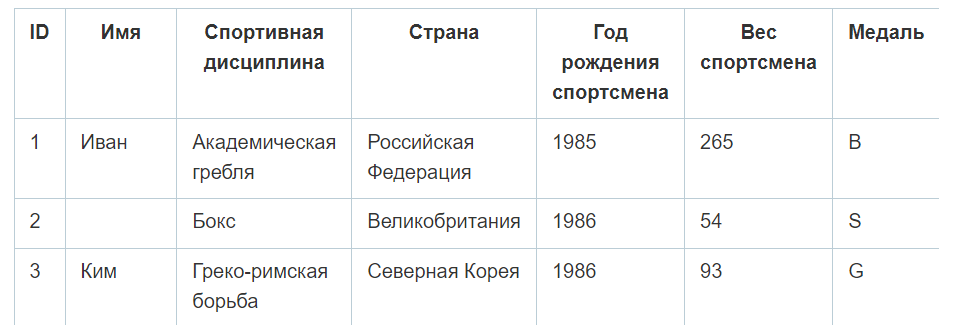
**Лекция №3. Предварительная обработка данных.**

Признаки (значения датасета) можно поделить **на категориальные** и **численные**.

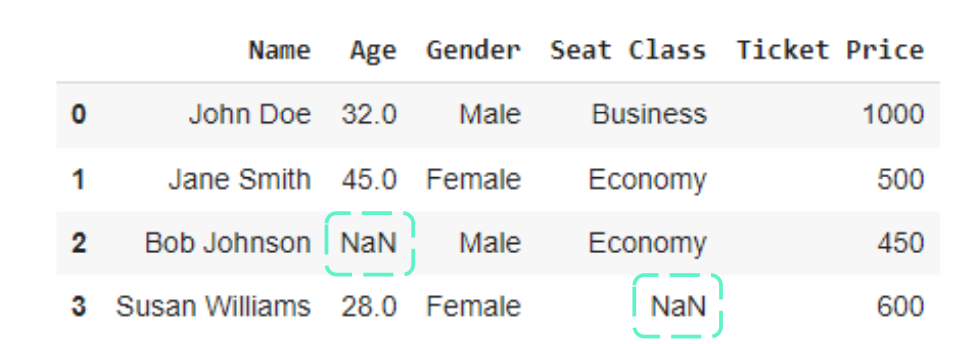


Категориальный признак – признак, значение которого можно отнести к какой-либо группе, но порядок значений в этой группе абсолютно не важен. Более того, между значениями категориальных признаков невозможно установить отношения "больше" или "меньше". Однако, никто не мешает исследователю отсортировать значения признака, например, по алфавиту для наглядности или по какому-либо другому критерию для какой-либо другой цели. Примеры категориальных признаков в моем наборе данных: Имя, Спортивная дисциплина, Страна, Год рождения спортсмена.

Значения численного признака является скаляром. Объекты в наборе данных можно упорядочить по значению такого признака в возрастающем или убывающем порядке. В качестве примеров численного признака можно привести Год рождения спортсмена и Вес спортсмена.

Год рождения спортсмена как к численным, так и к категориальным признакам. Дело в том, что многие признаки можно отнести одновременно к двум категориям. Но в модели такие признаки будут использоваться вполне однозначно: либо как численные, либо как категориальные. Классификация таких спорных признаков –творческий процесс и зависит от задачи, которую вы решаете.

Процесс, который должен выполняться в самом начале, – это **обработка пропущенных значений**. Выявление и удаление недостающих значений – основная процедура для моделей машинного обучения (ML). Существует два основных типа пропущенных значений: типы данных NaN и Null. Поскольку они могут быть вредными для моделей машинного обучения, эти переменные необходимо каким-то образом обрабатывать.



Пути решения:

• Удаление строк или столбцов: В Pandas, для удаления строк с отсутствующими значениями, можно использовать df.dropna(), а для удаления столбцов – df.dropna(axis=1).

• Заполнение средними значениями: Пропущенные значения можно заменить средними или медианными значениями из соответствующего столбца. В Pandas, для заполнения отсутствующих значений средними значениями можно использовать df.fillna(df.mean()).

