Конспект по теме "Работа с пропусками"

Метрики эффективности источника трафика

Теория

Существует множество **источников трафика**:

- Поисковая выдача (органический трафик)
- Контекстная реклама
- E-mail рассылка
- Социальные сети
- Переходы по ссылкам с других сайтов

Цель введения метрик эффективности источника трафика - оценка и сравнение источников трафика для выявления лучших. Знание эффективности источников позволяет оперативно управлять маркетинговой стратегией.

Имеется две методики подсчёта **конверсии сайтов**. Первая - считается **доля посетителей** сайта, совершивших **целевые действия**. Вторая - считается **доля целевых действий**.

Визит - последовательность действий посетителя от перехода на сайт и до момента, когда пользователь ничего больше не делал в течение 30 минут.

Ещё одна важная метрика для определения эффективности источника трафика - **доля повторных покупателей.** Эта метрика рассчитывается как отношение числа посетителей, совершивших хотя бы две покупки, к числу посетителей, совершивших хотя бы одну покупку.

Практика

В pandas можно делать **арифметические операции** над столбцами: сложение, вычитание, умножение, деление. Например:

```
data['column1'] = data['column12'] + data['column3']
```

User ID и куки

Теория

Информацию о поведении посетителей веб-страницы собирает специальный **счётчик** — несколько строчек кода в коде сайта — и отправляет её в **системы веб-аналитики**, например, Яндекс.Метрику. Счётчик собирает: общие сведения о посетителях, с какого источника трафика они заходят, просмотр конкретных страниц пользователем и покупки. В счётчиках каждому пользователю присваивается **уникальный номер**, который нужен чтобы отличить пользователя от остальных - **User ID**.

Данные счётчиков - "сырые", они затем с помощью систем веб-аналитики превращаются в отчёты об аудитории, посещаемости и источниках. В **отчётах** можно комбинировать разные метрики и визуализировать результаты.

Для определения пользователя, который повторно зашёл на сайт, применяются **куки** - специальные текстовые файлы, которые остались в памяти устройства после первого посещения и при повторном визите отправляются на сервер.

Но если пользователь заходит с разных браузеров, из-за того что у каждого браузера свои куки, ему присваиваются разные User ID, поэтому собираются дополнительные данные, например, e-mail. Для защиты персональных данных, User ID и email зашифровываются.

Текстовые файлы с информацией о посещении сайта называются **логами**.

Когда к набору полученных определённым образом данных добавляют новую информацию, это называется **обогащение данных**. В датасете

могут встречаться пропущенные данные. Иногда, их можно проигнорировать, а иногда нужно их обработать, заполнить для анализа.

Практика

Для поиска уникальных значений в столбце, применяется метод `unique()` : `data['column'].unique()`.

Для удаления строк с пропущенными значениями нужно вызвать метод `dropna()`, а для перенумерации - `reset_index()` с аргументом `drop=True`.

Вы обнаружили *NaN и None*

Теория

NaN и *None* - эти особые значения указывают, что никакого значения нет. *NaN* отвечает за отсутствующее в ячейке число. Его тип данных *float*, поэтому с *NaN* можно проводить математические операции. *None* принадлежит к нечисловому типу *NoneType,* и математические операции с ним неосуществимы. Значения *NaN* могут привести к некорректным результатам при группировке данных. Строки с этими значениями не всегда стоит удалять: часто пропуски можно восстановить.

Практика

В pandas, метод `value_counts()` возвращает уникальные значения с их количеством.

Метод `isnull()` возвращает булевский список, в котором `True` означает, что значение в колонке пропущено.

Для замены пропусков на какое-то значение, применяется метод `fillna()` с аргументом `value`.

Категориальные и количественные переменные

Переменные бывают двух типов: категориальные и количественные. **Категориальная переменная** принимает одно значение из ограниченного

набора, а **количественная** — любое числовое значение в диапазоне. Количественные переменные, в отличие от категориальных. обладают возможностью сравнения.

Также переменные могут быть **логическими (булевыми)**. Такие переменные указывают на истинность или ложность какого-либо события. Если событие истинно, то переменная принимает значение 1, соответствующее True, а если ложно — 0, соответствующий False.

Работа с пропусками в категориальных переменных

Теория

Перед обработкой пропусков, нужно ответить на вопрос, существует ли *закономерность* в появлении пропусков. Иными словами, *не случайно ли* их возникновение в наборе данных.

Пропуски бывают трёх типов:

- **Полностью случайные**: если вероятность встретить пропуск не зависит ни от каких других значений. Ответ на этот вопрос не зависит от характера самого вопроса и от других вопросов анкеты, а сам пропуск легко восстановить по имени.
- **Случайные:** если вероятность пропуска зависит от других значений в наборе данных, но не от значений собственного столбца. Пропущенное значение связано с тем, что, например, такой категории не существует.
- **Неслучайные:** если вероятность пропуска зависит от других значений, в том числе и от значений собственного столбца. Отсутствующее значение зависит как от характера вопроса, так и от значения переменной в другом столбце.

Практика

Существует несколько вариантов замены пропусков категориальных значений. Например, замена значением по умолчанию. Такой вариант хорошо подойдёт для заполнения случайных пропусков. В pandas для этого применяется метод `fillna()`.

Не все пустые значения можно заполнить методом `fillna()`. Например, к пропущенным значениям *None* его не применить — метод распознаёт только значения *NaN* в таблице. Для замены *None* вызывают метод `loc`. Логическая индексация позволит выделить все строки в необходимом столбце, которые содержат *None*, и заменить их на новое значение.

Для применения некоторых функций к определённым столбцам, применяется метод `agg()`. Название столбца и сами функции записываются в структуру данных — **словарь**. Словарь состоит из **ключа** и **значения**. Ключ - это название столбца, к которому нужно применить функции, а значением выступает список с названиями функций.

```
{'column':['function1','function2']}
```

После применения метода `agg()` , названия столбцов стали «двойными». Чтобы обратиться к результату применения функции `['function1']` к столбцу `['column']`, просто укажите их подряд:

```
data['column']['function1']
```

Работа с пропусками в количественных переменных

Теория

Пропуски в количественных переменных заполняют *характерными значениями*. Это значения, характеризующие состояние **выборки**, - набора данных, *выбранных* для проведения исследования. Чтобы примерно оценить типичные значения выборки, годятся **среднее арифметическое** или **медиана**.

Среднее арифметическое — это сумма всех значений, поделённая на количество значений.

Медиана — это такое число в выборке, что ровно половина элементов больше него, а другая половина — меньше.

Практика

Для получения среднего арифметического применяется метод `mean()`. Его применяют ко всей таблице, к отдельному столбцу или к сгруппированным данным.

Для нахождения медианы есть специальный метод `median()`, его можно применять к таблице, столбцу или сгруппированным данным.