СПбГЭТУ «ЛЭТИ»
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

Александр Калиниченко

ИНТЕЛЛЕКТУАЛЬНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В МЕДИЦИНЕ

Модуль 2. Методы искусственного интеллекта

Лекция 6. Деревья принятия решений

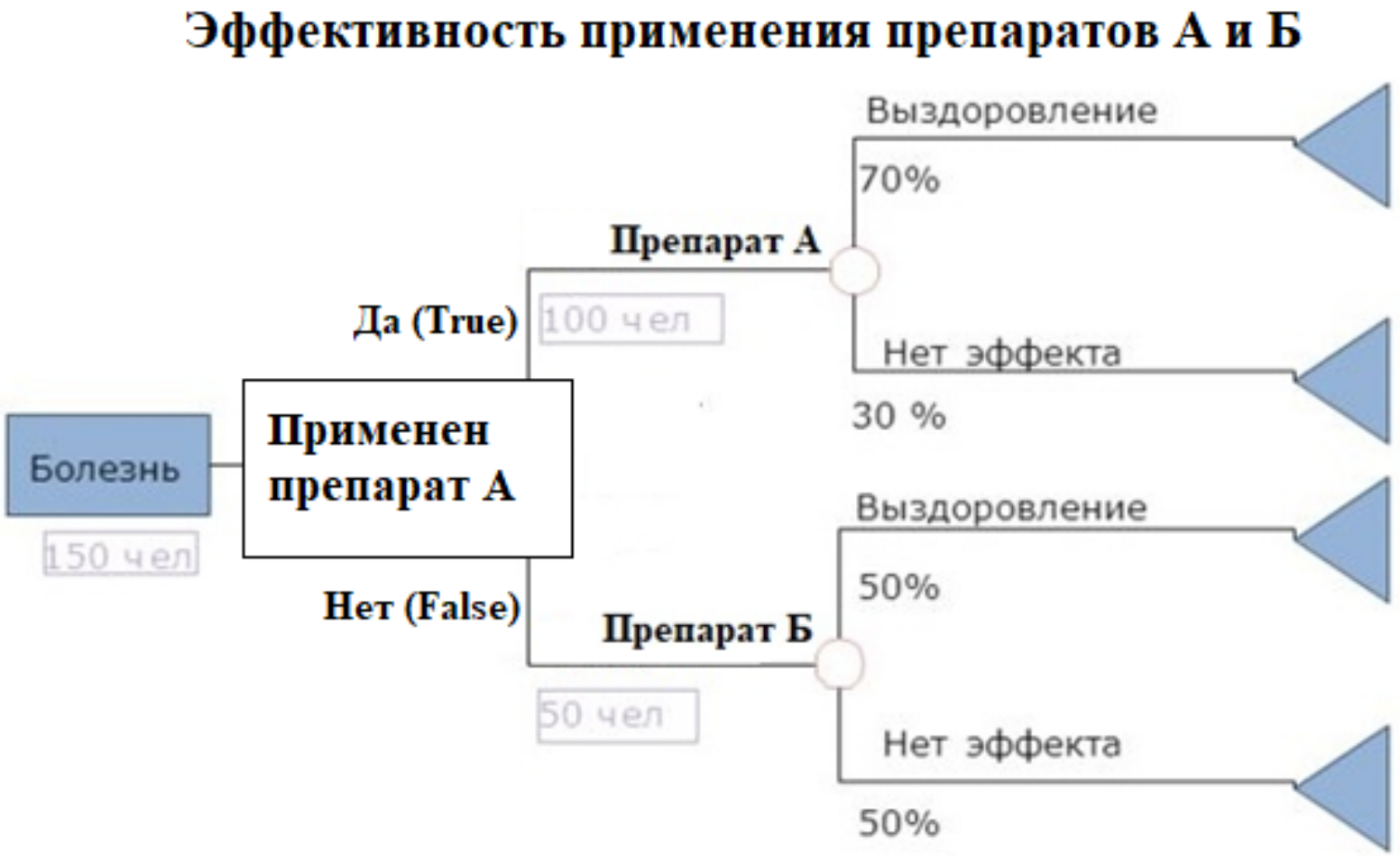
ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

- **Дерево принятия решений** — средство поддержки принятия решений, использующееся в машинном обучении, анализе данных и статистике
- Структура дерева представляет собой «листья» и «ветки». На рёбрах («ветках») дерева решения записаны атрибуты, от которых зависит целевая функция, в «листьях» записаны значения целевой функции

Дерево решений состоит из трёх типов узлов:

- Узлы *решения* — обычно представлены прямоугольниками
- *Вероятностные узлы* — представляются в виде круга
- *Замыкающие* узлы — представляются в виде треугольника

ПРИМЕР ДЕРЕВА ПРИНЯТИЯ РЕШЕНИЙ



Дерево решений не может содержать в себе циклические элементы. Каждый новый лист впоследствии может лишь расщепляться, отсутствуют сходящиеся пути

При конструировании дерева вручную, может возникнуть проблема его размерности, поэтому, как правило, дерево решения мы можем получать с помощью специализированного программного обеспечения

ТИПОЛОГИЯ ДЕРЕВЬЕВ

- **Дерево для классификации**, когда предсказываемый результат является классом, к которому принадлежат данные
- **Дерево для регрессии**, когда предсказываемый результат можно рассматривать как вещественное число

Преимущество деревьев решений в том, что они легко интерпретируемы, понятны человеку

Многие другие, хоть и более точные, модели не обладают этим свойством и могут рассматриваться скорее как "черный ящик", в который загрузили данные и получили ответ

ПОСТРОЕНИЕ ДЕРЕВА РЕШЕНИЙ

Какой признак выбрать первым?

Рассмотрим пример, где все признаки бинарные.

Игра "20 вопросов":

Один человек загадывает знаменитость, а второй пытается отгадать, задавая только вопросы, на которые можно ответить "Да" или "Нет".

Какой вопрос отгадывающий задаст первым делом? Конечно, такой, который сильнее всего уменьшит количество оставшихся вариантов.

К примеру, вопрос "Это Алла Пугачева?" в случае отрицательного ответа оставит огромное количество вариантов для дальнейшего перебора, а вот вопрос "Это женщина?" отсечет уже около половины знаменитостей. То есть, признак "пол" намного лучше разделяет выборку людей, чем признак "это Алла Пугачева ", "национальность-испанец" или "любит футбол".

