Введение в классические методы: Логистическая регрессия и деревья решений

Калашников Александр

Задача для разогрева:

Задача: В школе №123 был проведен опрос среди 300 старшеклассников о том, будут ли они рекомендовать новое мобильное приложение для общения "ChatHub" своим друзьям. 255 учеников ответили утвердительно.

Вопрос: Какова вероятность того, что случайно выбранный ученик порекомендует это приложение?

Переход к шансам:

Основные понятия:

• **Вероятность** — это мера того, насколько вероятно произойдет событие.
• **Шансы (Odds)** — это отношение вероятности того, что событие произойдет, к вероятности того, что оно не произойдет.

Вероятность рекомендации приложения:

$$p = \frac{255}{300} = 0.85$$

Шансы рекомендации приложения рассчитываются так:

$$\text{Шансы} = \frac{\text{вероятность «да»}}{\text{вероятность «нет»}} = \frac{p}{1 - p}$$

Подставим числа:

$$\text{Шансы} = \frac{0.85}{1 - 0.85} = \frac{0.85}{0.15} = 5.6$$

Нельзяграм:

Контекст: Анализ данных показывает, что среди молодых людей доля женщин, использующих Нельзяграм, составляет 61.08%, в то время как доля мужчин — 43.98%. Разница в долях составляет 17.1% и является статистически значимой.

Рассчитаем шансы для женщин:

$$\text{Шансы} = \frac{p}{1 - p} = \frac{0.6108}{1 - 0.6108} = 1.5694$$

Для мужчин:

$$\text{Шансы} = \frac{0.4398}{1 - 0.4398} = 0.7851$$

Модель лог. регрессии:

Простейшая линейная модель:

$$p = kx + b, \text{ где}$$

$$x = \begin{cases} 1, \text{если женщина} \\ 0, \text{если мужчина} \end{cases}$$

Простейшая модель лог. регрессии:

$$log\left(\frac{p}{1-p}\right) = kx + b$$

Получим:

$$log\left(\frac{p_{\text{жен}}}{1-p_{\text{жен}}}\right) = kx + b$$

$$log\left(\frac{p_{\text{муж}}}{1-p_{\text{муж}}}\right) = kx$$

Модель лог. регрессии:

$$log\left(\frac{p_{\text{жен}}}{1-p_{\text{жен}}}\right) = 0.4507 = kx + b$$

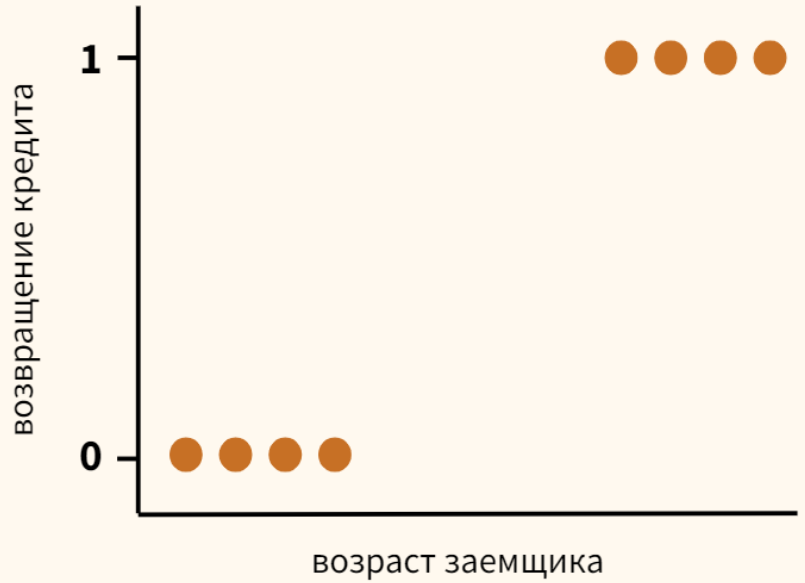
$$log\left(\frac{p_{\text{муж}}}{1-p_{\text{муж}}}\right) = -0.2419 = kx$$

В итоге обученная модель:

$$log(\text{Шансы}) = -0.2419 + 0.6926x$$

Задача бинарной классификации:

Рассмотрим задачу кредитного скоринга:

