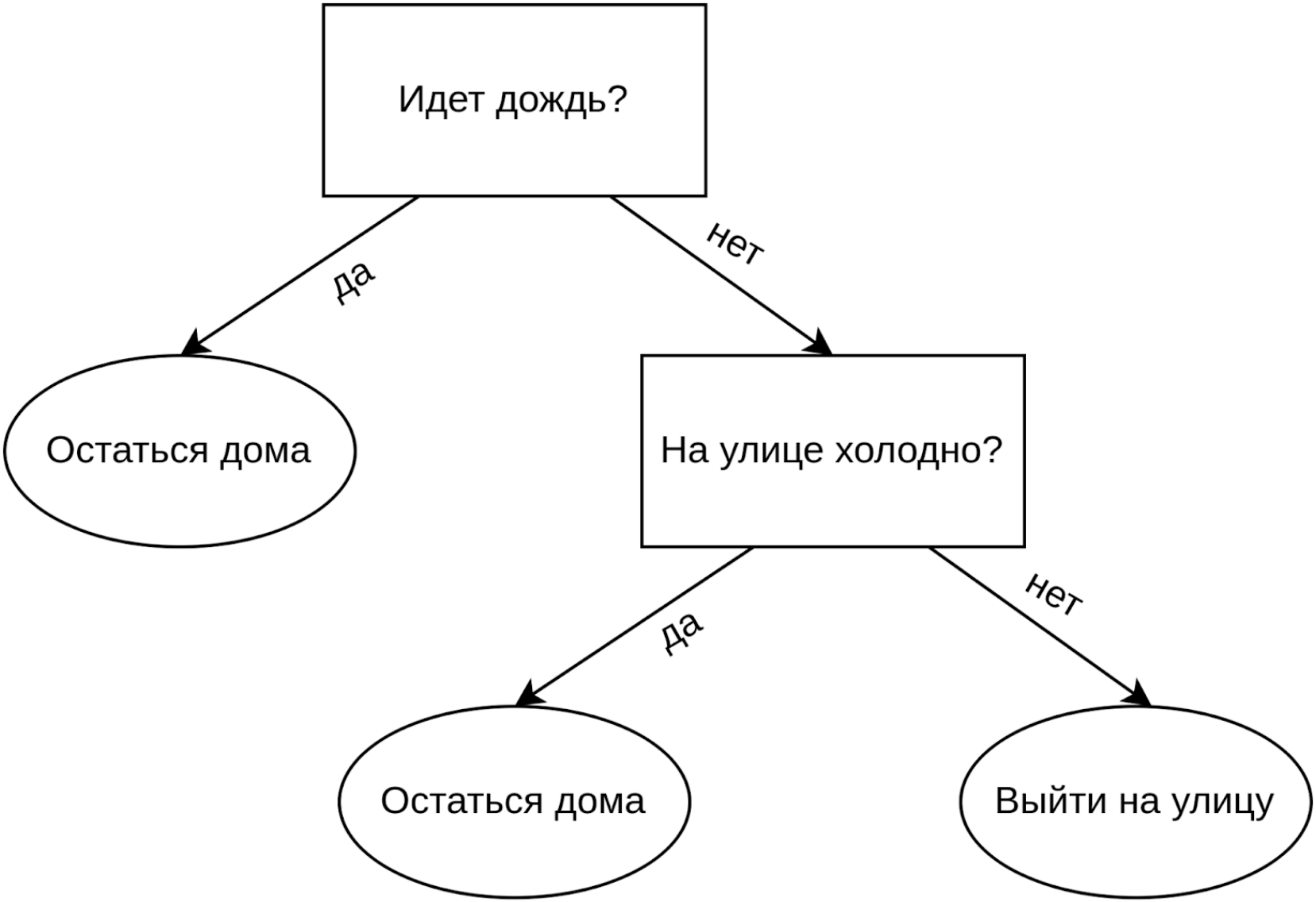
Лекция 6. Деревья решений

Основы