Лекция 7. Методы деревьев решений

Тема: CART, ID3, C4.5, Random Forest, Gradient Boosted Trees

---

1. Введение

Методы деревьев решений — это один из наиболее популярных и понятных подходов в анализе данных и машинном обучении.
Они применяются для задач **классификации**, **регрессии**, **отбора признаков** и **интерпретации моделей**.

Основная идея метода заключается в том, чтобы разбивать пространство признаков на подмножества, принимая решения в виде дерева, где каждый узел соответствует вопросу о значении признака, а листья — итоговым решениям или классам.

Пример:
Если доход < 200 000 → «отказать в кредите»,
иначе → если возраст < 25 → «проверить дополнительно»,
иначе → «одобрить кредит».

---

2. Принцип построения дерева решений

2.1. Основная идея

Алгоритм строит дерево сверху вниз, выбирая на каждом шаге **признак**, который лучше всего разделяет данные на подмножества с разными значениями целевой переменной.

Каждый **внутренний узел** содержит условие (например, «доход > 150 000»), а каждый **лист** содержит решение — предсказание класса или числового значения.

Процесс построения дерева называют **обучением (training)**, а процесс принятия решения — **инференцией (inference)**.

2.2. Критерии разбиения

При построении дерева выбирается такой признак и порог, которые дают наилучшее разделение данных.
Для этого используется **мера неоднородности (импурити)** подмножеств.

Основные меры:

1. **Энтропия (Entropy):**

   H(S)=−∑i=1kpilog⁡2piH(S) = - \sum_{i=1}^k p_i \log_2 p_iH(S)=−i=1∑kpilog2pi

   где pip_ipi — доля объектов класса iii.
   Чем меньше энтропия, тем более однородно множество.

2. **Индекс Джини (Gini):**

   G(S)=1−∑i=1kpi2G(S) = 1 - \sum_{i=1}^k p_i^2G(S)=1−i=1∑kpi2

   Он показывает вероятность ошибочной классификации случайно выбранного объекта.

3. **Информационный прирост (Information Gain):**

   IG(S,A)=H(S)−∑v∈Values(A)|Sv||S|H(Sv)IG(S, A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} H(S_v)IG(S,A)=H(S)−v∈Values(A)∑|S||Sv| H(Sv)

   — разность между исходной энтропией и энтропией после разбиения по признаку A.

---

3. Алгоритм ID3 (Iterative Dichotomiser 3)

3.1. История и идея

Алгоритм ID3 был предложен **Россом Куинланом** в 1986 году.
Это одна из первых и наиболее известных реализаций построения деревьев решений.

3.2. Принцип работы

1. Вычисляется **энтропия исходного множества**.
2. Для каждого признака вычисляется **информационный прирост**.
3. Выбирается признак с **максимальным приростом информации**.
4. Данные разбиваются по значениям признака.
5. Процесс повторяется рекурсивно, пока:
   - все объекты в подмножестве принадлежат одному классу, или
   - не остаётся признаков для разбиения.

3.3. Особенности

- Использует **энтропию** как критерий выбора признака.
- Работает только с **категориальными признаками**.
- Может переобучаться при большом количестве признаков.

---

4. Алгоритм C4.5

4.1. Модификация ID3

Алгоритм **C4.5** (также предложен Куинланом, 1993) — развитие ID3, устраняющее его ограничения.
C4.5 может работать с **непрерывными признаками**, **пропусками** и проводить **обрезку дерева (pruning)**.

4.2. Нововведения C4.5

1. **Обработка числовых признаков:**
   Алгоритм ищет порог ttt, при котором признак AAA делит множество на A≤tA \leq tA≤t и A>tA > tA>t.
2. **Использование Gain Ratio (нормализованного прироста информации):**

   GainRatio(A)=IG(S,A)SplitInfo(A)GainRatio(A) = \frac{IG(S, A)}{SplitInfo(A)}GainRatio(A)=SplitInfo(A)IG(S,A)

   где

   SplitInfo(A)=−∑v∈Values(A)|Sv||S|log⁡2|Sv||S|SplitInfo(A) = - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} \log_2 \frac{|S_v|}{|S|}SplitInfo(A)=−v∈Values(A)∑|S||Sv|log2|S||Sv|

3. **Обрезка дерева (Pruning):**
   После построения дерево упрощается, удаляя узлы, не улучшающие точность на проверочной выборке.

4.3. Преимущества и недостатки

Преимущества:

- Работает с непрерывными и категориальными данными.

- Менее подвержен переобучению.
- Высокая интерпретируемость.

Недостатки:

- Чувствительность к шуму.
- Возможны большие и сложные деревья.

---

5. Алгоритм CART (Classification and Regression Trees)

5.1. Общая характеристика

CART — универсальный алгоритм, разработанный **Брейманом и др. (1984)**. Он может строить деревья как для **классификации**, так и для **регрессии**.

