Функции и группировки в pandas

Константин Башевой
Аналитик-разработчик, Яндекс

Константин Башевой
Аналитик-разработчик
Яндекс

Помогаю аналитикам с инфраструктурой

Собираю инструменты обработки данных

Рассказываю как это весело

Последние 10 лет:

- Rambler&Co
- Ростелеком
- Яндекс

Что сегодня будет

Программа на сегодня

Функции
любой сложности
для dataframe

GROUP BY
как устроена
эта операция

Функции в pandas

Встроенных методов не всегда достаточно

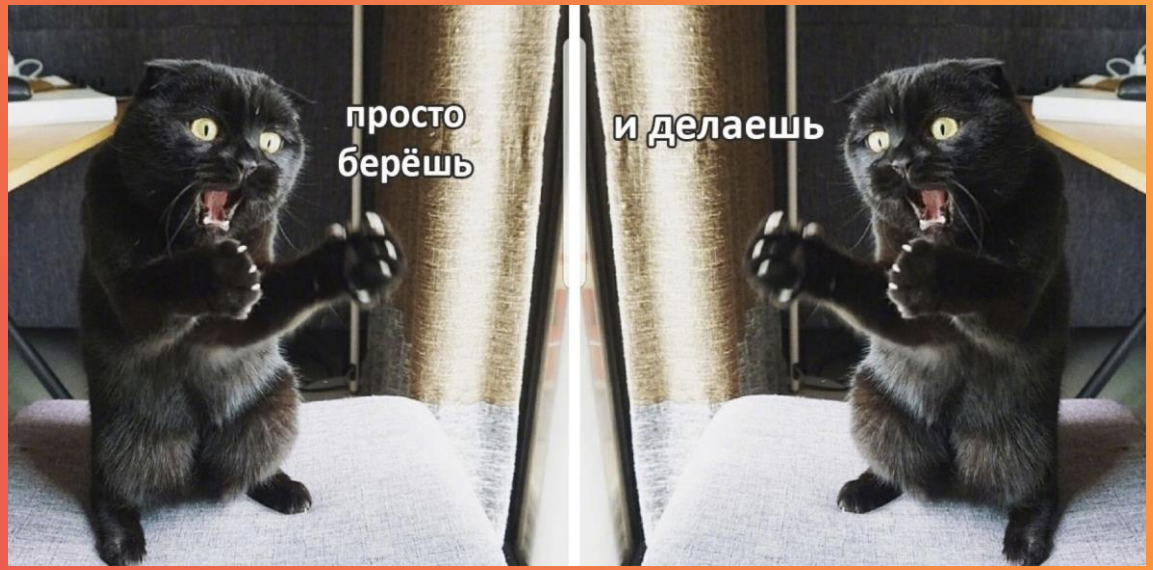
В одном столбце датафрейма есть ссылки:

https://awesome-site.ru/?utm_source=yandex&utm_medium=cpc&utm_campaign=a825749b87&utm_content=dev_{device_type}

Необходимо в отдельный столбец записать значение параметра utm_campaign

Датафрейм – это таблица

Как применить функцию к таблице?



Задачка

Имеется статистика переходов пользователей (user_id) в интернет-магазин (clicks) и заказов в нем (orders)

	user_id	clicks	orders
0	1	163	2
1	2	130	4
2	3	97	0

Задачка

Имеется статистика переходов пользователей (user_id) в интернет-магазин (clicks) и заказов в нем (orders)

	user_id	clicks	orders	calculated
0	1	163	2	False
1	2	130	4	False
2	3	97	0	True

Как посчитать произвольный столбец calculated?

Что будем считать

Если пользователь ничего не купил, то в calculated ставим True.

Если купил хотя бы раз, то ставим False.

	user_id	clicks	orders
0	1	163	2
1	2	130	4
2	3	97	0

```
def watcher(param):
    """Мне только посмотреть"""
    return param == 0
```

Метод apply

Метод apply – аналог цикла, который проходит по всем строкам датафрейма и применяет к каждой функцию watcher

Два режима использования:

1. В param передаются значения одного столбца
2. В param передается вся строка целиком

```
def watcher(param):
    """Мне только посмотреть"""
    return param == 0
```

Как применять функции к датафреймам

Вариант 1. В параметр param передаем столбец orders

	user_id	clicks	orders	watcher
0	1	163	2	False
1	2	130	4	False
2	3	97	0	True

```
df['watcher'] = df['orders'].apply(watcher)
```

```
def watcher(param):
    """Мне только посмотреть"""
    return param == 0
```