• Интерполяция: Для **временных рядов** и числовых данных вы можете использовать метод интерполяции для заполнения отсутствующих значений на основе соседних значений (метод Sklearn SimpleImputer () библиотеки Sklearn).



Временные ряды

**Масштабирование функций**

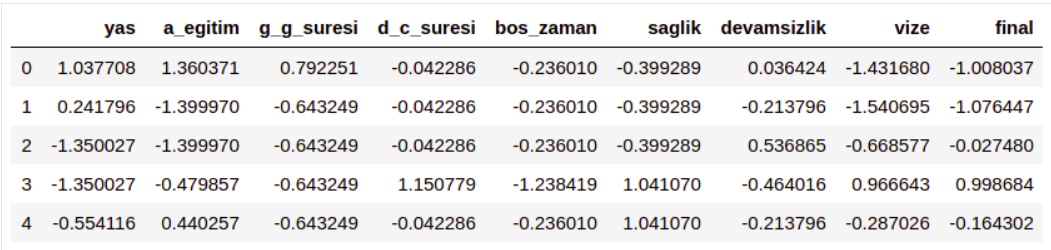
Масштабирование функций – еще один важный метод предварительной обработки. Цель состоит в том, чтобы уместить данные в определенный диапазон. Это обязательный метод, если существует огромная разница в порядке величин признаков в наборе данных, поскольку признаки больших величин могут доминировать над признаками меньших величин. Масштабирование функций может быть достигнуто двумя способами:

* стандартизация;
* нормализация.

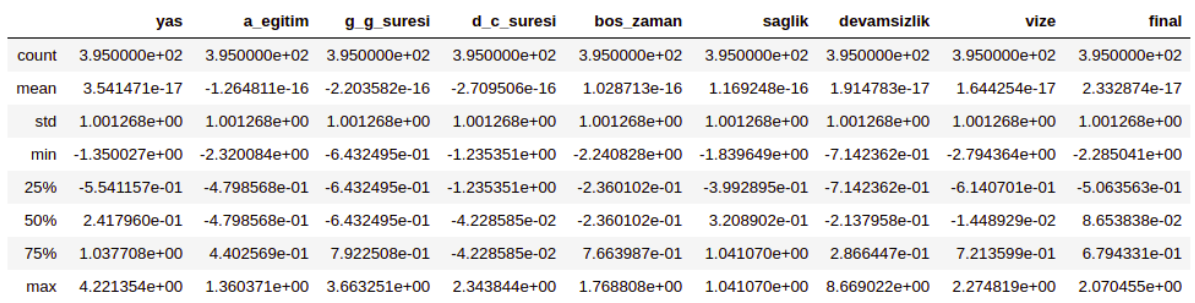
Поскольку масштабирование функций может применяться к числовым столбцам, нам нужно извлечь числовые столбцы из набора данных.



Как видно из рисунка выше, в то время как некоторые столбцы содержат небольшие целочисленные значения, некоторые содержат большие числовые значения. Поэтому было бы лучше, если бы масштабирование функций можно было выполнять в наборе данных. В этом примере применение масштабирования функций может быть не критично, но это повысит точность модели машинного обучения. Поскольку веса в моделях машинного обучения зависят от функций, огромный пробел или разница в значениях функций может упростить модель машинного обучения.



Как видно на рисунке выше, числовой набор данных стандартизирован. Если мы хотим проверить, что стандартизация выполнена правильно, мы можем использовать метод describe (), чтобы увидеть значения среднего и стандартного отклонения. Среднее значение каждого столбца почти равно 0, а значение стандартного отклонения - почти 1. Причина, по которой они не равны точно 0 и 1, заключается в том, что во время вычислений могли возникнуть ошибки усечения и округления.



