

Основы искусственного интеллекта. Лекция 3. Введение в машинное обучение с Python. Основные концепции машинного обучения, введение в библиотеки Scikit-learn

Преподаватель: **Қасенхан Арай Мейрамбайқызы, доктор PhD, ассоциированный профессор кафедры «Программная инженерия»**

a.kassenkhan@satbayev.university

Содержание

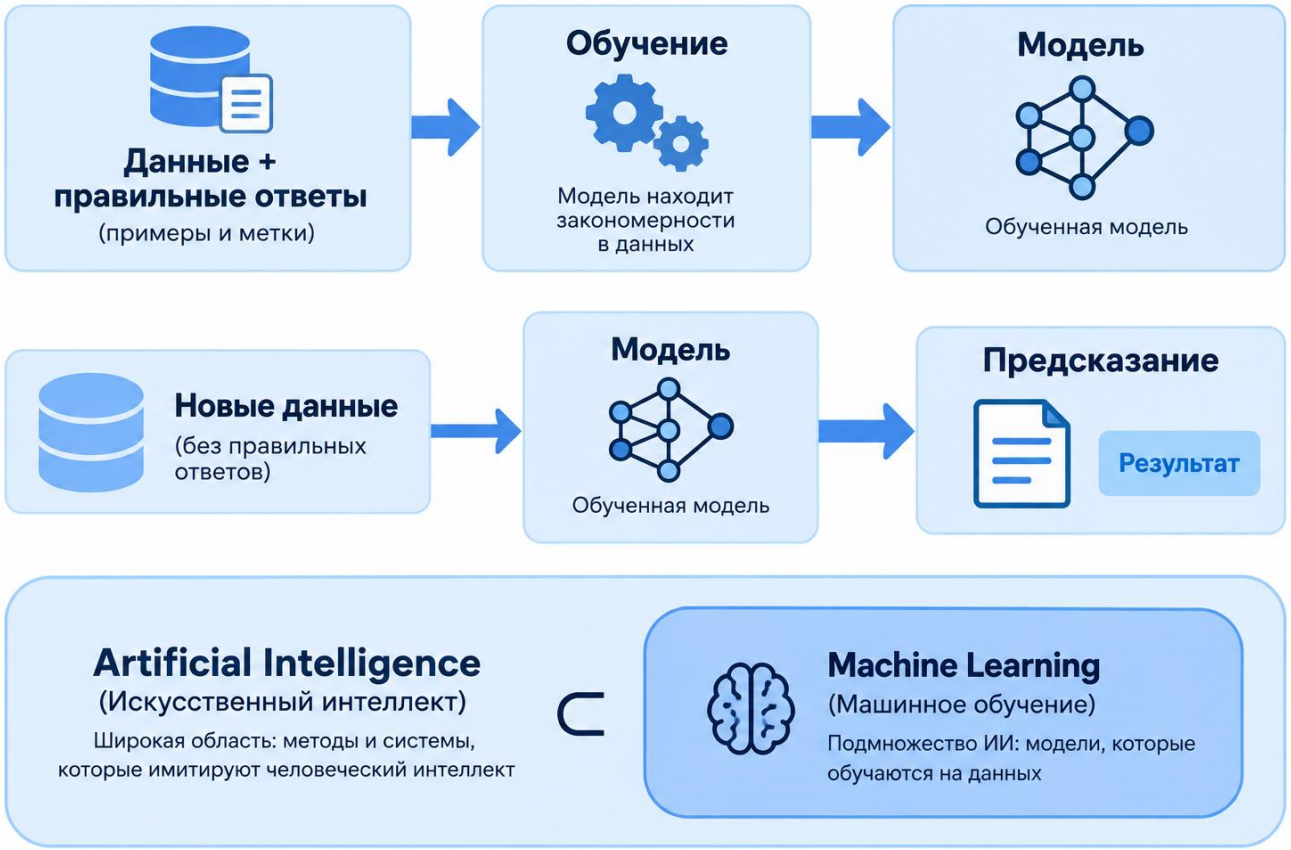
1. Что такое машинное обучение
2. Основные типы задач ML
3. Этапы машинного обучения
4. Scikit-learn
5. Первая ML-модель
6. Оценка результата

По завершению урока Вы будете знать:

1. **Как устроен полный процесс машинного обучения;**
2. **Чем отличаются классификация, регрессия и кластеризация;**
3. **Зачем разделять данные на обучающую и тестовую выборки;**
4. **Как обучить модель с помощью Scikit-learn;**
5. **Как получить прогноз и оценить качество модели.**

Что такое машинное обучение?