Как применять функции к датафреймам

Вариант 1. В параметр param передаем столбец orders

	user_id	clicks	orders	watcher
0	1	163	2	False
1	2	130	4	False
2	3	97	0	True

```
watcher(2) # False
```

```
watcher(4) # False
```

```
watcher(0) # True
```

```
df['watcher'] = df['orders'].apply(watcher)
```

А если функция использует несколько столбцов для вычислений?

	clicks	orders	user_id
0	163	2	1
1	130	4	2
2	97	0	3

```
def conversion(row):
    """Подсчет конверсии переходов в покупки"""
    return row['orders'] / row['clicks']
```

Как применять функции к датафреймам

Вариант 2. В параметр row передаем всю строку

	clicks	orders	user_id
0	163	2	1
1	130	4	2
2	97	0	3

```
df['conversion'] = df.apply(conversion, axis=1)
```

- axis=1 – в функцию будет передана строка
- axis=0 – будет передан столбец
- по умолчанию axis=0

Как применять функции к датафреймам

Вариант 2. В параметр row передаем всю строку

	clicks	orders	user_id	conversion
0	163	2	1	0.012270
1	130	4	2	0.030769
2	97	0	3	0.000000

```
row1 = pd.DataFrame({'clicks': [163],
                     'orders': [2],
                     'user_id': [1]})
conversion(row1)
```

```
0    0.01227
dtype: float64
```

```
df['conversion'] = df.apply(conversion, axis=1)
```

Как применять функции к датафреймам

Функция последовательно применяется ко всем строкам.

Название параметра не важно и не указывается в apply.

Применяя к Series, передаем значения одного столбца

```
df['watcher'] = df['orders'].apply(watcher)
```

Применяя к Dataframe – передаем всю строку.

Не забываем указывать axis=1

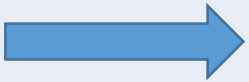
```
df['conversion'] = df.apply(conversion, axis=1)
```

Группировки

Группировка по столбцу

Группировка – подсчет определенной метрики для каждого уникального значения заданного столбца.

Номер заказа	Страна	Стоимость
1	Россия	100
2	Китай	80
3	Китай	90
4	Россия	140
5	Россия	90

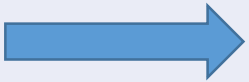


Страна	Метрика
Россия	…
Китай	…

Группировка по столбцу

Пример группировки с подсчетом суммы продаж по странам

Номер заказа	Страна	Стоимость
1	Россия	100
2	Китай	80
3	Китай	90
4	Россия	140
5	Россия	90

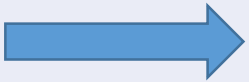


Страна	Сумма
Россия	330
Китай	170

Группировка по столбцу

Пример группировки с подсчетом средней стоимости продажи по странам

Номер заказа	Страна	Стоимость
1	Россия	100
2	Китай	80
3	Китай	90
4	Россия	140
5	Россия	90



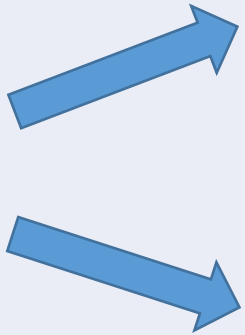
Страна	Среднее
Россия	110
Китай	85

Как это считается

Алгоритм расчета

Каждому уникальному значению свой датафрейм

Номер заказа	Страна	Стоимость
1	Россия	100
2	Китай	80
3	Китай	90
4	Россия	140
5	Россия	90



Номер заказа	Страна	Стоимость
1	Россия	100
4	Россия	140
5	Россия	90

Номер заказа	Страна	Стоимость
2	Китай	80
3	Китай	90

Алгоритм расчета

Применяем функцию (например, суммы) к каждому датафрейму

Номер заказа	Страна	Стоимость
1	Россия	100
4	Россия	140
5	Россия	90

Номер заказа	Страна	Стоимость
2	Китай	80
3	Китай	90

Алгоритм расчета

Применяем функцию (например, суммы) к каждому датафрейму

Номер заказа	Страна	Стоимость
1	Россия	100
4	Россия	140
5	Россия	90

apply

Страна	Сумма
Россия	330

Номер заказа	Страна	Стоимость
2	Китай	80
3	Китай	90

apply

Страна	Сумма
Китай	170

Алгоритм расчета

Применяем функцию (например, суммы) к каждому датафрейму

Номер заказа	Страна	Стоимость
1	Россия	100
4	Россия	140
5	Россия	90

apply

Страна	Сумма
Россия	330

Номер заказа	Страна	Стоимость
2	Китай	80
3	Китай	90

apply

Страна	Сумма
Китай	170

Страна	Сумма
Россия	330
Китай	170