Масштабирование данных

Масштабирование данных — это процедура, которая приводит значения признаков к одному диапазону, чтобы каждый признак был одинаково важен.

- При масштабировании закономерности в самих признаках сохраняются.
- Масштабирование применяется к количественным данным.

Для каждого алгоритма машинного обучения масштабирование нужно по разным причинам. Для линейной регрессии это важно для правильной интерпретации весов модели.

Основные методы масштабирования

- Стандартизация. При стандартизации признаков их стандартное отклонение становится равным 1, среднее значение — 0.

$$x_{standart} = \frac{x_i - \bar{x}}{\sigma_x},$$

где:

- x_i — значение признака,
- $\bar{x}$ — среднее значение признака,
- σ_x — стандартное отклонение значения x.

```
# импорт функции
from sklearn.preprocessing import StandardScaler

# выбор метода масштабирования
scaler = StandardScaler()

# настройка масштабирования на тренировочной выборке:
# вычисление среднего и стандартного отклонения
scaler.fit(X_train)

# масштабирование тренировочной выборки
X_train_scaled = scaler.transform(X_train)

# масштабирование тестовой выборки
X_test_scaled = scaler.transform(X_test)
```

- **Нормализация**. Нормализованные признаки лежат в диапазоне от 0 до 1, но при этом их среднее значение может отличаться.

$$x_{normaliz} = \frac{x_i - x_{min}}{x_{max} - x_{min}},$$

где:

- x_i — значение признака;
- x_{min} — минимальное значение признака;
- x_{max} — максимальное значение признака.

```
# импорт функции
from sklearn.preprocessing import MinMaxScaler

# выбор метода масштабирования
scaler = MinMaxScaler()

# настройка масштабирования на тренировочной выборке
scaler.fit(X_train)

# масштабирование тренировочной выборки
X_train_scaled = scaler.transform(X_train)

# масштабирование тестовой выборки
X_test_scaled = scaler.transform(X_test)
```

Кодирование данных

Кодирование приводит категориальные признаки к понятным модели стандартам. Большинство моделей не умеет напрямую работать с номинальными, порядковыми и бинарными данными, поэтому их нужно закодировать как количественные.

Ключевой метод для кодирования — **One-Hot Encoding**. Задумка — отобразить свойство объекта как набор бинарных признаков со значениями 0 и 1. Результаты этого преобразования можно записать в виде вектора из нулей и единиц.

Подготовка данных

Исходные значения категориального признака:

	Название фруктов
1	Яблоко
2	Груша
3	Банан
4	Яблоко
5	Банан

Значения категориального признака после кодирования:

	Название фруктов	ohe_Яблоко	ohe_Груша	ohe_Банан
1	Яблоко	1	0	0
2	Груша	0	1	0
3	Банан	0	0	1
4	Яблоко	1	0	0
5	Банан	0	0	1

```
# импорт функции
from sklearn.preprocessing import OneHotEncoder

# выбор метода кодирования, его инициализация
encoder = OneHotEncoder()

# обучение кодирования на тренировочных данных
encoder.fit(X_train)

# кодирование тренировочной выборки
X_train_ohe = encoder.transform(X_train)

# кодирование тестовой выборки
X_test_ohe = encoder.transform(X_test)
```