Модуль 1 · Python и анализ данных

Обработка и визуализация табличных климатических данных

Ph.D., старший преподаватель Мадина Турсумбаева

Кафедра метеорологии и гидрологии

Курс: Современные цифровые технологии в метеорологии

Цель лекции

Научить основным операциям с табличными климатическими данными в pandas — выбору, фильтрации, обработке пропусков, сортировке и группировке — и построению графиков на примере данных об энергетике и возобновляемых источниках по странам.

Лекция 1: один ряд, одна величина. Сегодня: много стран, много лет, много пропусков.

Результаты обучения

После лекции студент будет:

Знать

Шесть операций с таблицей: столбцы, строки, пропуски, сортировка, группировка, график

Длинный и широкий формат таблицы

Уметь

Выбрать столбцы и отфильтровать строки по условию

Найти и обработать пропуски

Сгруппировать данные и построить график по странам

Понимать

Агрегаты («Мир», континенты) — не страны

Заполненный пропуск — не наблюдение

Тип графика определяется вопросом

Сегодня: шесть операций с таблицей

1. Выбор столбцов

df[["a", "b"]] — список столбцов в двойных скобках

2. Фильтрация строк

df[условие] и isin — строки по условию и по списку значений

3. Пропуски

isna, dropna, fillna — найти, удалить или заполнить

4. Сортировка

sort_values — упорядочить по значению столбца

5. Группировка

groupby и agg — агрегировать по стране, году или месяцу

6. Визуализация

Matplotlib и seaborn — линии, сравнение стран, рассеяние

Данные: энергетика и возобновляемые источники

23 377

строк в наборе Our World in Data (Energy Dataset)

130

столбцов: мощность, выработка, доли источников

1 строка

= одна страна и один год

Почему это важно метеорологу?

Структура энергетики определяет выбросы парниковых газов.

Доли ВИЭ зависят от погоды: ветер, облачность, осадки и сток рек.

В работе используется показатель renewables_share_energy — доля ВИЭ в первичном потреблении энергии, %.

Загрузка и выбор столбцов

```
import pandas as pd
raw = pd.read_csv("owid-energy-data.csv")
raw.shape                                  # (23377, 130)
# список столбцов — в двойных скобках
df = raw[["country", "year", "iso_code",
          "renewables_share_energy"]]
df = df.rename(columns={"renewables_share_energy": "res"})
df.head(3)
```

Результат

country	year	iso_code	res
ASEAN (Ember)	2000	NaN	NaN
ASEAN (Ember)	2001	NaN	NaN
ASEAN (Ember)	2002	NaN	NaN

Две особенности

NaN — значения нет.

ASEAN — не страна: у группы нет кода ISO.

Запомните

[[ ]] — выбор нескольких столбцов.

rename — короткое понятное имя.

Длинный и широкий формат таблицы

Длинный формат

Одна строка — одна пара «страна, год». Так устроено большинство открытых наборов данных.

Столбцы: country, year, res.

Широкий формат

Строки — годы, столбцы — страны. Удобно сравнивать страны и считать разности строк.

Получается командой pivot.

```
# pivot: что станет строками, что столбцами, что в ячейках
wide = p.pivot(index="year", columns="country", values="res")
wide.shape                                  # (35, 79)
```

Перед pivot всегда решите: какой столбец станет строками, какой — столбцами и какие значения попадут в ячейки.

Фильтрация строк

```
kz = df[df["country"] == "Kazakhstan"]
len(kz)                                    # 41 строка
kz = df[(df["country"] == "Kazakhstan") &
        (df["year"] >= 1990)]
kz.tail(3)
df[df["country"].isin(["Kazakhstan", "Germany"])]
```

country	year	iso_code	res
Kazakhstan	2023	KAZ	4,179
Kazakhstan	2024	KAZ	5,120
Kazakhstan	2025	KAZ	NaN

Операторы

== , >= , < , != — сравнение

& «и», | «или» — каждое условие в скобках

isin — значение входит в список

Значение NaN у Казахстана в 2025 году означает: показатель ещё не заполнен.

Пропуски: сколько их и почему

73 %

значений показателя ВИЭ отсутствуют (72,7 %)

Почему их так много?

Ряд начинается с 1965 года.