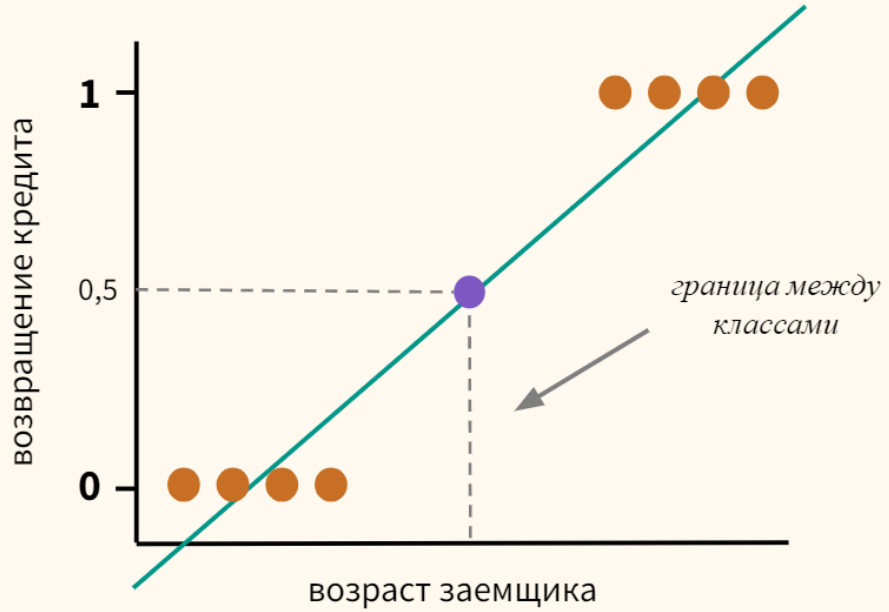


Предположим, что мы собрали данные и выявили зависимость возвращения кредита (ось y) от возраста заемщика (ось x).

Как мы видим, в среднем более молодые заемщики реже возвращают кредит. Возникает вопрос, с помощью какой модели можно описать эту зависимость?

Задача бинарной классификации:

Рассмотрим задачу кредитного скоринга:

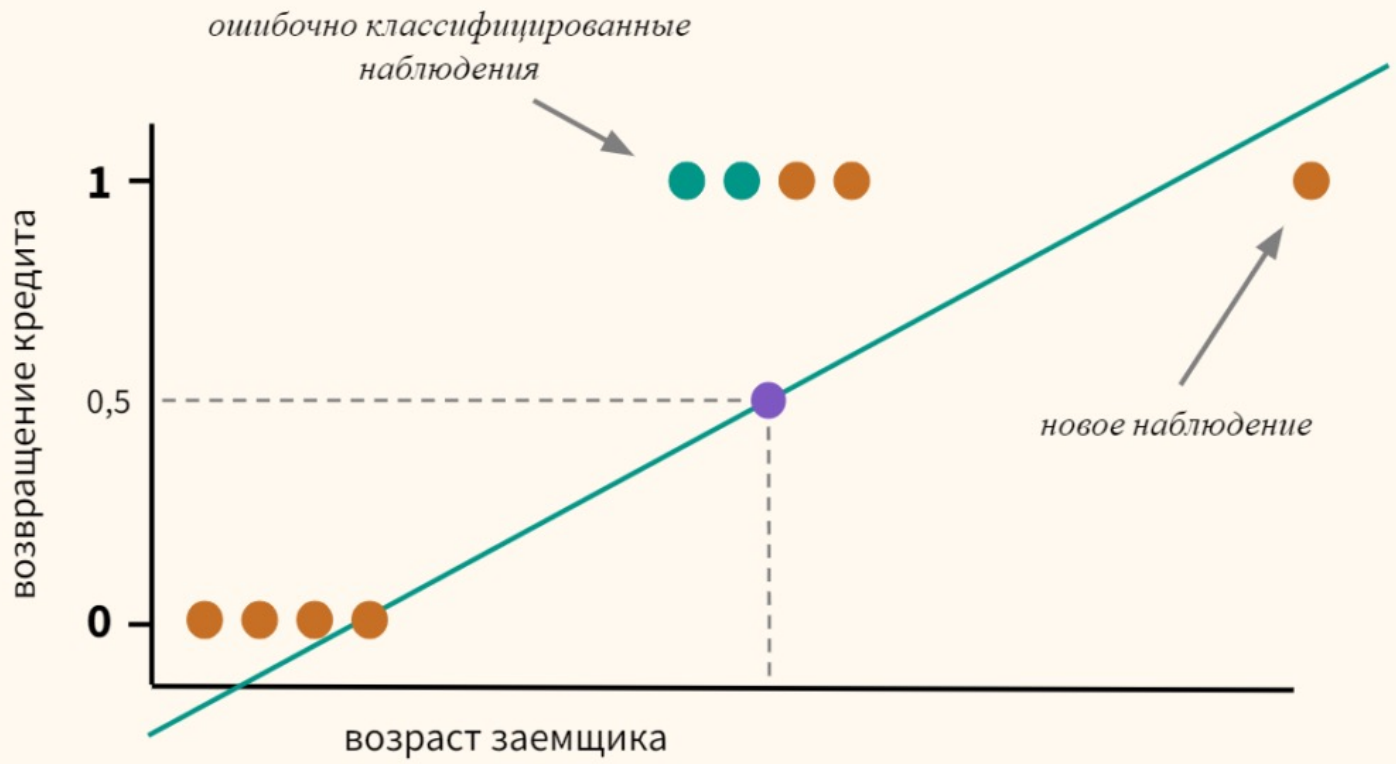


Казалось бы, можно построить линейную регрессию таким образом, чтобы она выдавала некоторое значение и, если это значение окажется ниже 0,5 — отнести наблюдение к классу 0, если выше — к классу 1.

- Если $f_w(x) < 0,5 \rightarrow \hat{y} = 0$
- Если $f_w(x) \geq 0,5 \rightarrow \hat{y} = 1$

Задача бинарной классификации:

Рассмотрим задачу кредитного скоринга:



Однако, появление новых данных сместит эту границу, и, как следствие, ничего не добавит, а только ухудшит точность модели.

