Практическая работа №8

# Обучение с учителем: Дерево решений

Обучение с учителем - это процесс обучения компьютера на основе данных, которые уже размечены, то есть имеют известные правильные ответы. По сути, это как учить школьника на примерах, где учителем выступает компьютер.

Дерево решений - это инструмент в машинном обучении, который помогает решать проблемы, когда нужно сделать выбор из нескольких вариантов.

Можно представить дерево решений как игру, где вы задаете вопросы и получаете ответы, пока не найдете ответ на свой вопрос. Каждый раз, когда вы задаете вопрос, вы выбираете, какой вопрос задать дальше, основываясь на предыдущих ответах. В конце концов, вы получаете ответ на свой вопрос.

Дерево решений использует такую же логику. Оно задает ряд вопросов, основываясь на имеющихся данных, и на основе ответов на эти вопросы принимает решение.

*Рассмотрим пример использования дерева решений для определения, является ли животное кошкой или собакой.  
Для начала дерево решений может задавать вопросы, например: "Животное маленькое или большое?", "У животного есть уши висячие или стоячие?", "Хвост животного длинный или короткий?".  
Если животное маленькое, у него есть уши висячие и хвост короткий, то дерево решений может определить, что это кошка. Если животное большое, у него есть уши стоячие и хвост длинный, то дерево решений может определить, что это собака.  
Дерево решений используется для классификации животных и на основе заданных вопросов определяет, к какому классу они относятся.*

Из одного узла в решающем дереве может выходить более двух ветвей. В решающем дереве каждый узел представляет собой тест на одном из признаков, а ребра, исходящие из узла, представляют возможные значения этого признака. Если признак имеет более двух возможных значений, то из узла будет выходить столько ребер, сколько возможных значений у этого признака. Например, если признаком является цвет, который может быть "красный", "зеленый" или "синий", то из узла будет выходить три ребра.

# Пример обучение модели линейной регрессией

Рассмотрим пример машинного обучения решающим деревом на датасете Iris.

from sklearn.datasets import load\_iris  
from sklearn.tree import DecisionTreeClassifier  
from sklearn.model\_selection import train\_test\_split  
from sklearn.metrics import accuracy\_score  
# Загрузка данных  
iris = load\_iris()  
X = iris.data  
y = iris.target  
# Разбиение данных на обучающую и тестовую выборки  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)  
# Создание модели обучающего дерева  
clf = DecisionTreeClassifier(max\_depth=2)  
clf.fit(X\_train, y\_train)  
# Предсказание на тестовых данных и вычисление точности  
y\_pred = clf.predict(X\_test)  
acc = accuracy\_score(y\_test, y\_pred)  
print('Точность на тестовых данных: {:.2f}'.format(acc))

Этот код загружает набор данных Iris, разделяет его на обучающую и тестовую выборки, создает объект классификатора решающего дерева с максимальной глубиной равной 2 и обучает модель на обучающих данных. Затем оценивается точность модели на тестовых данных и выводится на экран. Обратите внимание, что параметры модели (например, максимальная глубина дерева) могут быть настроены для улучшения ее качества.

*Максимальная глубина (max\_depth) - это параметр решающего дерева, который определяет максимальную глубину дерева, то есть максимальное количество узлов от корня до листового узла. Если значение параметра max\_depth равно None (по умолчанию), то дерево будет строиться до тех пор, пока каждый листовой узел не будет содержать только объекты одного класса или пока все листовые узлы не будут содержать меньшее количество объектов, чем заданное минимальное значение (min\_samples\_leaf).*

*Максимальная глубина решающего дерева влияет на сложность модели и ее склонность к переобучению. Если максимальная глубина слишком большая, модель может подстроиться под обучающие данные слишком тесно и переобучиться, что приведет к плохой обобщающей способности модели на новых данных. Если максимальная глубина слишком мала, модель может быть недостаточно сложной для представления сложных зависимостей в данных. Поэтому выбор оптимальной максимальной глубины дерева является важным шагом при обучении решающих деревьев.*

*Склонность к переобучению в машинном обучении проявляется, когда модель слишком точно подстраивается под обучающие данные, в результате чего ее способность обобщать данные, которые она не видела ранее, снижается. Вместо обобщения модель начинает запоминать шумы и особенности в обучающих данных, что приводит к ухудшению ее качества на новых данных.*

*Существует несколько признаков, которые могут свидетельствовать о склонности модели к переобучению:*

* *Переобученная модель имеет очень высокую точность на обучающих данных, но плохо работает на тестовых данных.*
* *Различие между точностью на обучающих и тестовых данных слишком большое.*
* *Модель имеет очень большое число параметров, а количество данных для обучения невелико.*
* *Модель имеет высокую сложность и низкую интерпретируемость.*

*Для борьбы с переобучением можно использовать различные методы, такие как регуляризация, уменьшение сложности модели, использование ансамблевых методов, аугментация данных и т.д.*

# Задание

Создать модель обучения дерева решений для датасет breast\_cancer и вычислить точность данной модели на тестовых данных.

1. загрузить датасет breast\_cancer
2. разбить данный датасет на обучающую и тестовую выборки
3. создать модель обучающего дерева, и обучить решающее дерево на обучающей выборке, с применением параметра max\_depth=3
4. создать предсказания на тестовой выборке
5. вычислить точность предсказания с помощью метрики accuracy\_score из библиотеки Scikit-learn

Ответ необходимо предоставить в виде файла \*.ipynb