Машинное обучение (ML) — подход, при котором модель обучается на данных, выявляет закономерности и использует их для прогнозирования.



Основные задачи машинного обучения

Классификация:

Предсказать категорию

Пример: Spam / Not Spam

Регрессия:

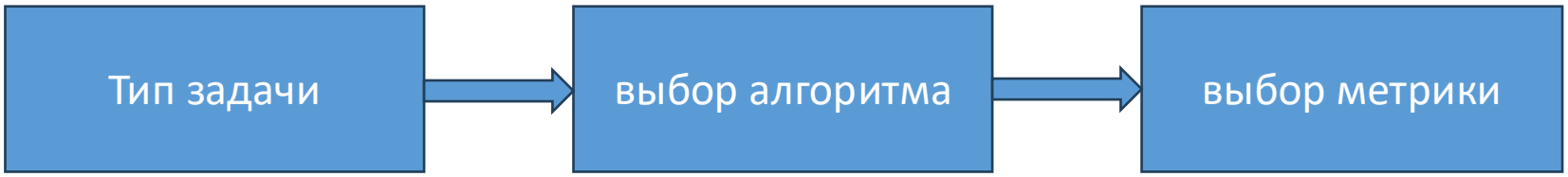
Предсказать числовое значение

Пример: Характеристики → Цена

Кластеризация:

Найти группы похожих объектов

Пример: Клиенты → Группы



Этапы машинного обучения



Scikit-learn — библиотека машинного обучения для Python

- Classification
- Regression
- Clustering
- Preprocessing
- Model Selection
- Evaluation

```
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

model.score(X_test, y_test)
```



Данные для модели: X и y

Используем встроенный Iris Dataset:

```
from sklearn.datasets import load_iris

iris = load_iris()

X = iris.data
y = iris.target



print(X.shape)   # (150, 4)
print(y.shape)   # (150,)
```

X — признаки (features)

- sepal length
- sepal width
- petal length
- petal width

y — целевая переменная (target)

- Setosa
- Versicolor
- Virginica

Разделение данных: Train и Test




```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42
)
```

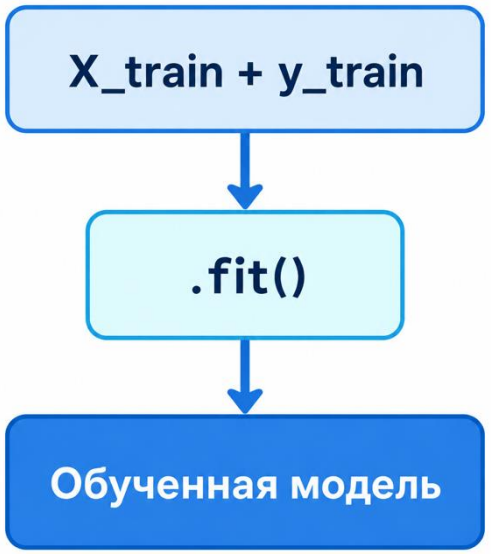
 # Test — данные, которые модель не видит при обучении.

