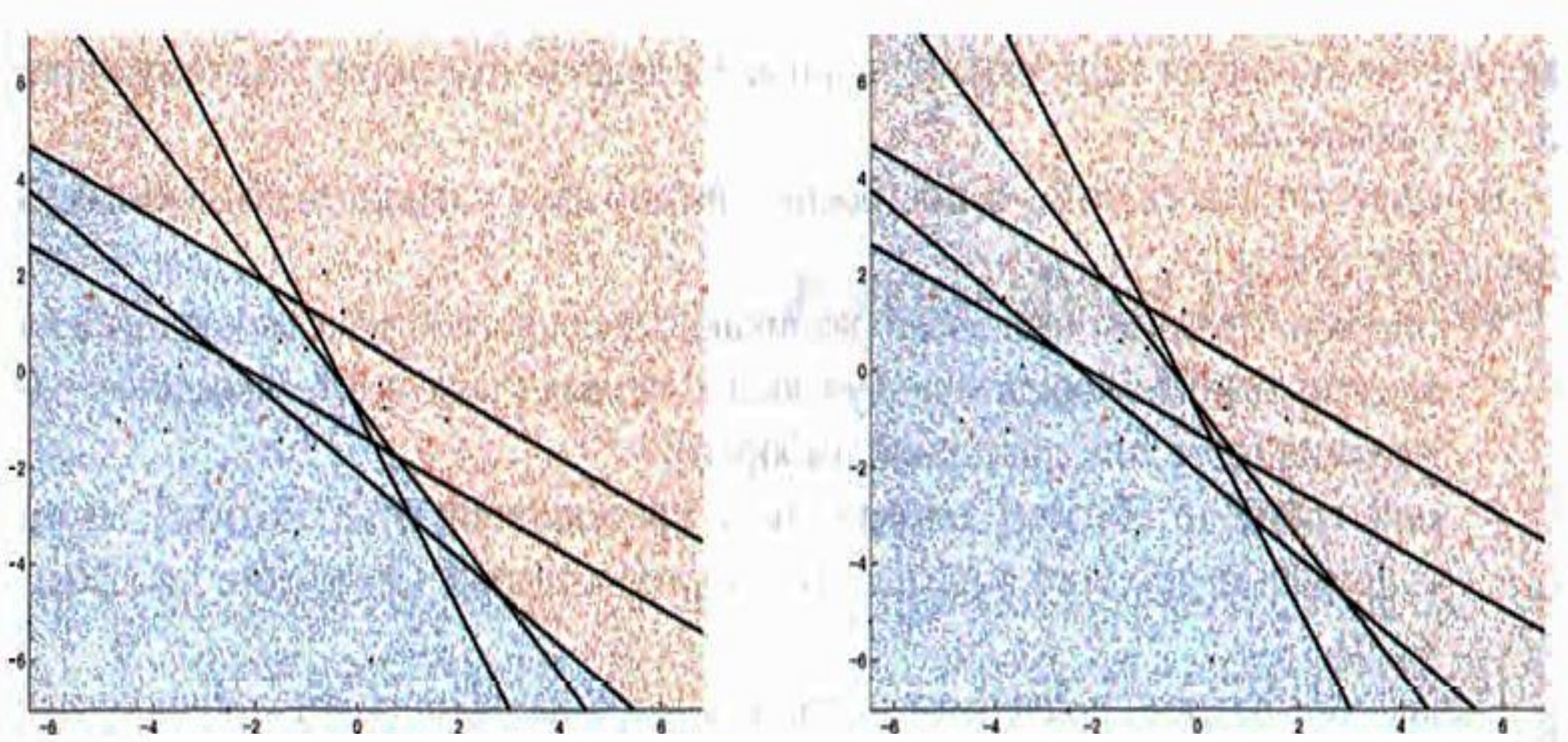
АНСАМБЛИ МОДЕЛЕЙ

Бэггинг, бустинг,
градиентный бустинг

Бэггинг

- **Бэггинг** (баггинг) - от англ. ***B****ootstrap* ***agg****regat***ing**, это технология классификации, использующая ансамбли моделей, каждая из которых обучается независимо. Результат классификации определяется путем голосования. Бэггинг позволяет снизить процент ошибки классификации в случае, когда высока дисперсия ошибки базового метода.
- Пример бэггинга – случайный лес