Для многих стран показатель не рассчитывается.

Последний год ещё не заполнен.

```
df.isna().sum()                          # пропуски по столбцам
df["res"].isna().mean() * 100            # доля пропусков, %
clean = df.dropna(subset=["res"])        # удалить строки без значения
clean.shape
```

Результат

```
country 0   year 0
iso_code 6112   res 16998
72,7 %   (6379, 4)
```

dropna(subset=...) ищет пропуски только в указанных столбцах.

Удалить или заполнить?

Удалить · dropna

Значение нельзя восстановить.

Интерполировать · interpolate

Короткий разрыв в плавном ряду.

Заполнить · fillna

Только осознанно и с пометкой в отчёте.

```
med = df["res"].median()              # медиана по всем строкам
kz["res"].max()                       # максимум Казахстана с 1990
df["res"] = df["res"].fillna(med)     # так делать нельзя без пометки
```

Медиана по всем странам (6,55 %) выше максимума Казахстана с 1990 г. (5,77 %): заполнение создаёт «рекорд», которого не было.

Заполненное значение — не наблюдение.

Сортировка: кто лидирует по доле ВИЭ?

```
y = clean[clean["year"] == 2024]
y.sort_values("res", ascending=False).head(10)
# по убыванию (ascending=False); head(10)
```

Ошибка в результате

Пятая строка — не страна, а агрегат Other South America (EI).

Код отработал без ошибок, а ответ неверен.

Страна	ВИЭ, %
Iceland	80,498
Norway	71,812
Sweden	51,318
Brazil	49,616
Other South America (EI)	44,722
Austria	43,347
New Zealand	42,554
Denmark	41,722
Switzerland	39,868
Portugal	39,741

Ловушка: агрегаты среди стран

```
# у агрегатов нет кода страны
c = clean[clean["iso_code"].notna()]
c["country"].nunique()                  # 79
y = c[c["year"] == 2024]
```

- В столбце country есть не только страны: мир, континенты и группы стран.
- Фильтр по iso_code убирает агрегаты: у большинства из них кода нет.
- Казахстан в 2024 году — 61-е место из 79 (5,1 %).

Сначала выясните, что лежит в столбце

value_counts() и unique() покажут все значения.

Сравните количество строк до и после фильтра.

Только потом считайте рейтинги и средние.

Данные: Our World in Data, Energy Dataset (github.com/owid/energy-data).

Группировка: разделить, применить, объединить

1

Разделить

На группы: страна, год, станция, месяц

2

Применить

mean, median, max, sum или несколько функций сразу

3

Объединить

Одна таблица: по одной строке на каждую группу

```
p = c[c["year"] >= 1990]
g = p.groupby("country")["res"].mean()    # по странам
g.sort_values(ascending=False).head(4)
g["Kazakhstan"]                                  # 3,78
# несколько функций по годам
p.groupby("year")["res"].agg(["count", "median", "max"])
```

Год	count	median	max
1990	79	4,1	71,6
2000	79	3,8	70,8
2010	79	5,6	83,0
2020	79	10,6	86,1
2024	79	13,0	80,5

В метеорологии так же: среднее по месяцам, максимум по станциям, сумма осадков по годам.

Изменение с 2000 по 2024 год: метод pivot

```
p = c[c["year"] >= 1990]
wide = p.pivot(index="year", columns="country", values="res")
wide.shape                                  # (35, 79)
# loc — строка по метке (году)
change = wide.loc[2024] - wide.loc[2000]
change.sort_values(ascending=False).head(4)
change["Kazakhstan"]                        # -0,56
```

Результат

```
Denmark      35,0
Portugal     27,3
Lithuania    22,2
Germany      21,2
```

Вывод

Рост доли ВИЭ в большинстве стран

Важно

Разность двух процентов — процентные пункты (п. п.). Разность строк pivot-таблицы сразу даёт результат для всех стран.