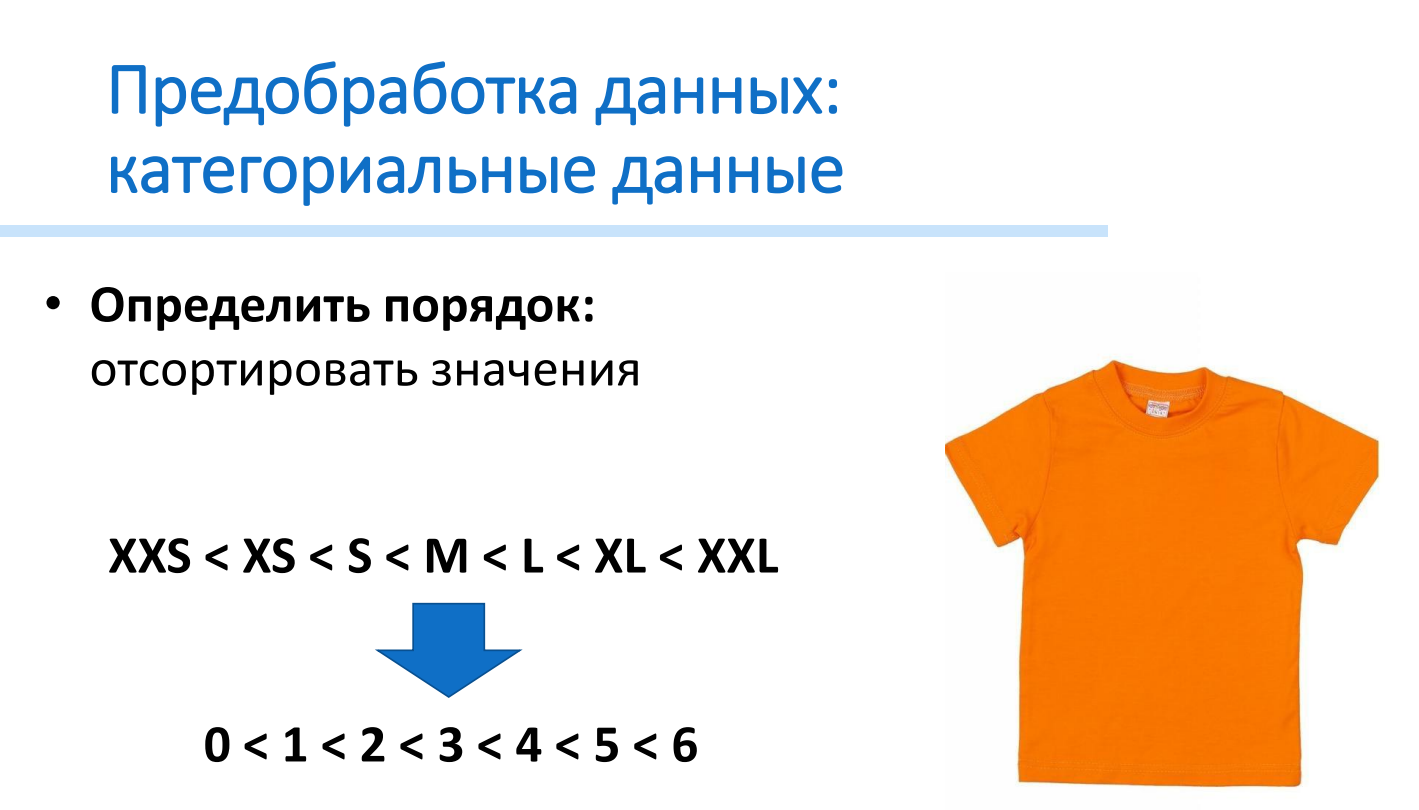
Нормализация – это приведение всех значений признака к новому диапазону. Например, к диапазону [0, 1] (бинаризация). Это полезно, поскольку значения признаков могут изменяться в очень большом диапазоне. Причем, значения разных признаков могут отличаться на несколько порядков. А после нормализации они все будут находиться в узком (и, часто, едином) диапазоне.

Нормализация: распределение данных центрируется вокруг нуля.