ЭНТРОПИЯ

Описанный выше подход интуитивно соответствует понятию прироста информации, основанному на **энтропии**

Энтропия Шеннона определяется для системы с N возможными состояниями следующим образом:

$$S = -\sum_{i=1}^{N} p_i \log_2 p_i$$

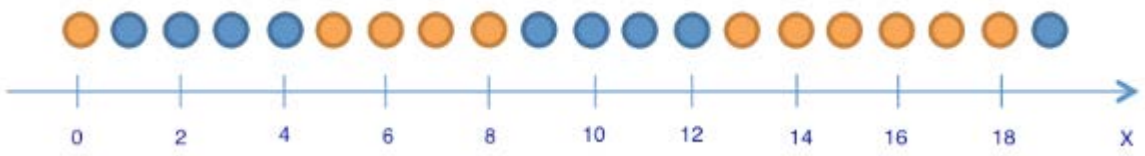
где p_i – вероятности нахождения системы в i-м состоянии

Чем выше энтропия, тем менее упорядочена система и наоборот

Энтропия помогает формализовать эффективное разделение выборки

ПРИМЕР, ИЛЛЮСТРИРУЮЩИЙ ЭНТРОПИЮ (1)

Будем предсказывать цвет шарика по его координате:



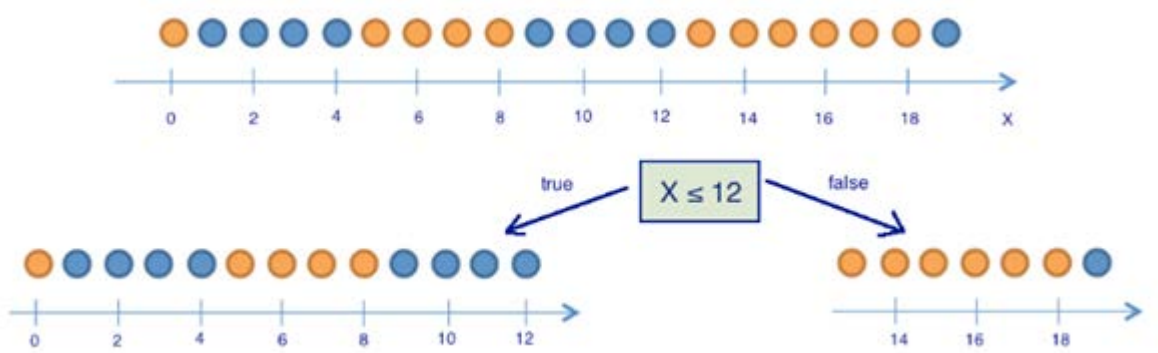
Здесь 9 синих шариков и 11 желтых. Если мы наудачу вытащили шарик, то он с вероятностью $p1=9/20$ будет синим и с вероятностью $p2=11/20$ – желтым

Значит, энтропия начального состояния $S0=-9/20\log2(9/20)-11/20\log2(11/20)\approx1$

Само это значение пока ни о чем нам не говорит

ПРИМЕР, ИЛЛЮСТРИРУЮЩИЙ ЭНТРОПИЮ (2)

Посмотрим, как изменится энтропия, если разбить шарики на две группы – с координатой меньше либо равной 12 и больше 12:



В левой группе оказалось 13 шаров, из которых 8 синих и 5 желтых. Энтропия этой группы равна $S_1 = -5/13 \log_2(5/13) - 8/13 \log_2(8/13) \approx 0.96$

В правой группе оказалось 7 шаров, из которых 1 синий и 6 желтых. Энтропия правой группы равна $S_2 = -1/7 \log_2(1/7) - 6/7 \log_2(6/7) \approx 0.6$

Как видим, энтропия уменьшилась в обеих группах по сравнению с начальным состоянием

ПРИМЕР, ИЛЛЮСТРИРУЮЩИЙ ЭНТРОПИЮ (3)

Поскольку энтропия – по сути степень хаоса (или неопределенности), уменьшение энтропии называют приростом информации. Формально прирост информации (information gain, IG) при разбиении выборки по признаку Q (в нашем примере это признак "x⩽12") определяется как:

$$IG(Q) = S_O - \sum_{i=1}^{q} \frac{N_i}{N} S_i,$$

где q – число групп после разбиения, Ni – число элементов выборки, у которых признак Q имеет i-ое значение. В нашем случае после разделения получилось две группы (q=2) – одна из 13 элементов (N1=13), вторая – из 7 (N2=7). Прирост информации получился

$$IG(x \leq 12) = S_0 - \frac{13}{20} S_1 - \frac{7}{20} S_2 \approx 0.16.$$

Получается, разделив шарики на две группы по признаку "координата меньше либо равна 12", мы уже получили более упорядоченную систему, чем в начале

ПРИМЕР, ИЛЛЮСТРИРУЮЩИЙ ЭНТРОПИЮ (4)

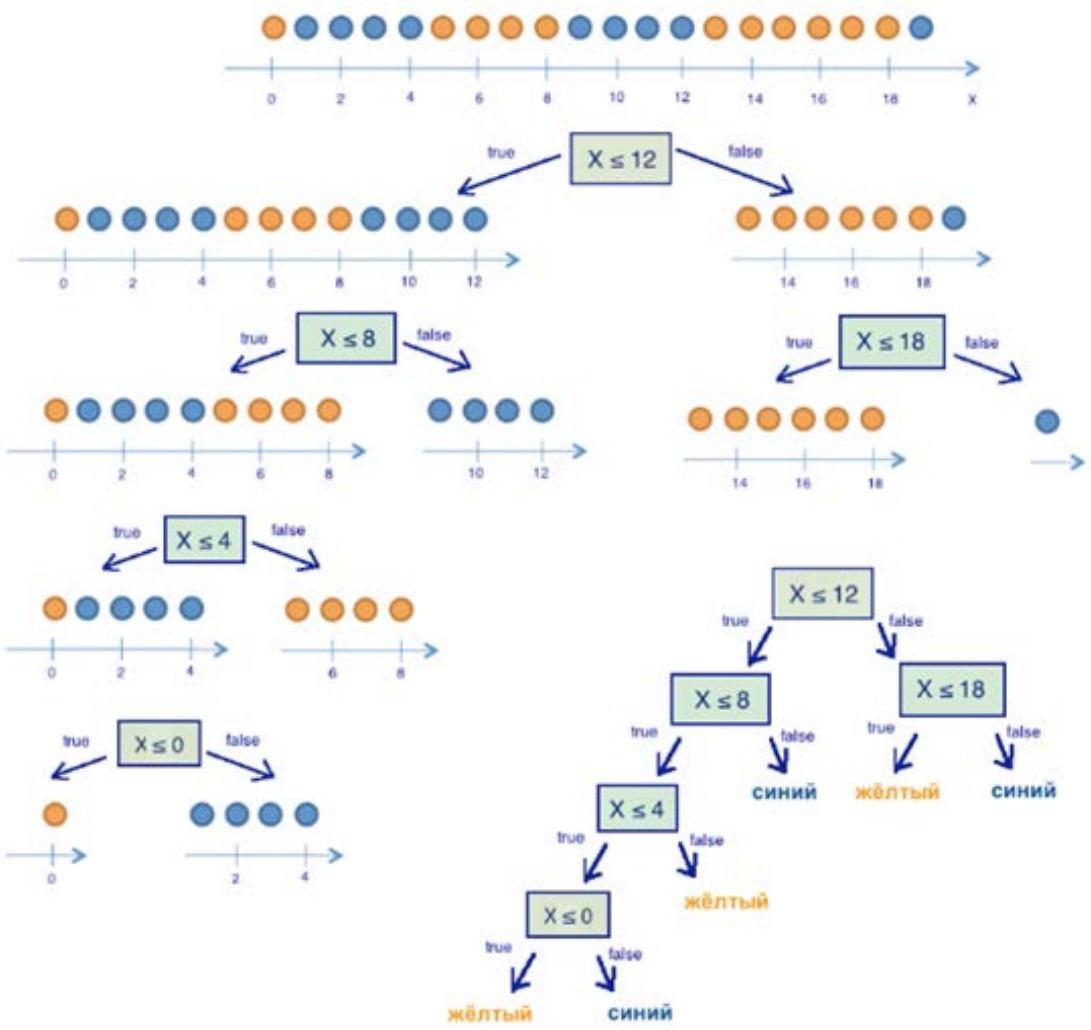
Продолжим деление шариков на группы до тех пор, пока в каждой группе шарики не будут одного цвета