интеллектуального анализа данных

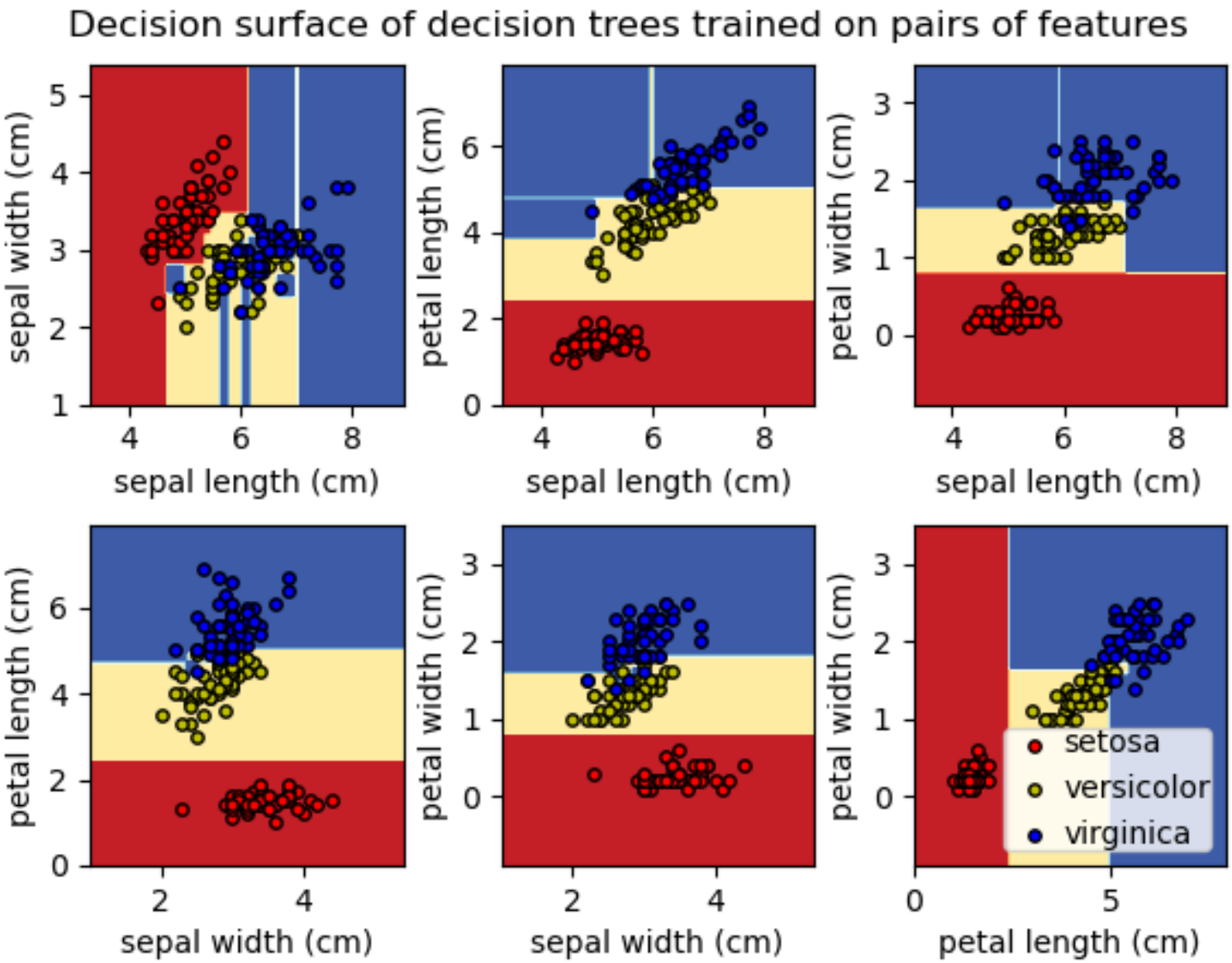
Полузёров Т. Д.