График 1: одна страна

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(10, 5))
sns.lineplot(data=kz, x="year", y="res", marker="o")
plt.title("Доля ВИЭ, Казахстан (1990–2024)")
plt.xlabel("Год"); plt.ylabel("Доля ВИЭ, %")
plt.grid(); plt.show()
```

seaborn работает с таблицей

data, x, y — имена столбцов, а не массивы чисел.

Доля — это отношение: она падает, если потребление растёт быстрее выработки ВИЭ.

Данные: Our World in Data, Energy Dataset (github.com/owid/energy-data).

График 2: сравнение стран

```
names = ["Kazakhstan", "United States",
         "China", "Germany", "Brazil"]
sel = df[df["country"].isin(names) & (df["year"] >= 1990)]
plt.figure(figsize=(10, 5))
sns.lineplot(data=sel, x="year", y="res",
             hue="country", marker="o")
plt.legend(title="Страна"); plt.grid()
```

Совпадение названий

hue — столбец, по которому линии получают разные цвета.

Названия должны совпадать с данными: United States, а не USA.

Данные: Our World in Data, Energy Dataset. Названия стран на рисунке переведены.

Какой график выбрать?

Как менялось во времени?

Линейный график: одна линия на страну

У кого больше?

Столбчатая диаграмма: сравнение значений

Какие значения встречаются чаще?

Гистограмма: распределение величины

Как связаны две величины?

Диаграмма рассеяния: точка на каждую пару значений

Тип графика определяется вопросом, а не вкусом. Подписи осей и единицы — обязательны.

Типичные ошибки и их исправление

Ошибка	Что происходит	Как исправить
Агрегат принят за страну	Рейтинг «стран» содержит «Мир» и континенты	Фильтровать по iso_code
Пропуск заполнен без пометки	Искусственный «рекорд» в данных	Указывать метод заполнения в отчёте
Название не совпало	Страна пропала из графика	Проверить через unique()
Разность процентов названа %	Смешаны относительная и абсолютная величины	Писать «п. п.»
Группировка по неверному столбцу	Ответ на другой вопрос	Сначала записать вопрос, потом код

Заключение

1	2	3	4	5	6
Столбцы	**Строки**	**Пропуски**	**Порядок**	**Группы**	**График**
df[["a", "b"]]	условия, isin	isna, dropna, fillna	sort_values	groupby, agg, pivot	Matplotlib, seaborn

Главный вывод

Шесть операций pandas — выбор столбцов, фильтрация строк, обработка пропусков, сортировка, группировка и график — покрывают большую часть анализа таблиц. Агрегаты нужно отделять от стран, заполненный пропуск не является наблюдением, а тип графика выбирают под вопрос.

Вопросы для самопроверки

1. Почему результат фильтрации по country может включать «Мир»? Как это проверить?

2. Чем медиана по всем строкам отличается от медианы по одной стране и почему fillna этим опасен?

3. Как из длинной таблицы получить широкую и зачем это нужно при сравнении стран?

4. Какой тип графика выбрать, чтобы показать связь доли ВИЭ и выбросов CO_2?

Семинар 2 и следующая лекция

Семинар 2 · Выбросы CO_2 по странам

Лидеры по выбросам и изменение с 2000 года.

Те же шесть операций на данных Our World in Data: выбор, фильтр, пропуски, группировка, график.

Вопросы: кто лидирует? Где выбросы растут, а где снижаются?

Лекция 3

Статистическая связь между климатическими показателями.

Диаграмма рассеяния, корреляция Пирсона, линейная регрессия.

Литература

- Ritchie, H., Rosado, P., & Roser, M. Energy. Our World in Data. https://ourworldindata.org/energy
- Our World in Data. Energy Dataset. https://github.com/owid/energy-data
- Ritchie, H., Rosado, P., & Roser, M. CO_2 and Greenhouse Gas Emissions. Our World in Data. https://ourworldindata.org/co2-and-greenhouse-gas-emissions
- McKinney, W. (2022). Python for Data Analysis (3rd ed.). O'Reilly Media. https://wesmckinney.com/book/
- VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media. https://jakevdp.github.io/PythonDataScienceHandbook/
- Wickham, H. (2014). Tidy data. Journal of Statistical Software, 59(10), 1–23. https://doi.org/10.18637/jss.v059.i10
- Waskom, M. L. (2021). seaborn: statistical data visualization. JOSS, 6(60), 3021. https://doi.org/10.21105/joss.03021
- pandas development team. Documentation: Group by; Reshaping and pivot tables. https://pandas.pydata.org/docs/