Для правой группы потребовалось всего одно дополнительное разбиение по признаку "координата меньше либо равна 18", для левой – еще три. Очевидно, энтропия группы с шариками одного цвета равна нулю (1×log21=0), что соответствует представлению, что группа шариков одного цвета – упорядоченная

В итоге мы построили дерево решений, предсказывающее цвет шарика по его координате.

Конечно, такое дерево решений может плохо работать для новых объектов (определения цвета новых шариков), поскольку оно идеально подстроилось под обучающую выборку (изначальные 20 шариков)

Эта проблема называется **переобучением**



АЛГОРИТМ ПОСТРОЕНИЯ ДЕРЕВА

Построенное дерево является в некотором смысле оптимальным – потребовалось только 5 "вопросов" (условий на признак x), чтобы "подогнать" дерево решений под обучающую выборку, то есть чтобы дерево правильно классифицировало любой обучающий объект. При других условиях разделения выборки дерево получится глубже

В основе популярных алгоритмов построения дерева решений лежит принцип **жадной максимизации** прироста информации – на каждом шаге выбирается тот признак, при разделении по которому прирост информации оказывается наибольшим
Дальше процедура повторяется рекурсивно, пока энтропия не окажется равной нулю или какой-то малой величине (если дерево не подгоняется идеально под обучающую выборку во избежание переобучения)

ЭВРИСТИКИ «РАННЕЙ ОСТАНОВКИ»

В разных алгоритмах применяются разные эвристики для "ранней остановки" или "отсечения", чтобы избежать построения переобученного дерева
Кроме энтропии предлагаются следующие критерии:

- Неопределенность Джини (Gini impurity):

$$G = 1 - \sum_k (p_k)^2$$

Максимизацию этого критерия можно интерпретировать как максимизацию числа пар объектов одного класса, оказавшихся в одном поддереве

- Ошибка классификации (misclassification error):

$$E = 1 - \max_k p_k$$

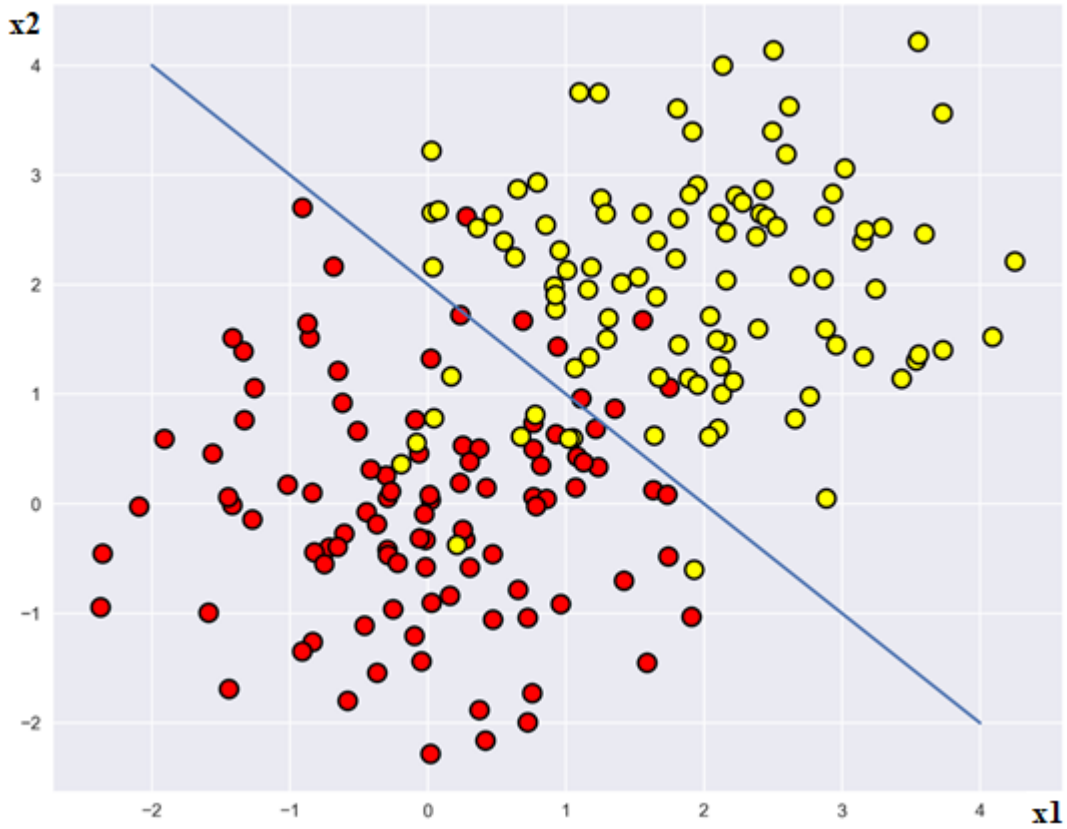
На практике ошибка классификации почти не используется, а неопределенность Джини и прирост информации работают почти одинаково

ПРИМЕР ПОСТРОЕНИЯ ДЕРЕВА РЕШЕНИЙ (1)