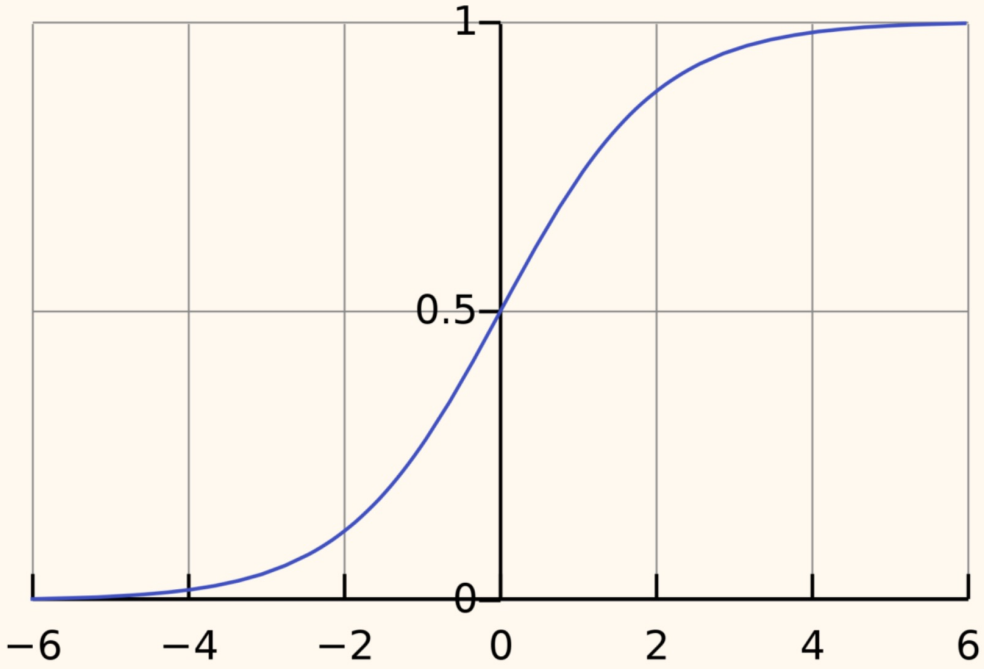
Теперь часть наблюдений, принадлежащих к классу 1, будет ошибочно отнесено моделью к классу 0.

Кроме этого, линейная регрессия по оси у выдает значения, сильно выходящие за пределы интересующего нас интервала от нуля до единицы.

Функция лог. регрессии

Возможное решение упомянутых сложностей — пропустить значение линейной регрессии через **сигмоиду** (sigmoid function), которая при любом значении X не выйдет из необходимого нам диапазона $0 \leq h(x) \leq 1$**:**

$$g(z) = \frac{1}{1 + e^{-z}}$$



Функция лог. регрессии

Тогда мы можем построить линейную модель, значение которой будет подаваться в сигмоиду.

$$h(x) = \frac{1}{1 + e^{-(kx+b)}}$$

В этом смысле никакой ошибки в названии «логистическая регрессия» нет. Этот алгоритм решает задачу классификации через модель линейной регрессии.

Интерпретация коэффициентов

Для любого значения x через $h(x)$ мы будем получать вероятность от 0 до 1, что объект принадлежит к классу y = 1. Например, если класс 1 означает, что заемщик вернул кредит, то $h(x)$ = 0.8 говорит о том, что согласно нашей модели, для данного заемщика (x) вероятность возвращения кредита составляет 80 процентов.

В общем случае мы можем записать вероятность вот так:

$$h(x) = P(y = 1 \mid x; (k, b) = \theta)$$

Для y = 0 получим соответственно обратную вероятность.

Решающая граница

Решающая граница (decision boundary) — это порог, который определяет к какому классу отнести то или иное наблюдение. Если выбрать порог на уровне 0,5, то все что выше или равно этому порогу мы отнесем к классу 1, все что ниже — к классу 0.

$$y = 1, h_\theta(x) \geq 0,5$$

$$y = 0, h_\theta(x) < 0,5$$

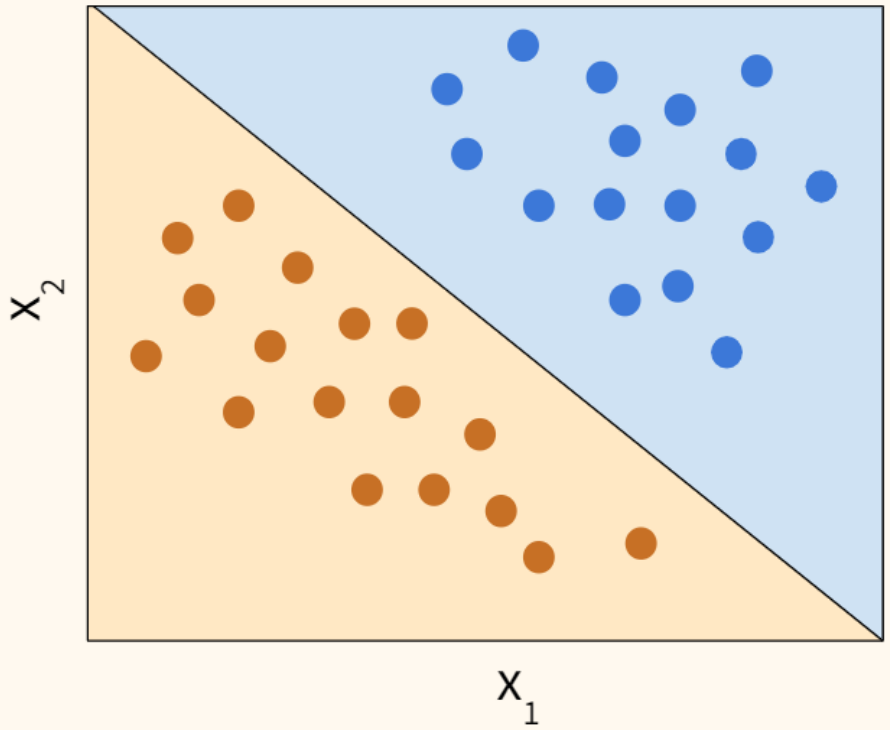
Обратите внимание на сигмоиду. Сигмоида принимает значения больше 0.5, если, $z \geq 0$.

