись и способ хранения.

Два способа доступа

- .loc выбирает метку индекса.
- .iloc выбирает положение от нуля.
- Уникальность меток проверяется отдельно.

Series сохраняет связь значения с его меткой при отборе и арифметике.

Series: от конструкции к проверяемому числу

Пример Python

```
import pandas as pd

scores = pd.Series(
    [72, 80, 68], index=['p1', 'p2', 'p3'], name='rating'
)
print(scores.loc['p2'])
print(scores.iloc[0])
print(round(scores.mean(), 2))
```

Результат: 80; 72; 73.33

Получение данных

- p2 является меткой.
- 0 является первым положением.

Расчет

- Среднее: сумма значений / 3.
- Округляем отображение результата.

Сложение Series выравнивает метки

Метка	Первая Series	Вторая Series	Сумма
a	10	2: находится второй	12
b	20	1: находится первой	21
c	Нет значения	Если добавить 5	Пропуск без правила заполнения

Выравнивание связывает одинаковые метки, а не позиции; перед арифметикой проверяем несовпадения индексов разных источников.

DataFrame: общая ось строк, разные признаки

Организация таблицы

- Строка представляет наблюдение.
- Столбец представляет признак.
- Ячейка содержит значение признака.

Создание и контроль

- Словарь списков задает столбцы.
- Индекс и columns подписывают оси.
- shape и dtypes проверяют структуру.

Перед расчетом назовите смысл обеих осей и ожидаемую форму результата.

Один столбец: Series или DataFrame?

Пример Python

```
players = pd.DataFrame({
    'Name': ['Ada', 'Boris', 'Chen'],
    'Age': [20, 24, 22],
    'SprintSpeed': [70, 80, 75]
}, index=['p1', 'p2', 'p3'])
print(players.shape)
print(players['Age'].shape)
print(players[['Age']].shape)
```

Результат: (3, 3); (3,); (3, 1)

Один ключ

- players['Age'] возвращает Series.
- Результат имеет одну ось.

Список ключей

- players[['Age']] возвращает DataFrame.
- Результат сохраняет две оси.

Первый просмотр таблицы: разные вопросы

Инструмент	На какой вопрос отвечает	Ограничение
head / tail	Как выглядят крайние записи?	Это только фрагмент
shape / columns	Каков размер и список полей?	Не проверяет достоверность
info	Каковы типы и заполненность?	Не устанавливает смысл шкал
describe	Каковы основные статистики?	Требует содержательного отбора полей
isna	Где служебные пропуски?	Не распознает все ошибки источника

Читаем CSV и проверяем контракт

Пример Python

```
from pathlib import Path

football = pd.read_csv(Path('data') / 'football.csv')
print(football.shape)
print(football[['Name', 'Age', 'Position']].head(3))
football.info()
print(football.isna().sum().sum())
```

Результат: Форма: (12897, 42). Пропущенных ячеек: 0.

Расположение

- Запускаем из корня комплекта.
- Path задает переносимый относительный путь.

Контракт файла

- Проверяем размер и обязательные поля.
- Считаем пропущенные ячейки по всему набору.

Паспорт football.csv: что нельзя додумывать

Время и охват

- Исторический срез характеристик.
- Актуальность и полнота охвата не гарантированы.

Шкалы

- Wage/Value: единицы и период не подтверждены.
- SprintSpeed и Penalties являются рейтингами.

Идентификация

- Name может повторяться.
- Unnamed: 0 не является аналитическим признаком.

Паспорт ограничивает вывод тем, что действительно известно о происхождении и шкалах данных.

Как сделать результат воспроизводимым

1

Зафиксировать вход

Путь, версию файла, смысл полей и ограничения.

2

Записать шаги

Импорты, загрузку, преобразования и формулу результата.

3

Проверить

Форму, значения контрольного примера и границы показателя.

4

Сохранить отдельно

Производные файлы в work_results; исходник сохраняется.

- Экспорт обычной таблицы: index=False.
- Чистый повторный запуск подтверждает отсутствие скрытых зависимостей.

Описание данных и объяснение причин

Описание

- Сообщает, что наблюдается в выбранном файле.
- Указывает группы, показатель и число записей.
- Сохраняет ограничения шкал и охвата.

Причинный вывод

- Требует отдельного дизайна исследования.
- Не возникает из одного различия средних.
- Не подтверждается правдоподобием истории.

Различие наблюдаемых групп само по себе не устанавливает причину этого различия.

Ошибки, которые дают правдоподобный ответ

Структура

- Путать метку с позицией.
- Не замечать Series вместо DataFrame.

Смысл

- Усреднять идентификатор.
- Приписывать рейтингу физические единицы.

Доказательство

- Судить о файле только по head.
- Выдавать описание групп за причину.

Успешное выполнение кода проверяем отдельно от смысла полученного числа и его подписи.

Что нужно уметь после лекции

1. Анализ начинается с вопроса и определения одной строки.
2. Series связывает значения с метками; арифметика выравнивает индекс.
3. DataFrame хранит признаки разных типов на общей оси наблюдений.
4. Выбор одного ключа и списка ключей дает разную размерность.
5. CSV проверяют по всему файлу; смысл шкал фиксируют в паспорте.

Проверьте понимание структуры данных

1. Почему число строк не всегда равно числу людей?
2. Чем метка Series отличается от позиции элемента?
3. Как меняется форма при df["Age"] и df[["Age"]]?
4. Почему сумма Unnamed: 0 не является полезным показателем?
5. Какие утверждения о Penalties допускает этот файл?

Самостоятельная работа

Создайте две Series с переставленными и частично разными метками. Предскажите сумму, соберите DataFrame и объясните его форму и пропуски.

Источники и документация

Документация

1. pandas: Series и DataFrame

https://pandas.pydata.org/docs/user_guide/dsintro.html

2. pandas: первые шаги

https://pandas.pydata.org/docs/user_guide/10min.html

3. pandas: чтение и запись

https://pandas.pydata.org/docs/user_guide/io.html

Основа лекции

- Лекция 1.pdf
- Pandas. Работа с данными.pdf
- Лабораторная работа 1.docx