Рассмотрим пример применения дерева решений для синтетических данных. Два класса будут сгенерированы из двух нормальных распределений с разными средними

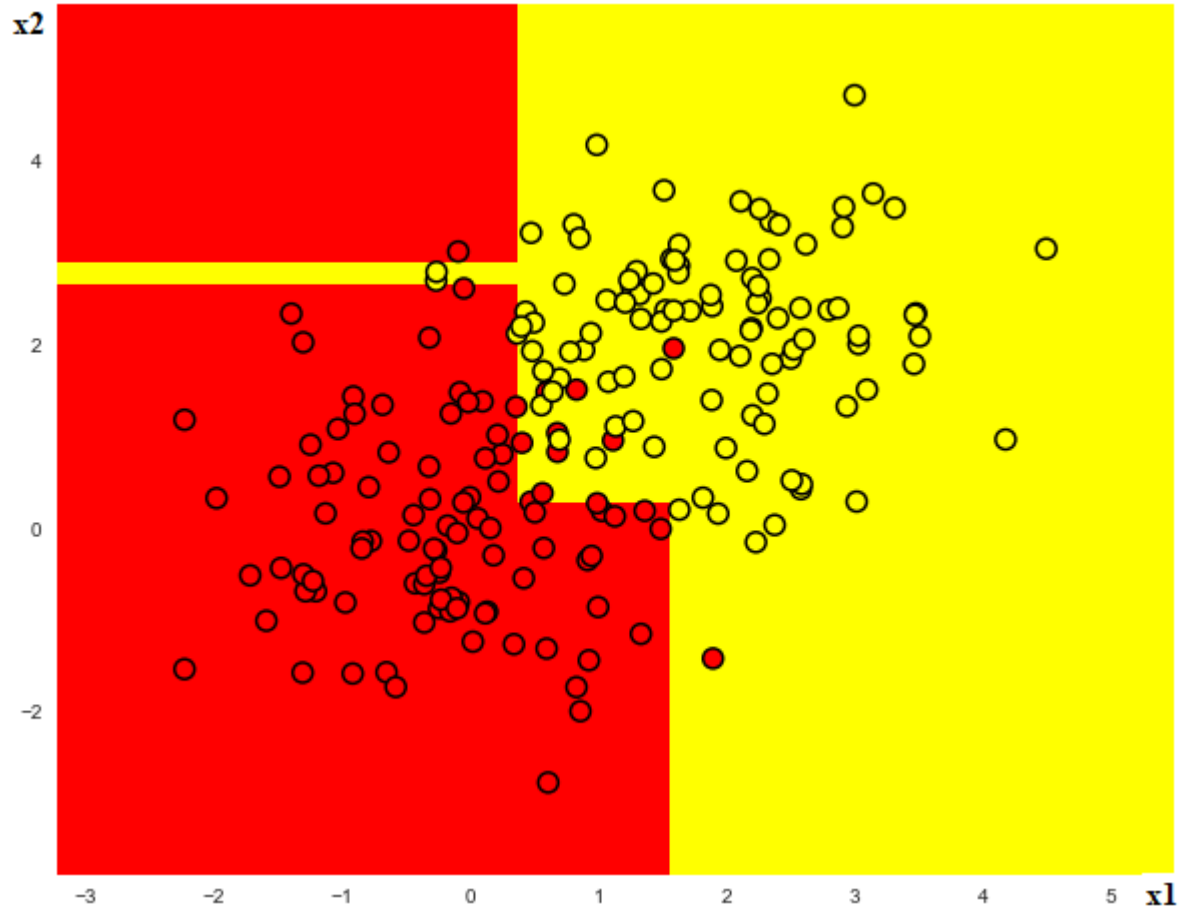
Отобразим данные. Неформально, задача классификации в этом случае – построить какую-то "хорошую" границу, разделяющую 2 класса (красные точки от желтых). Возможно, прямая будет слишком простой границей, а какая-то сложная кривая, огибающая каждую красную точку – будет слишком сложной и будем много ошибаться на новых примерах из того же распределения, из которого пришла обучающая выборка

Интуиция подсказывает, что хорошо на новых данных будет работать какая-то гладкая граница, разделяющая 2 класса, или хотя бы просто прямая (в n-мерном случае – гиперплоскость)



ПРИМЕР ПОСТРОЕНИЯ ДЕРЕВА РЕШЕНИЙ (2)

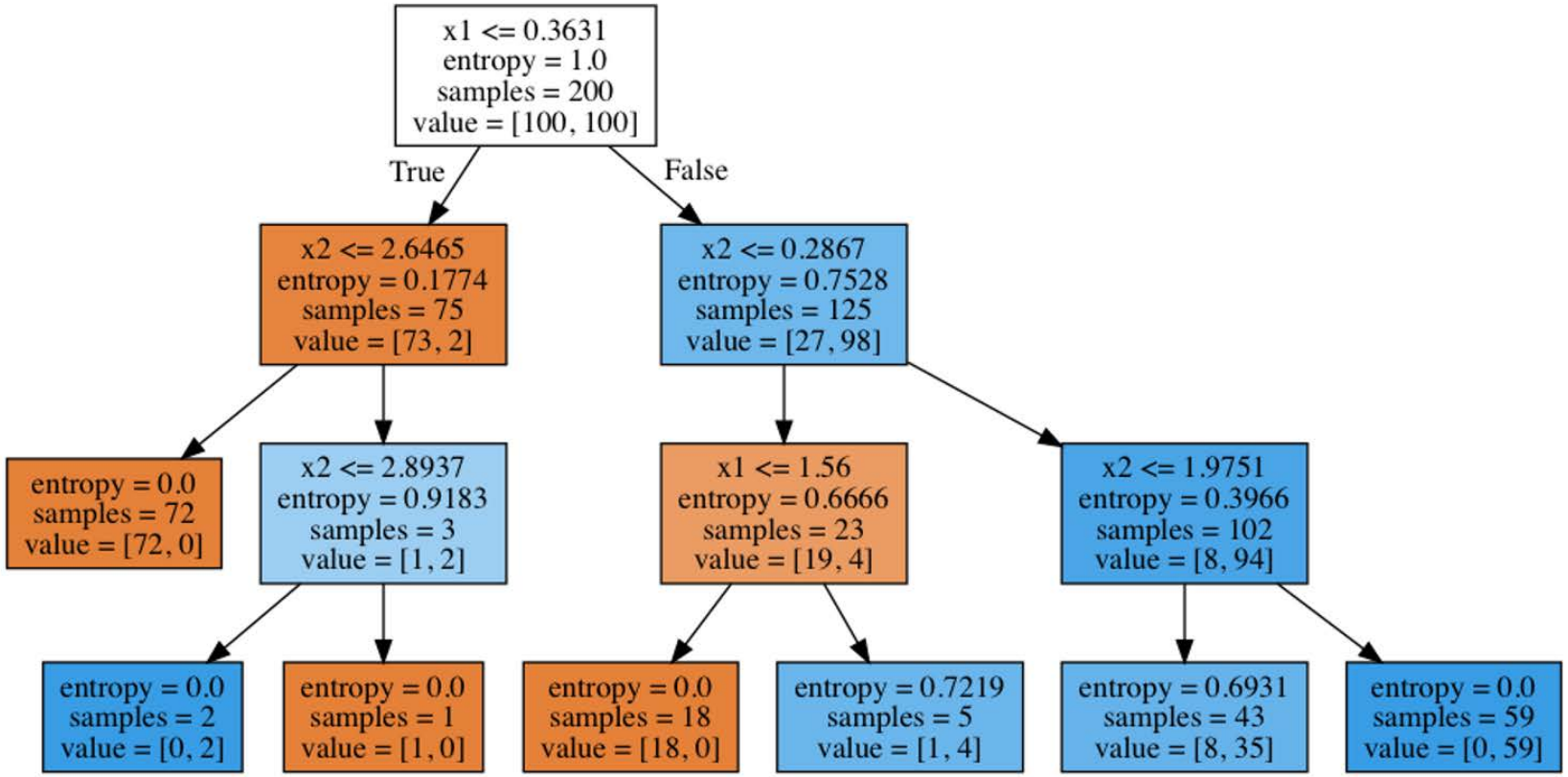
Попробуем разделить эти два класса, обучив дерево решений. В дереве будем использовать ограничение глубины дерева «3». Визуализируем полученную границу разделения классов