БГУ ФПМИ

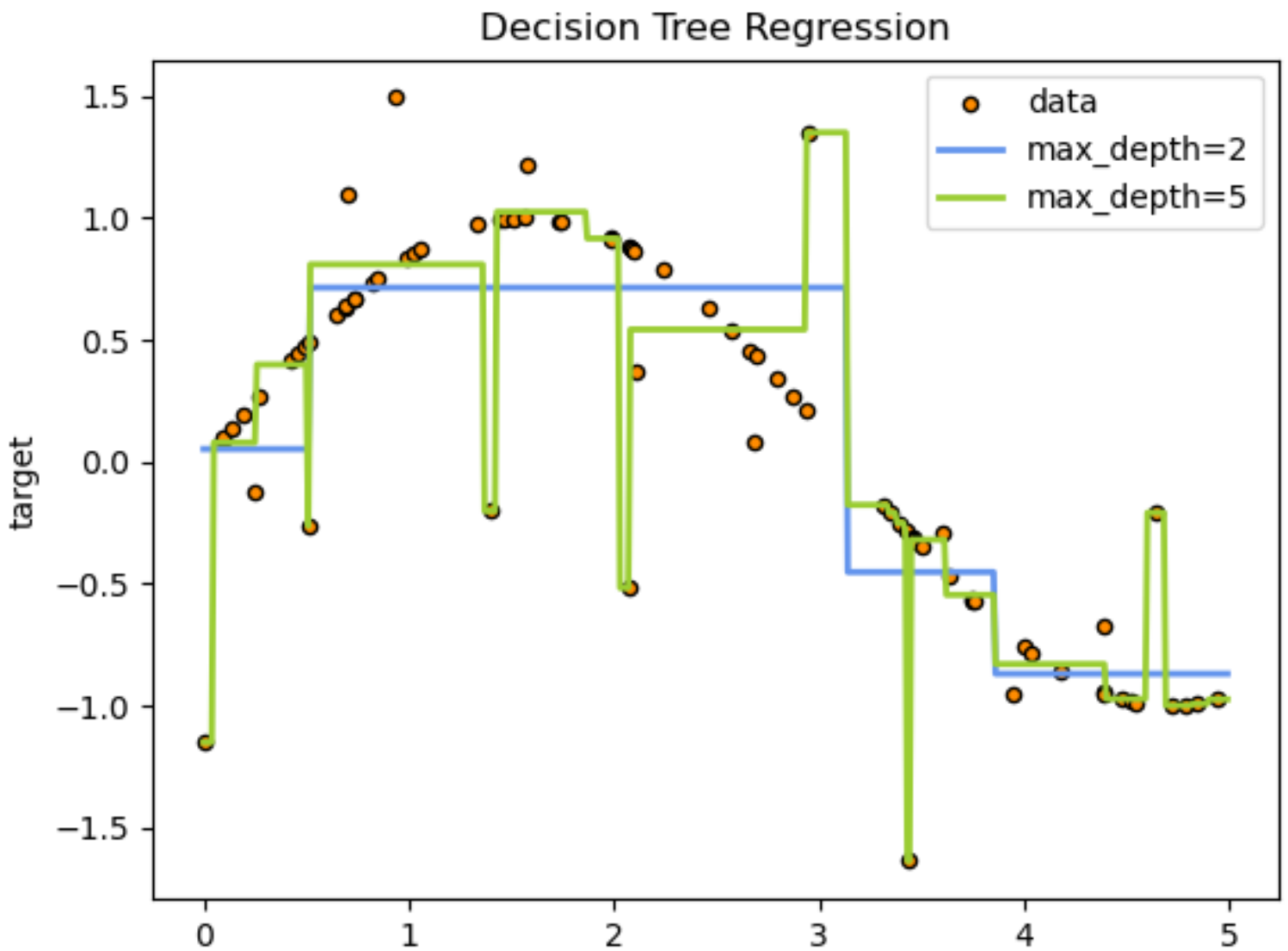
Пример дерева решений



Пример классификации



Пример регрессии



Компоненты бинарного дерева

Дерево состоит из 2-х типов вершин:

- **Внутренняя** вершина v хранит в себе предикат $b_v : \mathbb{X} \to \{0, 1\}$
- **Листовая вершина** v хранит выходное значение $c_v \in \mathbb{Y}$

Любая внутренняя вершина имеет ровно двух потомков.
Любая листовая вершина не имеет потомков.

Глубина дерева h — длина наибольшего маршрута от корня до листа.

Алгоритм работы дерева

Алгоритм $a(x)$ работает по схеме:

1. Стартуем из корня
2. Вычисляем предикат в текущей вершине
3. Если $b_v = 1$ - шагаем в право, $b_v = 0$ - в лево
4. Пока не дошли до листовой вершини, повторяем с шага 2
5. Возвращаем значение в листе c_v

Голосование конъюнкций

Объект x доходит до листовой вершины v тогда и только тогда, когда выполняется конъюнкция $K_v(x)$ всех предикатов на пути от корня до v.

