Конспект по теме «Категоризация данных»

Знакомство с данными

Кроме внутренних преобразований важны и внешние. Необходимо привести данные в удобный и читаемый вид.

Одна из подзадач в этом процессе — именование столбцов. Для этого применяется метод `rename()`.

Классификация по типу

В датасетах могут встречаться категории в виде названий, их длина может быть различной.

К чему приводит такой способ хранения?

- Усложняется визуальная работа с таблицей;
- Увеличивается размер файла и время обработки данных;
- Чтобы отфильтровать данные по типу обращения, приходится набирать его полное название (а в нём можно допустить ошибки);
- Создание новых категорий и изменение старых отнимает много времени.

Для того, чтобы оптимально хранить информацию о категориях, создаётся отдельный файл-словарь, названию категории соответствует номер. Этот номер в дальнейшем используется вместо текстового наименования категории в таблице.

Классификация по возрастным группам

Часто объекты, имеющие определённое значение признака, присутствуют в наборе только единожды. Работать с такими отрывками и делать из них

статистические выводы нельзя. Поэтому, с такими данными проводится **категоризация** — объединение данных в категории.

Одним из вариантов категоризации является выделение возрастных групп, например: до 18, от 18 до 65, старше 65.

Подобные правила классификации в Python удобно представлять в виде функций, которые принимают параметр — значение признака, а возвращают соответствующую категорию.

Для того, чтобы получить столбец с группой на основе столбца с каким-то другим признаком, можно воспользоваться написанной нами функцией `group` и методом `apply()`.

```
data['column_group'] = data['column'].apply(group)
```

Функция для одной строки

Когда для категоризации недостаточно значения в одном каком-то столбце, то в функцию можно передать и всю строку как *Series*. В теле этой функции можно также получать значения в каком-то определённом столбце.

Применение метода apply() в случае обработки строки имеет два отличия:

1) Метод `apply()` вызывается не к столбцу data`['age']`, а к датафрейму `data`;

2) По умолчанию Pandas передаёт в функцию `group()` столбец. Чтобы на вход в функцию отправлялись строки, нужно указать параметр `axis = 1` метода `apply()`.