- $h_\theta(x) \geq 0,5$ и $y = 1$, когда $kx + b \geq 0$
- $h_\theta(x) < 0,5$ и $y = 0$, когда $kx + b < 0$

Уравнение решающей границы

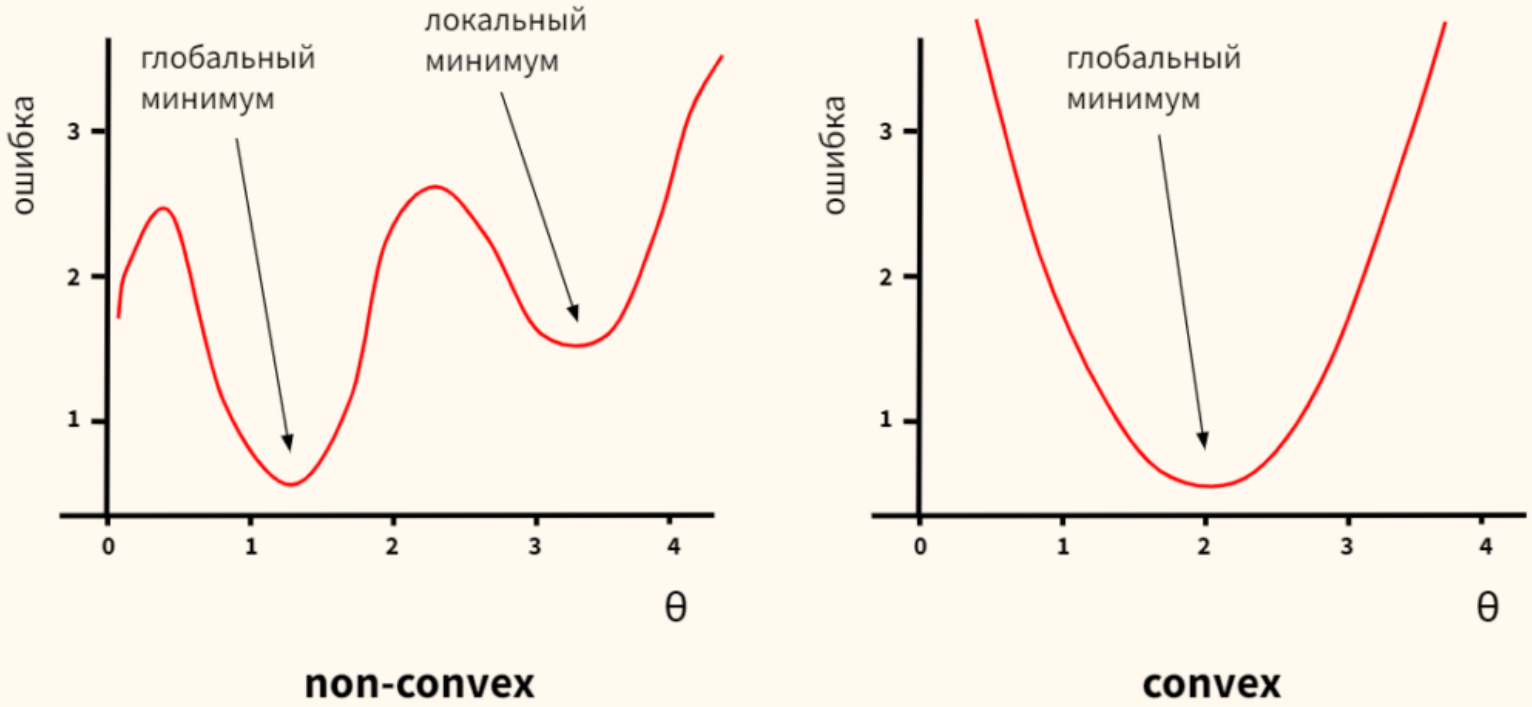
Предположим, что у нас есть два признака x1 и x2. Вместе они образуют некоторое пространство. Мы можем представить это пространство на координатной плоскости, дополнительно выделив цветом наблюдения, относящиеся к разным классам.

Кроме того, представим, что мы уже построили модель логистической регрессии, и она провела для нас соответствующую границу между двумя классами.