ПРИМЕР ПОСТРОЕНИЯ ДЕРЕВА РЕШЕНИЙ (3)

Дерево "нарезает" пространство на 7 прямоугольников (в дереве 7 листьев). В каждом таком прямоугольнике прогноз дерева будет константным, по превалированию объектов того или иного класса

Вначале было 200 объектов, 100 — одного класса и 100 – другого. Энтропия начального состояния была максимальной – 1. Затем было сделано разбиение объектов на 2 группы в зависимости от сравнения признака x1 со значением 0.3631. При этом энтропия и в левой, и в правой группе объектов уменьшилась. И так далее, дерево строится до глубины 3



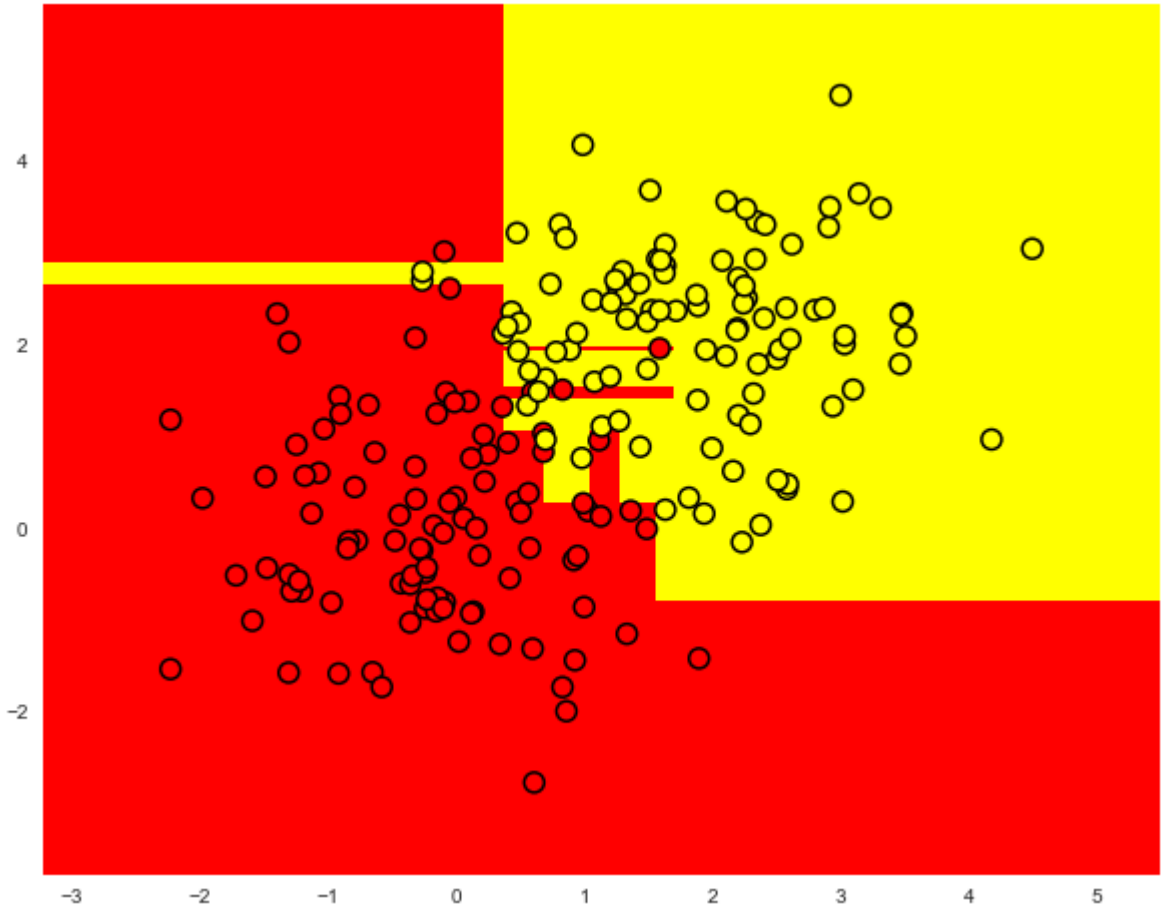
ГЛУБИНА ПОСТРОЕНИЯ ДЕРЕВА

В принципе дерево решений можно построить до такой глубины, чтоб в каждом листе был ровно один объект. Но на практике это не делается (если строится только одно дерево) из-за того, что такое дерево будет переобученным – оно слишком настроится на обучающую выборку и будет плохо работать на прогноз на новых данных. Где-то внизу дерева, на большой глубине, будут появляться разбиения по менее важным признакам

Есть два исключения, ситуации, когда деревья строятся до максимальной глубины:

- *Случайный лес* (композиция многих деревьев) усредняет ответы деревьев, построенных до максимальной глубины
- *Стрижка дерева (pruning)*. При таком подходе дерево сначала строится до максимальной глубины, потом постепенно, снизу вверх, некоторые вершины дерева убираются за счет сравнения по качеству дерева с данным разбиением и без него (сравнение проводится с помощью кросс-валидации)