T — множество листовых вершин.
$A_v = \{x \in X | K_v(x) = 1), v \in T\}$ — множества выделяемые конъюнкциями. Причем эти множества не пересекаются, а в объединении совпадают с X.

Решающее дерево есть **кусочно постоянная функция**

$$a(x) = \sum_{v \in T} c_v K_v(x)$$

причем при любом x только одно слагаемое равно единице.

Предикаты

Предикат - любая решащая функция $b : \mathbb{X} \to \{0, 1\}$

- Пороговая функция $b(x) = [x_i > t]$
- Линейный $b(x) = [\langle x, \omega \rangle > t]$
- Метрический $b(x) = [\rho(x, x_v) > t]$, где x_v - некоторый объект выборки

Выбор сложных предикатов - излишен.
Поэтому ограничимся $b(x) = [x_i > t]$

Свойства дерева

После определения дерева видно несколько свойств:

- Дерево есть набор из $|T|$ конъюнкций, непересекающихся и покрывающих все пространство X.
- Если длины конъюнкций небольшие — дерево легко интерпретируется
- Алгоритм $a(x)$ — кусочно постоянная функция. Градиентные методы не применимы.
- Дерево способно полностью выучить обучающую выборку. В этом случае обобщающая способность низкая.

Решающий пень

Пусть имеется размеченная выборка
$X = (x_1, \dots x_\ell), x_i \in \mathbb{R}^n, Y = (y_1, ..., y_\ell)$.

Необходимо минимизировать функцию качества $L(X)$.
Используя предикат $b(x|j,t) = [x^j > t]$ хотим разбить
$X = X_L \cup X_R$, чтобы

$$L(X) > L(X_L) + L(X_R)$$

И даже

$$j^*, t^* = \arg\min_{j,t} L(X_L) + L(X_R)$$

Сложность построения

Для вычисления предиката $b(x)$ по выборке необходимо $O(\ell)$.
А для перебора всевозможных пар (j, t) потребуется $O(\ell n)$.
И итоговая сложность построения пня есть $O(\ell^2 n)$.

Если строить дерево произвольной глубины на котором достигается минимум на тестовой выборке — задача становится NP-полной. Поэтому разрешим себе строить не оптимальные, но хорошие деревья.

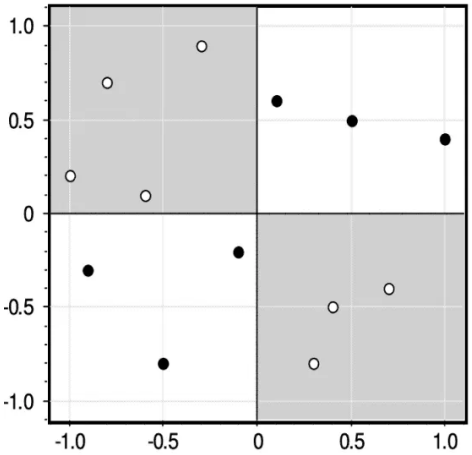
Жадный алгоритм

Очевидной идеей есть строить дерево рекурсивно, как для случая решающего пня.

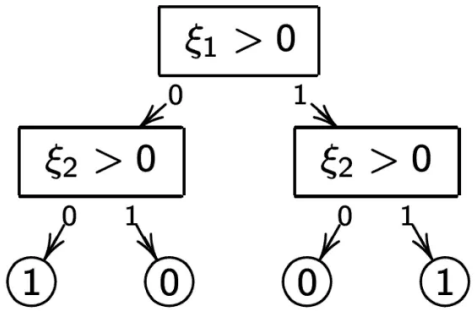
Алгоритм :
На входе имеется выборка X_m

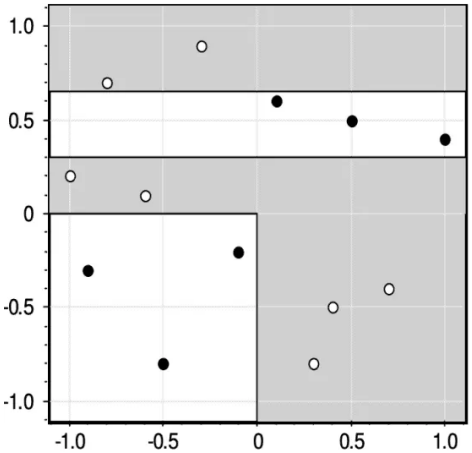
1. Создать вершину v
2. Если выполнен критерий остановки $Stop(X_m)$ — делаем v листом и приписываем ответ $Ans(X_m)$
3. Иначе, перебираем предикаты и находим (i^*, j^*) который производит разбиение $X = X_L \cup X_R$ максимизируя критерий ветвления $Gain(X_m, i, j)$
4. Вызвать рекурсивно для X_L и X_R эту процедуру