5.2. Принцип работы

1. Использует **бинарные разбиения** — каждый узел делится только на два поддерева.
2. Для классификации — критерий **индекса Джини**,
   для регрессии — **минимизация дисперсии** выходного значения.
3. После построения проводится **обрезка дерева** для предотвращения переобучения.

5.3. Пример

Для задачи классификации клиентов по платежеспособности CART может использовать признаки:

- доход,
- возраст,
- наличие кредита,
  и сформировать бинарное дерево с условиями «доход > 200 000», «возраст < 30» и т.д.

5.4. Преимущества CART

- Универсальность (работает и с классификацией, и с регрессией).
- Простая интерпретация.

- Хорошая производительность.

Недостатки:

- Склонность к переобучению без обрезки.
- Малые изменения в данных могут сильно изменить структуру дерева.

---

6. Случайный лес (Random Forest)

6.1. Основная идея

Случайный лес — это **ансамбль деревьев решений**, объединённых в одну модель.
Он был предложен **Лео Брейманом (2001)** и значительно повысил устойчивость деревьев.

6.2. Принцип работы

1. Из обучающей выборки случайным образом выбираются подмножества (bootstrapping).
2. Для каждого подмножества строится дерево решений.
3. На каждом узле выбирается случайное подмножество признаков.
4. Итоговое решение принимается **голосованием (classification)** или **усреднением (regression)** результатов всех деревьев.

6.3. Преимущества случайного леса

- Высокая точность.
- Устойчивость к переобучению.
- Хорошо работает на больших и разнородных данных.
- Можно оценивать важность признаков (feature importance).

Недостатки:

- Потеря интерпретируемости по сравнению с одним деревом.
- Большие вычислительные затраты при обучении множества деревьев.

---

7. Градиентный бустинг (Gradient Boosted Trees)

7.1. Основная идея

Градиентный бустинг — это другой ансамблевый подход, в котором деревья обучаются **последовательно**, а не параллельно.
Каждое новое дерево исправляет ошибки предыдущих.

7.2. Математическая интуиция

1. Строится первое дерево, предсказывающее исход.
2. Вычисляются **остатки (ошибки)**.
3. Следующее дерево обучается на этих остатках.
4. Итоговое предсказание — сумма всех деревьев с весами.

Fm(x)=Fm−1(x)+ηhm(x)F_m(x) = F_{m-1}(x) + \eta h_m(x)Fm(x)=Fm−1(x)+ηhm(x)

где
η\etaη — скорость обучения (learning rate),
hm(x)h_m(x)hm(x) — новое дерево, обученное на ошибках.

7.3. Популярные реализации

- **XGBoost** — ускоренная реализация с регуляризацией.
- **LightGBM** — оптимизирован для больших данных.
- **CatBoost** — учитывает категориальные признаки и градиентный сдвиг.

7.4. Преимущества и недостатки

Преимущества:

- Высокая точность на сложных задачах.
- Возможность тонкой настройки параметров.
- Хорошая работа с разнородными признаками.

Недостатки:

- Сложность настройки гиперпараметров.
- Более медленное обучение, чем у Random Forest.

---

8. Сравнение методов деревьев решений

Метод	Тип	Критерий	Особенности	Преимущества
ID3	Классификация	Энтропия	Категориальные признаки	Простота
C4.5	Классификация	Gain Ratio	Непрерывные признаки, обрезка	Гибкость
CART	Классиф./Регрессия	Джини / Дисперсия	Бинарные разбиения	Универсальность
Random Forest	Ансамбль	Джини	Независимые деревья	Точность, устойчивость
Gradient Boosting	Ансамбль	Градиент ошибки	Последовательное обучение	Лучшая точность

---

9. Заключение

Методы деревьев решений являются фундаментом многих алгоритмов машинного обучения.
Они просты в интерпретации, эффективны и служат базой для ансамблевых моделей, таких как Random Forest и Gradient Boosting, которые сегодня считаются **одними из лучших алгоритмов классификации и регрессии**.

Эти методы находят применение в самых разных областях:

- финансовый скоринг и кредитные риски,
- медицина,
- маркетинг,
- прогнозирование спроса и цен.

Вместе они представляют собой мощный инструментарий анализа данных, сочетающий простоту и высокую предсказательную силу.

Список литературы

1. Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J. *Classification and Regression Trees.* — Wadsworth, 1984.
2. Quinlan, J. R. *C4.5: Programs for Machine Learning.* — Morgan Kaufmann, 1993.
3. Breiman, L. *Random Forests.* — Machine Learning, 2001.
4. Friedman, J. H. *Greedy Function Approximation: A Gradient Boosting Machine.* — Annals of Statistics, 2001.

5. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*. — O’Reilly, 2022.