ПРИМЕР ПЕРЕОБУЧЕНИЯ ДЕРЕВА



Основные способы **борьбы с переобучением** в случае деревьев решений:

- *Искусственное ограничение глубины* или минимального числа объектов в листе: построение дерева просто в какой-то момент прекращается
- *Стрижка* дерева

ДЕРЕВО РЕШЕНИЙ В ЗАДАЧЕ РЕГРЕССИИ (1)

При прогнозировании количественного признака идея построения дерева остается та же, но **меняется критерий качества**

Дисперсия вокруг среднего:

$$D = \frac{1}{\ell}\sum_{i=1}^{\ell}(y_i - \frac{1}{\ell}\sum_{i=1}^{\ell} y_i)^2,$$

где l – число объектов в листе, y_i – значения целевого признака. Фактически, минимизируя дисперсию вокруг среднего, мы ищем признаки, разбивающие выборку таким образом, что значения целевого признака в каждом листе примерно равны

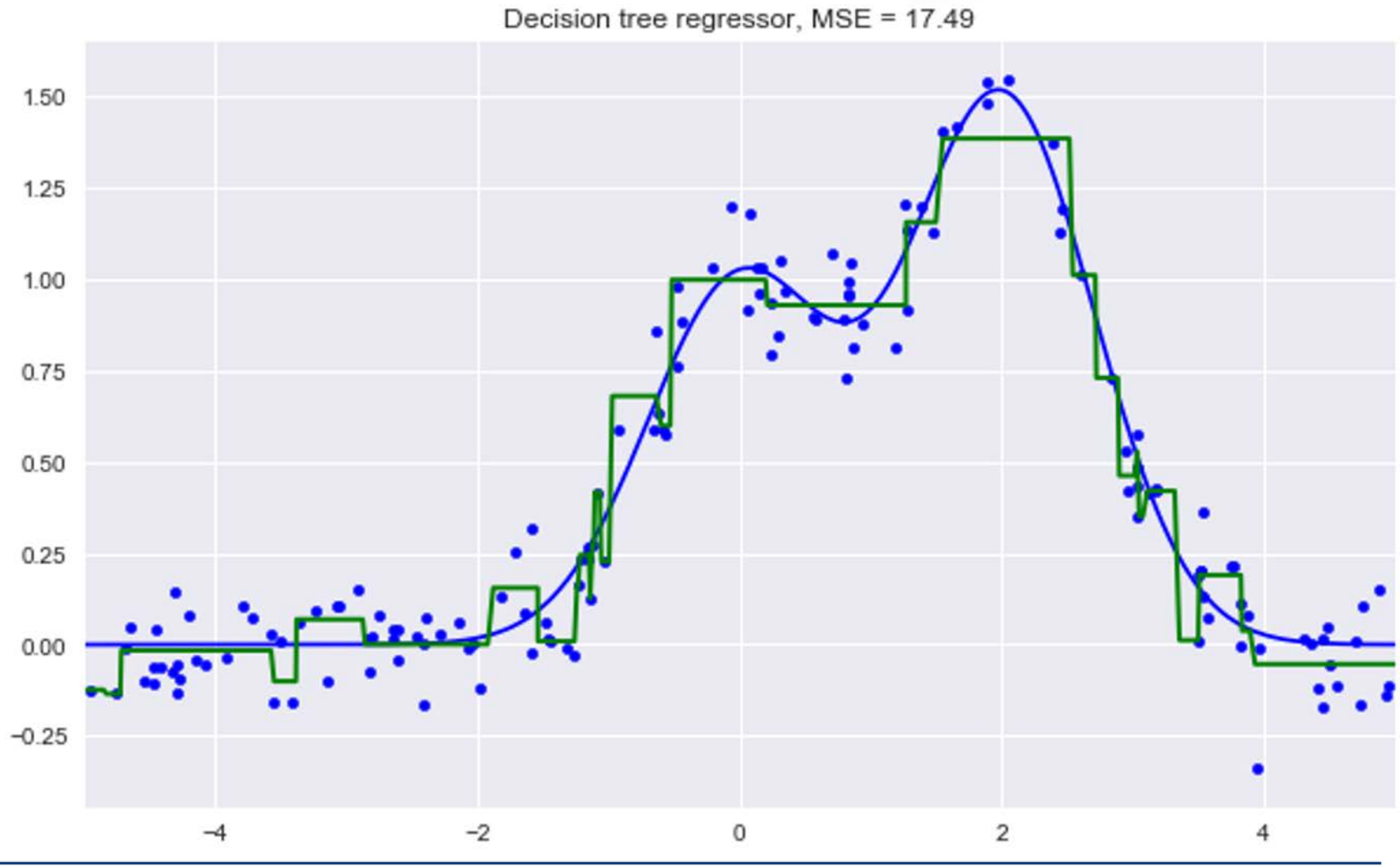
ДЕРЕВО РЕШЕНИЙ В ЗАДАЧЕ РЕГРЕССИИ (1)

Сгенерируем данные, распределенные вокруг функции

$$f(x) = e^{-x^2} + 1.5 * e^{-(x-2)^2}$$

с некоторым шумом, обучим на них дерево решений и изобразим, какие прогнозы делает дерево