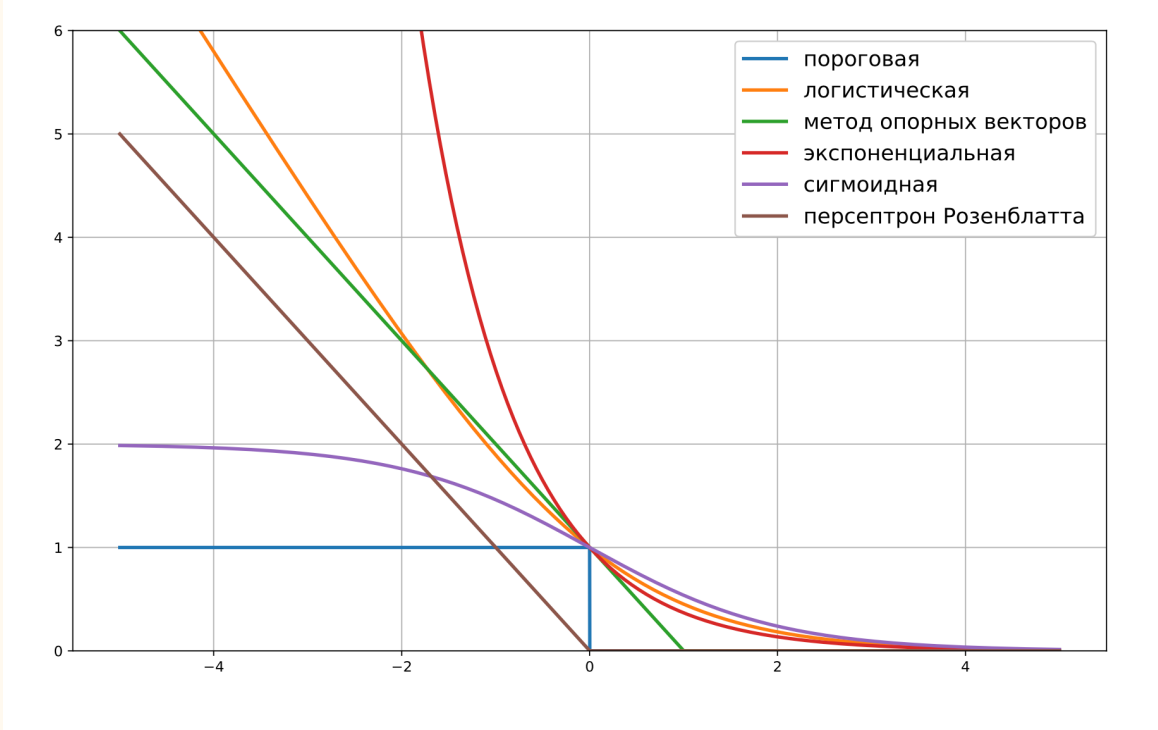
Функция ошибки

В модели логистической регрессии мы не можем использовать MSE. Дело в том, что если мы поместим результат сигмоиды (представляющей собою нелинейную функцию) в MSE, то на выходе получим невыпуклую функцию (non-convex), глобальный минимум которой довольно сложно найти.



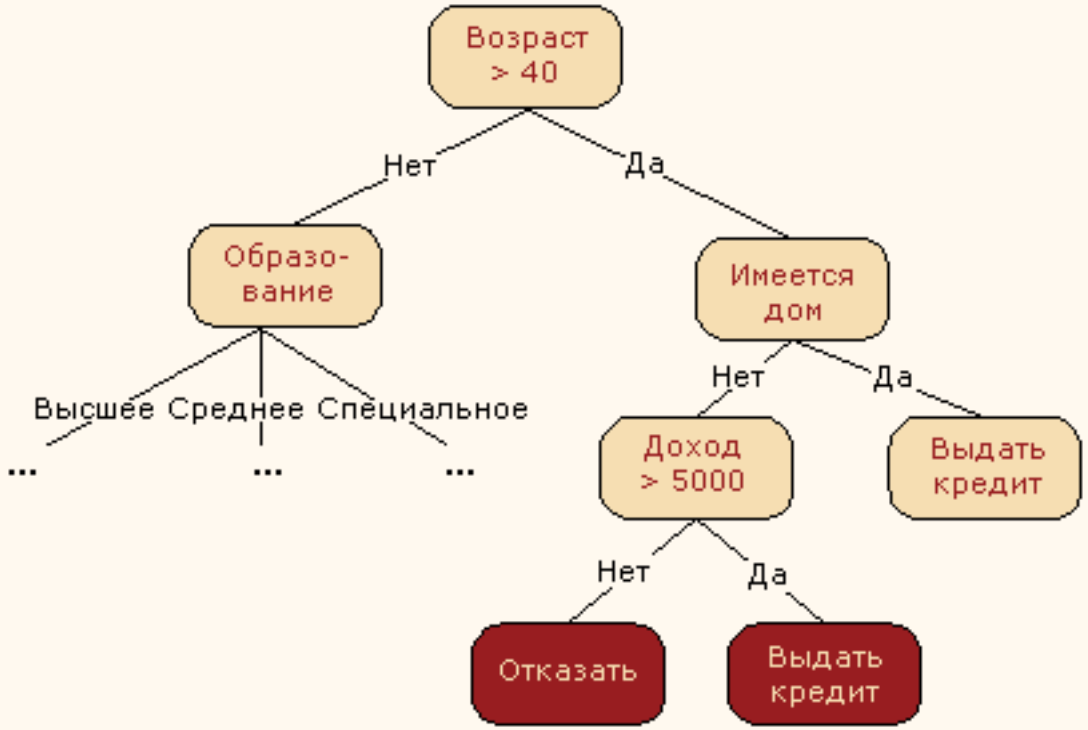
Функция ошибки

Можем ввести так называемую «пороговую функцию потерь» которая будет говорить нам 0 (если ошиблись) или 1 (иначе).