Неоптимальность жадного алгоритма

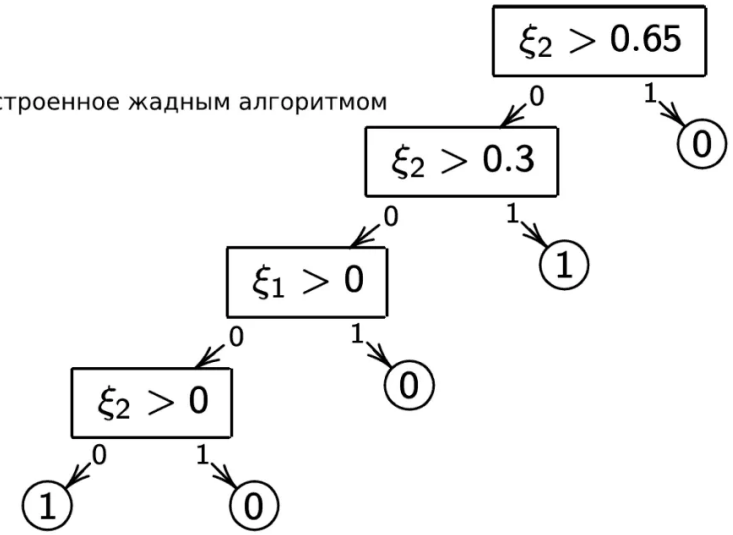


Оптимальное дерево





Дерево, построенное жадным алгоритмом



Листовые вершини

Пусть выполен критерий остановки и необходимо выбрать ответ для данного листа $Ans(X_m)$.

Можно выбирать из следующих соображений:

- Классификации — метка наиболее частого класса, или можно оценить вероятности классов по частотам.
- Регрессия — среднее, медиана или другая статистика.
- Простая модель, например для объктов дошедших до текущего листа выдавать ответ модели, обученной на соответсвующей подвыборке.

Критерий остановки

В качестве критерия остановки можно взять:

- Объекты в вершине принадлежат одному классу или достаточно однородны
- Объектов в вершине слишком мало
- Неудается подобрать удачное разбиение $X = X_L \cup X_R$
- Ограничение на глубину дерева

Критерий ветвления

Как определить оптимальное разбиение здесь и сейчас? Имеем на вход подвыборку X_m
Пусть задача функция ошибки $L(y, c)$

Если текущую вершину сделать листовой и подобрать c, то достиигнем минимальной средней ошибки

$$H(X_m) = \min_{c \in Y} \frac{1}{|X_m|} \sum_{i=1}^{m} L(y_i, c)$$

Величину H называют **неопределенностью** (impurity).

После разбиения

Быть может сможем разбить $X_m = X_L \cup X_R$ используя предикат b и решить в каждой новой вершине задачу отдельно.

Средняя ошибка после разбиения

$$\frac{1}{|X_m|}\left(\sum_{x \in X_L} L(y_i, c_L) + \sum_{x \in X_R} L(y_i, c_R)\right) =$$

$$= \frac{|X_L|}{|X_m|} H(X_L) + \frac{|X_R|}{|X_m|} H(X_R)$$

Прирост информативности

Рассмотрим снижение неопределенности после разбиения

$$Gain(b, X_m) = H(X_m) - \frac{|X_L|}{|X_m|} H(X_L) - \frac{|X_R|}{|X_m|} H(X_R)$$

Если значение велико, значит нам выгодно произвести такое разбиение. Кроме того, чем больше *Gain* — тем лучше.