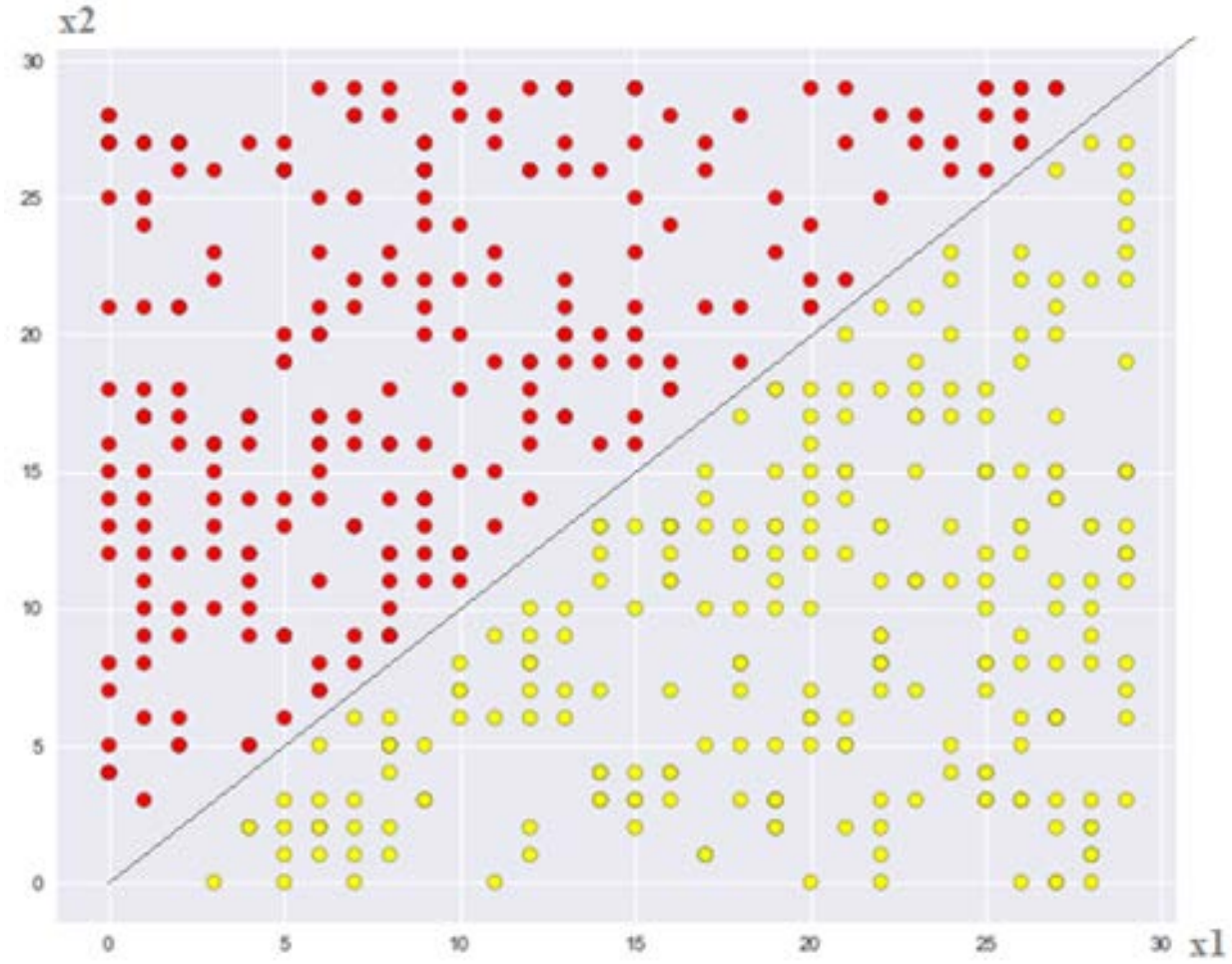
Видим, что дерево решений аппроксимирует зависимость в данных кусочно-постоянной функцией



СЛОЖНЫЙ СЛУЧАЙ ДЛЯ ДЕРЕВЬЕВ РЕШЕНИЙ (1)

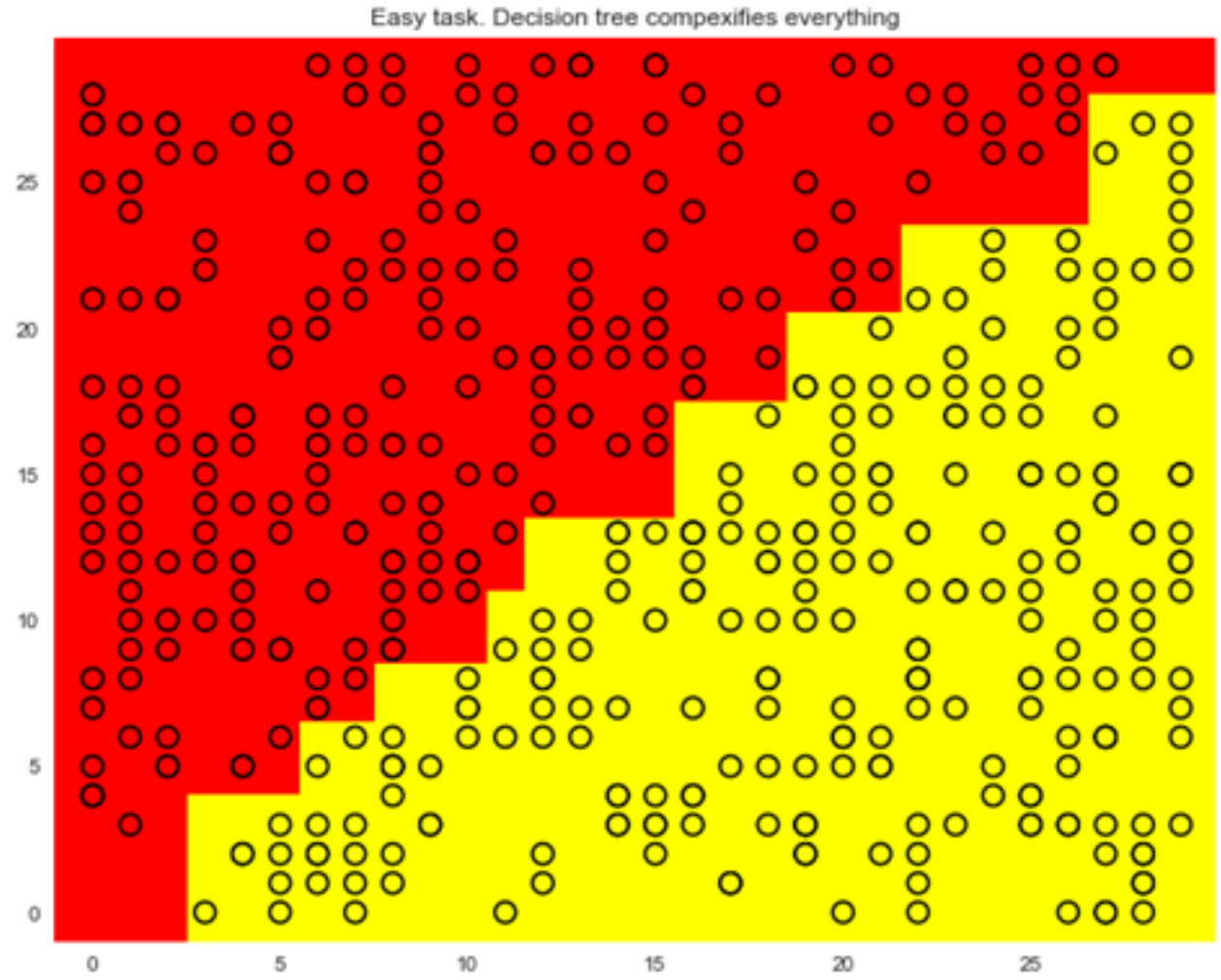
Приведем очень простой пример задачи классификации, с которым дерево справляется, но делает все как-то "сложнее", чем хотелось бы

Создадим множество точек на плоскости (2 признака), каждая точка будет относиться к одному из классов (+1, красные, или -1 – желтые). Если смотреть на это как на задачу классификации, то вроде все очень просто – классы разделяются прямой.