Деревья решений

Деревья решений — это метод в машинном обучении, используемый для классификации и регрессии. Они помогают в принятии решений, разбивая данные на подмножества на основе различных критериев. Каждое "разветвление" дерева представляет собой точку решения, где выбирается один из нескольких возможных вариантов, а каждый "лист" дерева представляет конечный результат или решение.



Энтропия

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

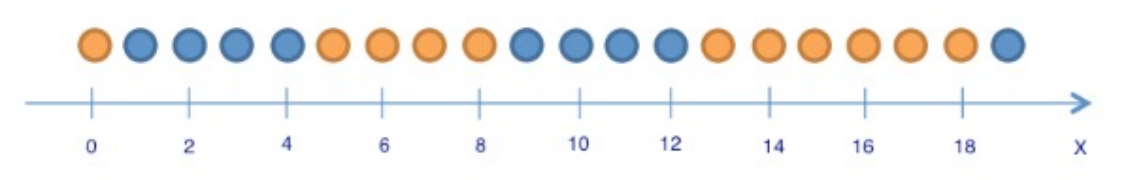
$$S = -\sum_{i=1}^{N} p_i \log_2 p_i ,$$

где p_i вероятности нахождения системы в i-ом состоянии.

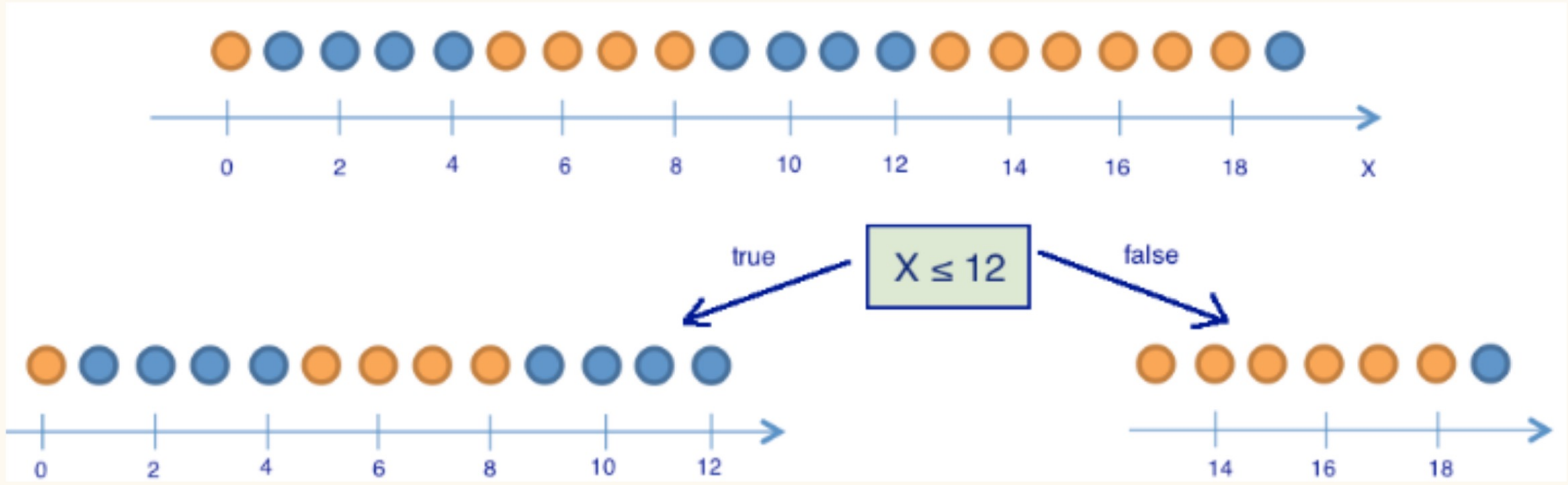
Это очень важное понятие, используемое в физике, теории информации и других областях. Опуская предпосылки введения (комбинаторные и теоретико-информационные) этого понятия, отметим, что, интуитивно, энтропия соответствует степени хаоса в системе. Чем выше энтропия, тем менее упорядочена система и наоборот. Это поможет нам формализовать «эффективное разделение выборки».