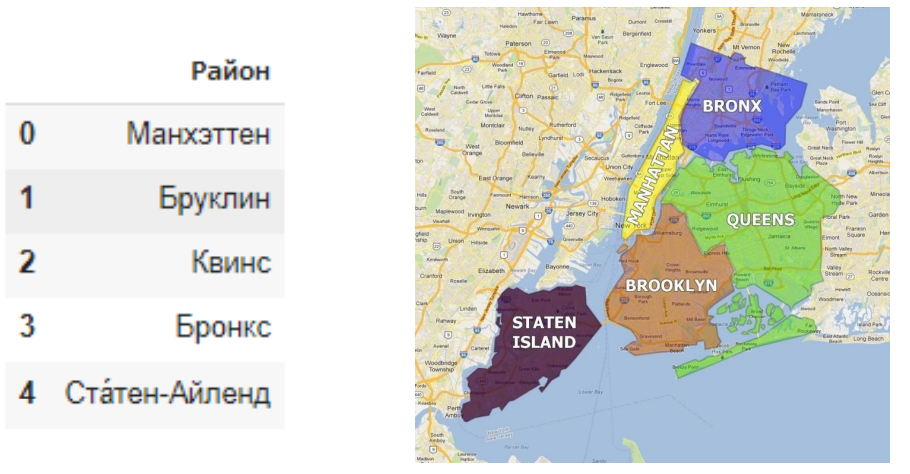
**Предобработка данных: числовые данные**

Стандартизация: распределение данных центрируется вокруг нуля и стандартное отклонение равно единице.

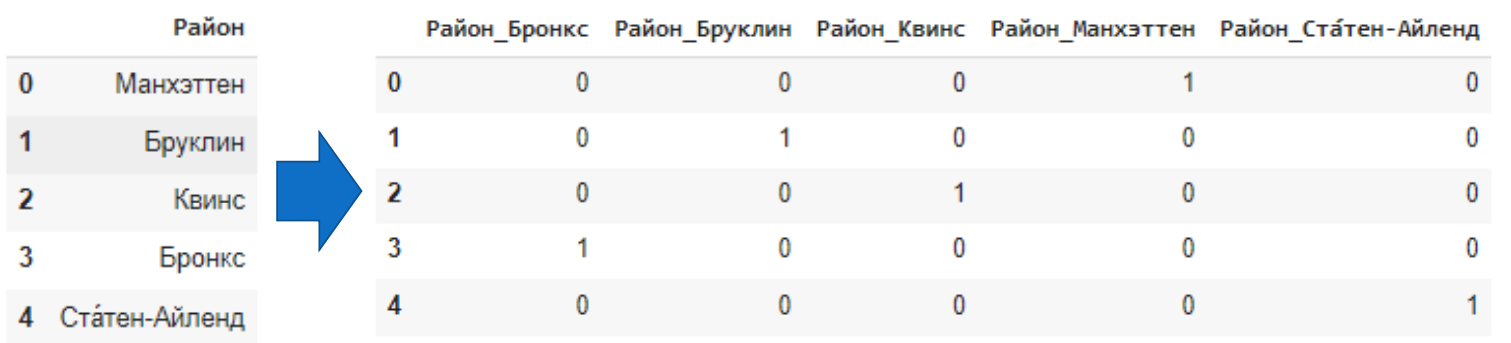
**Предобработка данных: категориальные данные**



Однако, как поступать с признаками, значения которых нельзя выразить численно? Можно заменить их значения численным идентификатором?



Например, вместо значения "Манхэттен" использовать значение 1, а вместо "Бруклин" – 2. Тогда линейная модель будет работать. Но, если поступить таким образом, будет утеряно свойство категориальности признака. Иными словами, модель будет пытаться сравнивать идентификаторы признаков между собой. Но они не сравнимы по значению.

Чтобы бороться с этой проблемой, был придуман способ преобразовать исходный признак в несколько новых, бинарных признаков, называемый One-hot encoding. Например, можно признак "Район" превратить в 5 новых бинарных признаков следующим образом: 

Для каждого из этих признаков линейная модель будет использовать независимый коэффициент и, таким образом, не будет сравнивать категориальные значения одного и того же признака между собой.