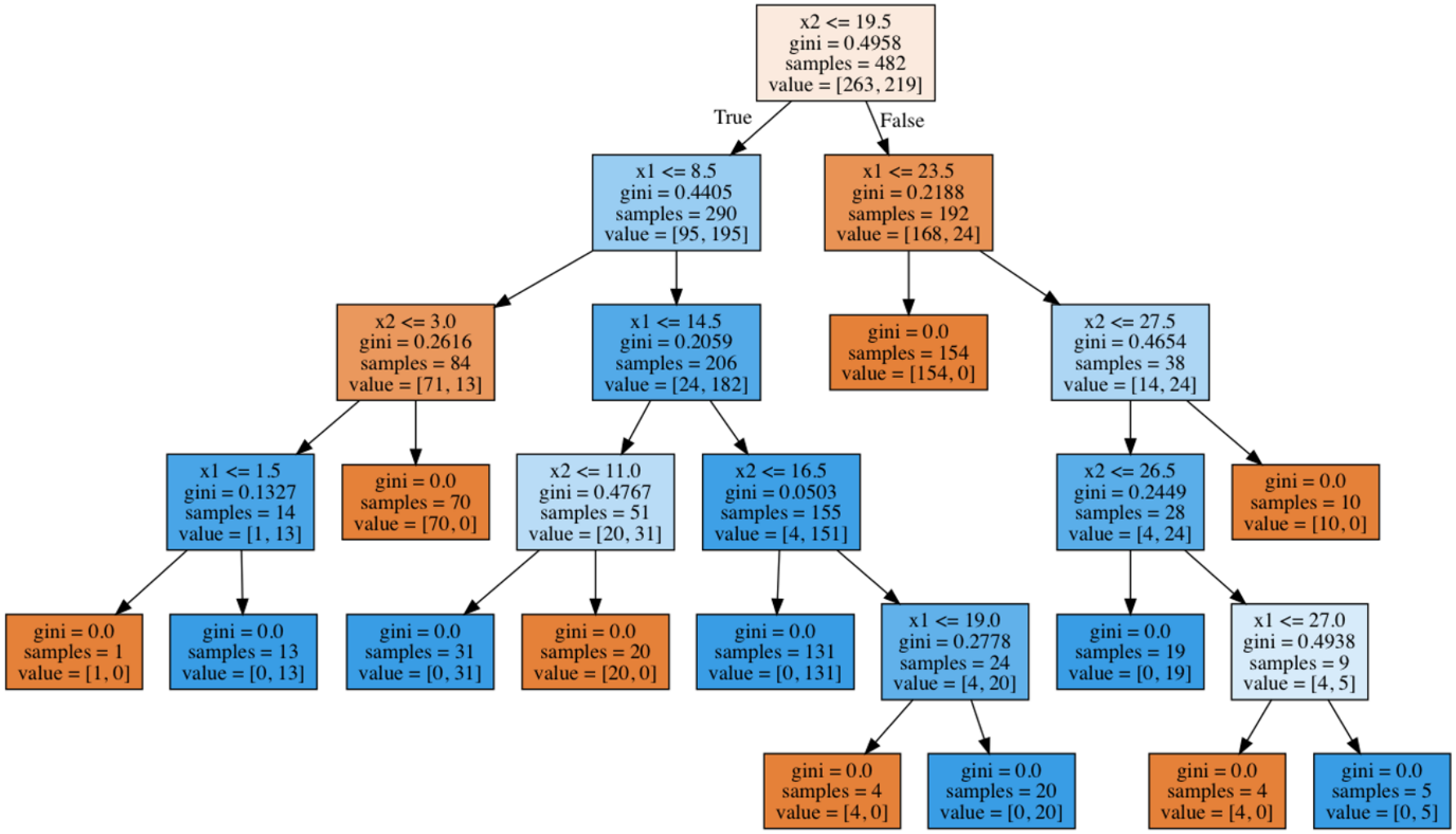
СЛОЖНЫЙ СЛУЧАЙ ДЛЯ ДЕРЕВЬЕВ РЕШЕНИЙ (2)

Однако дерево решений строит слишком сложную границу и само по себе оказывается глубоким. Кроме того, представьте, как плохо дерево будет обобщаться на пространство вне представленного квадрата 30×30, обрамляющего обучающую выборку.



СЛОЖНЫЙ СЛУЧАЙ ДЛЯ ДЕРЕВЬЕВ РЕШЕНИЙ (3)

Вот такая сложная конструкция, хотя правильное решение (хорошая разделяющая поверхность) – это всего лишь прямая x1=x2.



ПЛЮСЫ ДЕРЕВЬЕВ РЕШЕНИЙ

- Порождение четких правил классификации, понятных человеку. Это свойство называют интерпретируемостью модели
- Деревья решений могут легко визуализироваться, то есть может "интерпретироваться" как сама модель (дерево), так и прогноз для отдельного взятого тестового объекта (путь в дереве)
- Быстрые процессы обучения и прогнозирования
- Малое число параметров модели
- Поддержка и числовых и категориальных признаков

МИНУСЫ ДЕРЕВЬЕВ РЕШЕНИЙ

- Деревья очень чувствительны к шумам во входных данных, вся модель может кардинально измениться, если немного изменится обучающая выборка (например, если убрать один из признаков или добавить несколько объектов), поэтому и правила классификации могут сильно изменяться, что ухудшает интерпретируемость модели
- Разделяющая граница, построенная деревом решений, имеет свои ограничения (состоит из гиперплоскостей, перпендикулярных какой-то из координатных осей), и на практике дерево решений по качеству классификации уступает некоторым другим методам
- Необходимость отсекать ветви дерева (pruning) или устанавливать минимальное число элементов в листьях дерева или максимальную глубину дерева для борьбы с переобучением
- Нестабильность. Небольшие изменения в данных могут существенно изменять построенное дерево решений. С этой проблемой борются с помощью ансамблей деревьев решений (рассмотрим далее)
- Проблема поиска оптимального дерева решений сложна, поэтому на практике используются эвристики типа **жадного** поиска признака с максимальным приростом информации, которые не гарантируют нахождения глобально оптимального дерева
- Сложно поддерживаются пропуски в данных
- Модель умеет только интерполировать, но не экстраполировать. То есть дерево решений делает константный прогноз для объектов, находящихся в признаковом пространстве вне параллелепипеда, охватывающего все объекты обучающей выборки