Энтропия

Для иллюстрации того, как энтропия поможет определить хорошие признаки для построения дерева, приведем тот же игрушечный пример. Будем предсказывать цвет шарика по его координате:



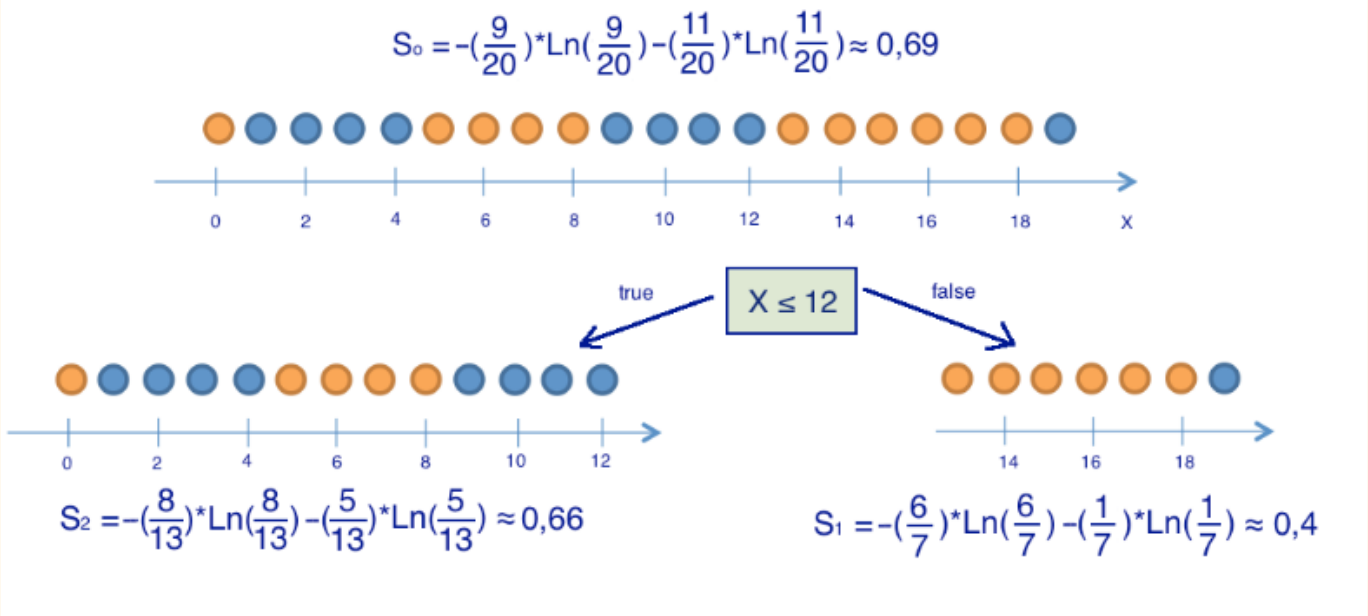
Здесь 9 синих шариков и 11 желтых. Если мы наудачу вытащили шарик, то он с вероятностью $p_1 = \frac{9}{20}$ будет синим и с вероятностью $p_2 = \frac{11}{20}$ – желтым. Значит, энтропия состояния $S_0 = -\frac{9}{20}\log_2 \frac{9}{20} - \frac{11}{20}\log_2 \frac{11}{20} \approx 1$. Само это значение пока ни о чем нам не говорит. Теперь посмотрим, как изменится энтропия, если разбить шарики на две группы – с координатой меньше либо равной 12 и больше 12.



Энтропия

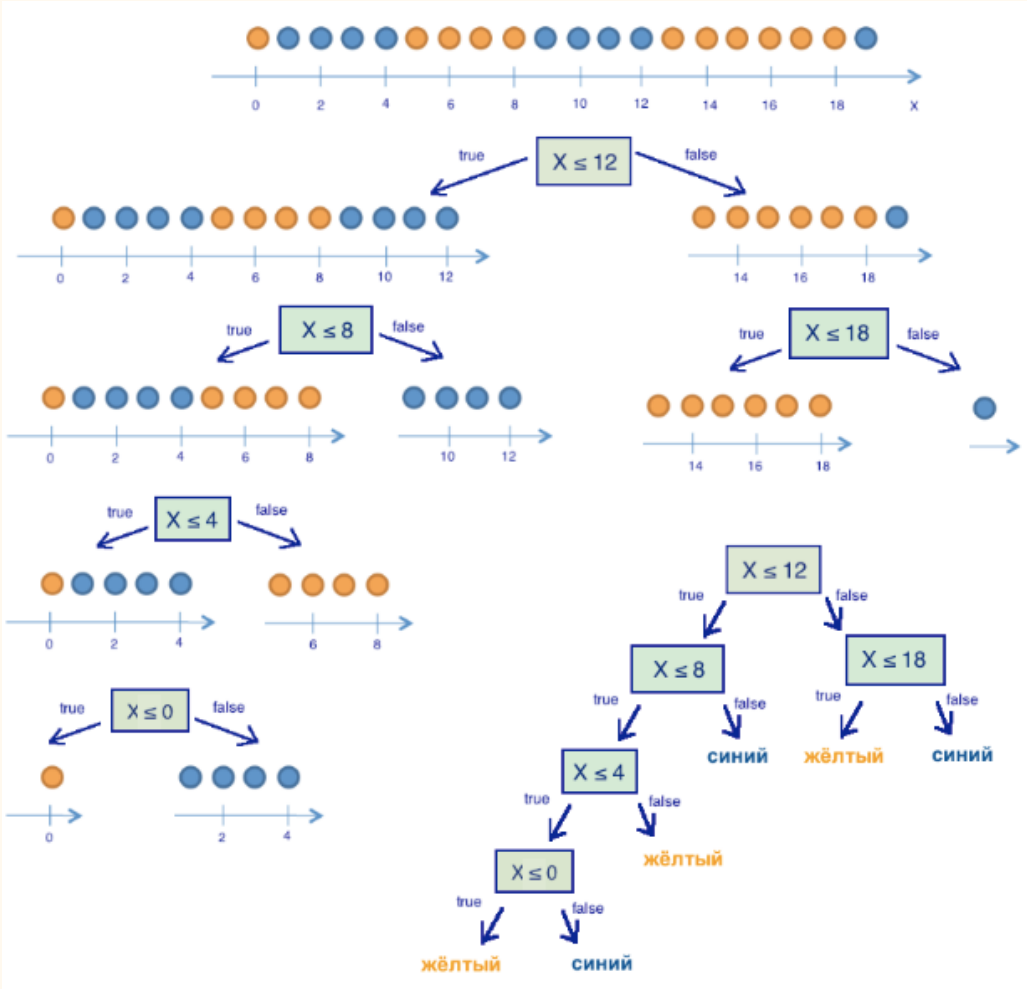
Поскольку энтропия – по сути степень хаоса (или неопределенности) в системе, уменьшение энтропии называют приростом информации. Формально прирост информации (information gain, IG) при разбиении выборки по признаку определяется как:

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



Энтропия

Получается, разделив шарики на две группы по признаку "координата меньше либо равна 12", мы уже получили более упорядоченную систему, чем в начале. Продолжим деление шариков на группы до тех пор, пока в каждой группе шарики не будут одного цвета.



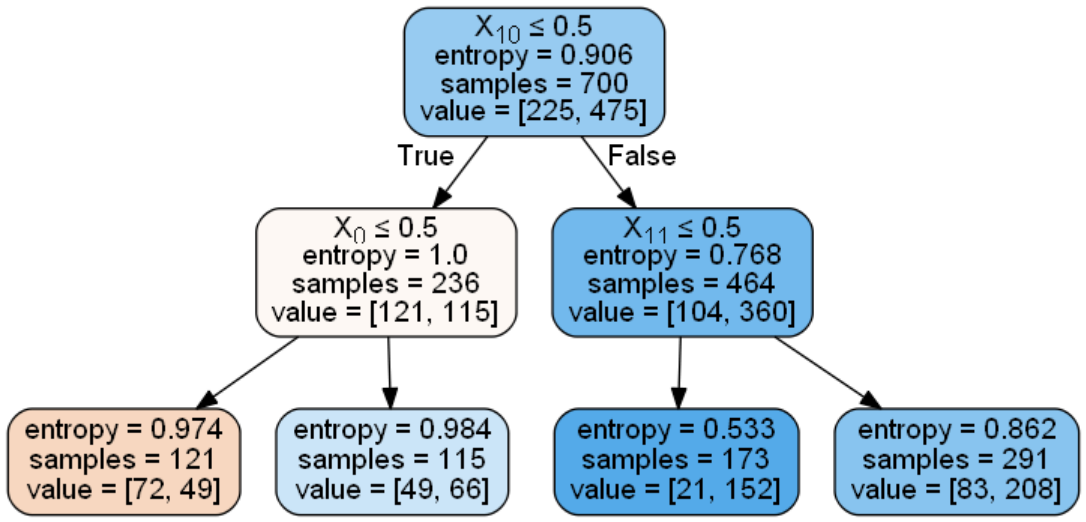
Алгоритм

1. **Инициализация**: Весь тренировочный набор данных считается корнем дерева.

2. **Выбор Признака для Разбиения**:
   - Рассчитайте энтропию текущего узла.
   - Для каждого признака вычислите прирост информации (разница между начальной энтропией и средневзвешенной энтропией после разбиения).
   - Выберите признак с максимальным приростом информации для разбиения.

3. **Разбиение**: Создайте подузлы, используя выбранный признак.

4. **Рекурсия**: Повторяйте шаги 2 и 3 для каждого нового подузла, пока не достигнуты условия остановки (например, все данные в узле принадлежат одному классу).



Итоги

Логистическая Регрессия:

- **Тип**: Линейная модель классификации.
- **Основа**: Использует логистическую функцию (сигмоид) для оценки вероятностей принадлежности к классам.
- **Применение**: Хорошо подходит для бинарной классификации.
- **Преимущества**: Проста в понимании и реализации, выдает вероятностные результаты.
- **Ограничения**: Предполагает линейные отношения между признаками и логарифмом шансов.

Деревья Решений:

- **Тип:** Нелинейная модель классификации и регрессии.
- **Структура**: Иерархическая структура с ветвлениями, представляющая решения и их возможные последствия.
- **Выбор Разбиения**: Использует меры как энтропию и прирост информации для определения лучших точек разбиения.
- **Преимущества**: Легко интерпретируемы, могут обрабатывать как числовые, так и категориальные данные.
- **Ограничения**: Склонны к переобучению, чувствительны к изменениям в данных.