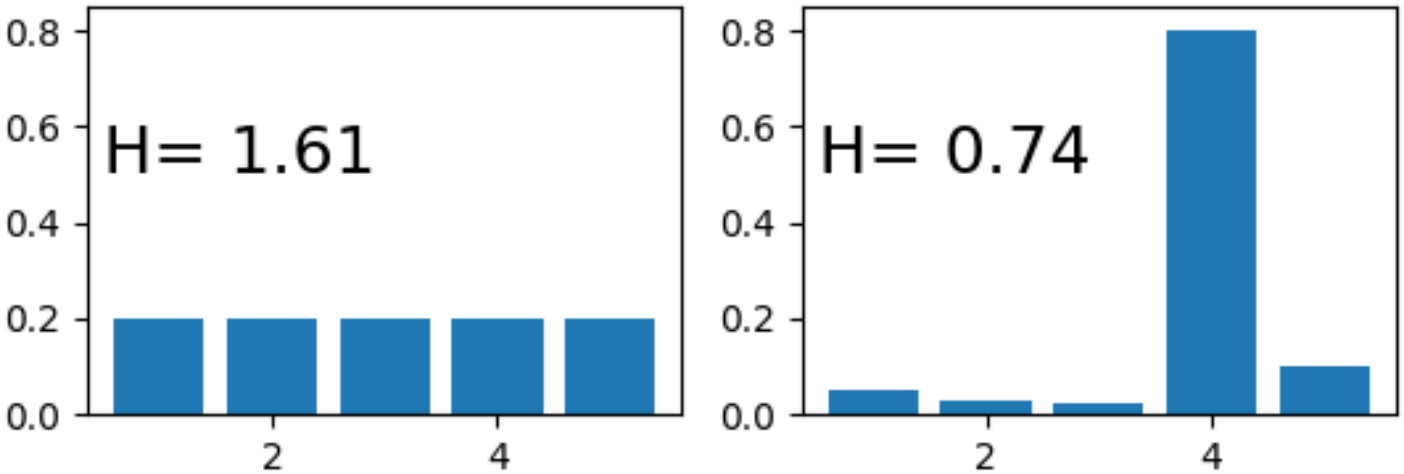
Предикат b при котором достигается наибольшее значение *Gain* и выбирается для текущей вершины.

Энтропия

Энтропия — мера неопределенности распеделения

$$H(X_m) = -\sum_{k=1}^{K} p_k \ln p_k$$

- Энтропия равна нулю у тривиальной случайной величины (принимает одно значение с вероятностью 1)
- Наибольшее значение энтропии у равномерных распределений



Критерий Джини

Также используется критерий Джини.

$$H(X_m) = \sum_{k=1}^{K} p_k(1 - p_k)$$

$H(X_m)$ равно математическому ожиданию числа неправильно классифицированных объектов в случае, если мы будем приписывать им случайные метки из дискретного распределения, заданного вероятностями $(p_1, \ldots, p_k)$

Дисперсия в задаче регрессии

Допустим, решаем задачу регрессии с квадратичной мерой ошибки $L(y, c) = (y - c)^2$

В листовой вершине имеем выборку X_m и необходимо выбрать константу c — ответ алгоритма.
Минимум потерь L на выборке X_m достигается при

$$c = \frac{1}{m} \sum_{i=1}^{m} y_i = \bar{y}$$

Неопределенность в этом случае (по определению)

$$H(X_m) = \frac{1}{m} \sum_{i=1}^{m} (y_i - \bar{y})^2$$

есть **дисперсия** y в выборке X_m.

Пустые значения

Допустим у части входящей выборки X_m значение признака x^i — пропущено. Обозначим множетсво таких объектов V_m.

На этапе построения выбора сплита по этому признаку — не будем учитывать такие объекты, используем $X_m \setminus V_m$.

Этап применения

На этапе применения отправим объект в оба поддерева и потом учтем результаты пропрорционально вероятности попадания в соответсвующее поддерево.

В задаче регрессии итоговый взвешенный ответ

$$\hat{y} = \frac{|X_L|}{|X_m|}\hat{y}_L + \frac{|X_R|}{|X_m|}\hat{y}_R$$

Регуляризация деревьев

Для улучшения обобщающей способности и снижения переобучения применяются методы реуляризации деревьев.

- **pre-pruning** — они же критерии ранней остановки,
- **pruning** — строим дерево без ограничений, а затем запускаем процедуру удаления малоэффективных поддеревьев и вершин.

Заключение

- Дерево решений — кусочно постоянная функция
- Сильно склонно к переобучению
- При небольшой глубине отлично интерпретируется
- Построенные деревья не оптимальные, строятся по набору эвристик
- Умеют работать с категориальными признаками, пропусками