Обучение модели

```
from sklearn.neighbors import KNeighborsClassifier

model = KNeighborsClassifier(n_neighbors=3)

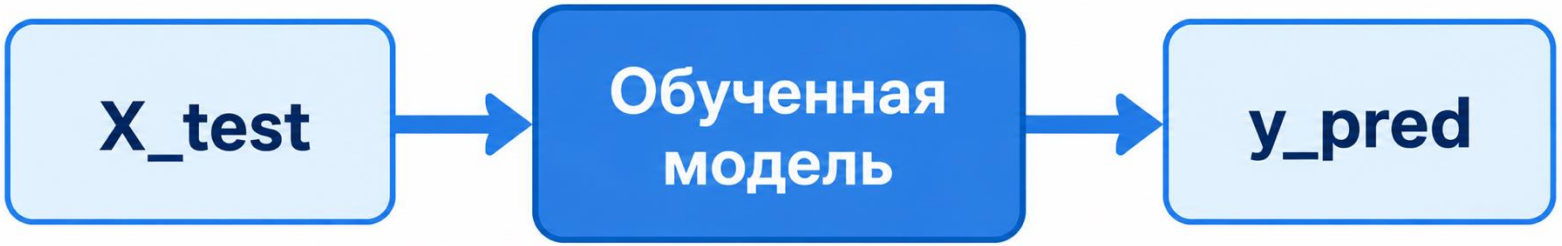
model.fit(X_train, y_train)
```



n_neighbors=3 → учитываем 3 ближайших объекта

Получаем предсказание

```
y_pred = model.predict(X_test)
```




```
print(y_test[:5])

print(y_pred[:5])
```

.fit() → обучение
.predict() → прогноз

Оценка модели

```
from sklearn.metrics import accuracy_score

accuracy = accuracy_score(y_test, y_pred)

print("Accuracy:", accuracy)
```

Классификация:

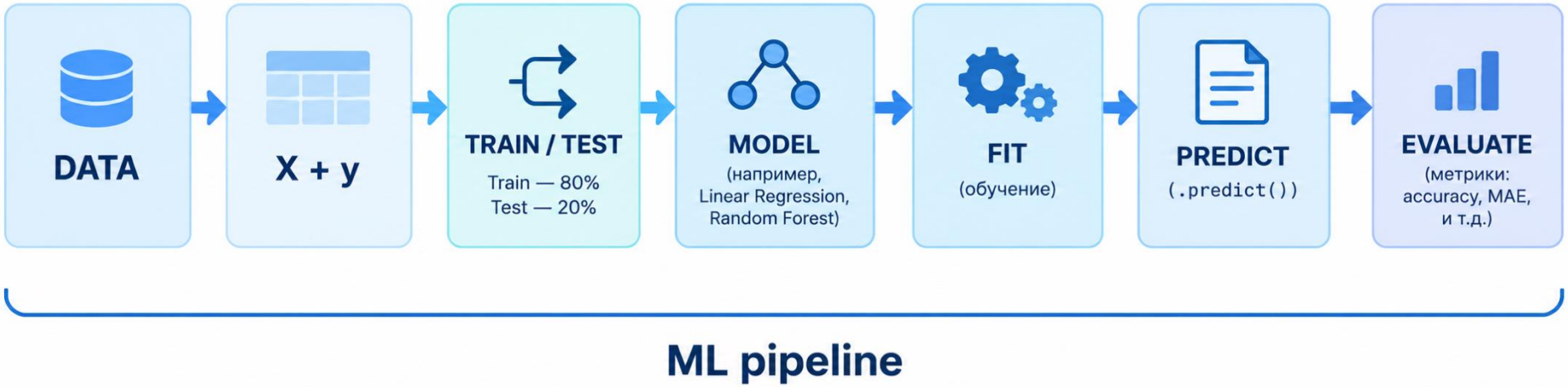
- Accuracy
- Precision
- Recall
- F1-score

Регрессия:

- MSE
- R^2

Метрика выбирается в зависимости от задачи.

Первая ML-модель: весь процесс



Первая ML-модель: код

```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score

iris = load_iris()
X, y = iris.data, iris.target

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

model = KNeighborsClassifier(n_neighbors=3)
model.fit(X_train, y_train)

y_pred = model.predict(X_test)

print(accuracy_score(y_test, y_pred))
```

Заключение

Машинное обучение — это не только выбор алгоритма. Это последовательный процесс от подготовки данных до оценки результата.