Бэггинг линейных классификаторов

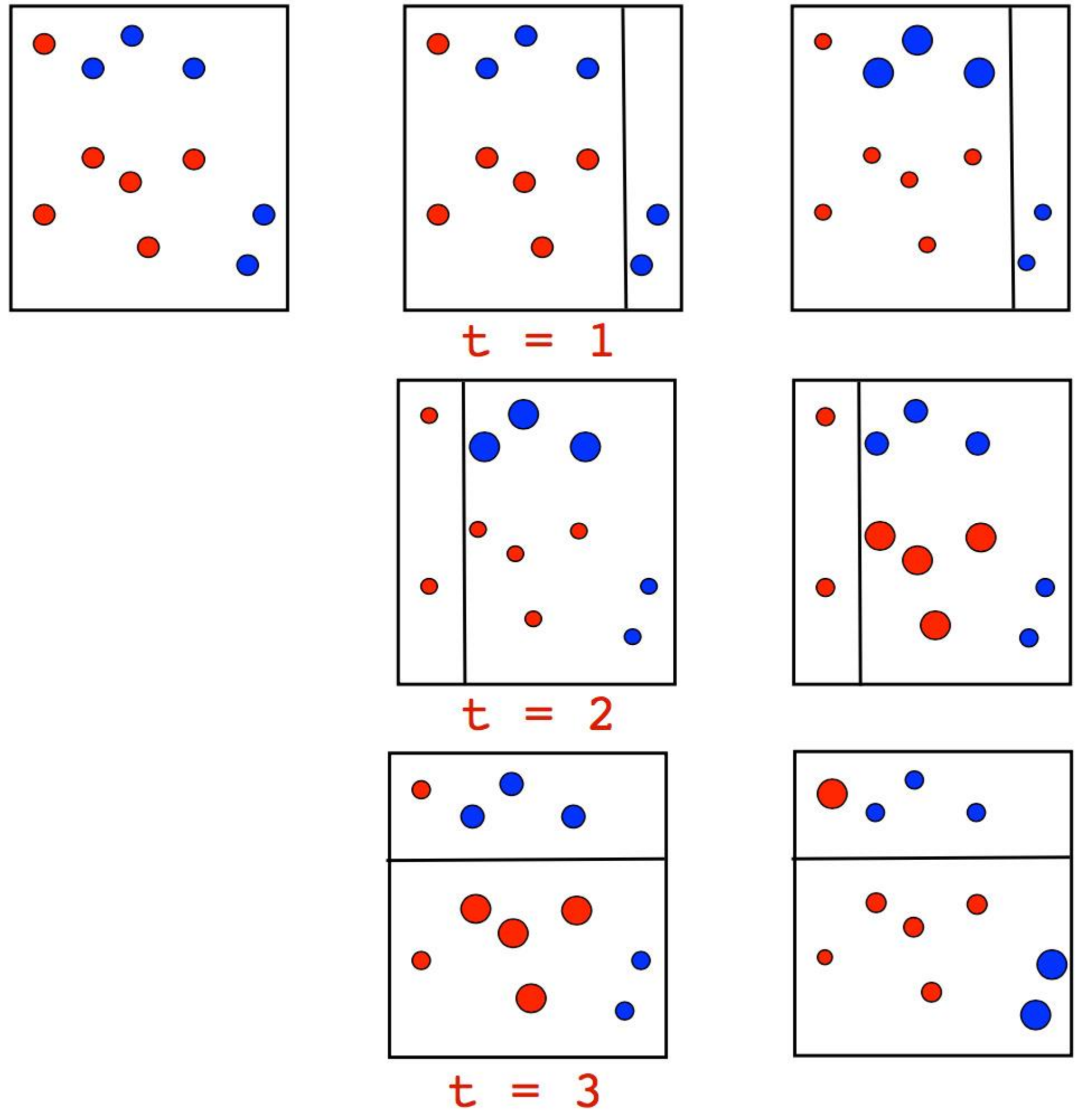


(Слева) Ансамбль из пяти базовых линейных классификаторов, построенный из усиливающих выборок с помощью баггинга. Решающим правилом является большинство голосов, оно порождает кусочно-линейную решающую границу. **(Справа)** Если преобразовать голоса в вероятности, то ансамбль превращается в группирующую модель: каждый сегмент пространства объектов получает немного отличающуюся вероятность

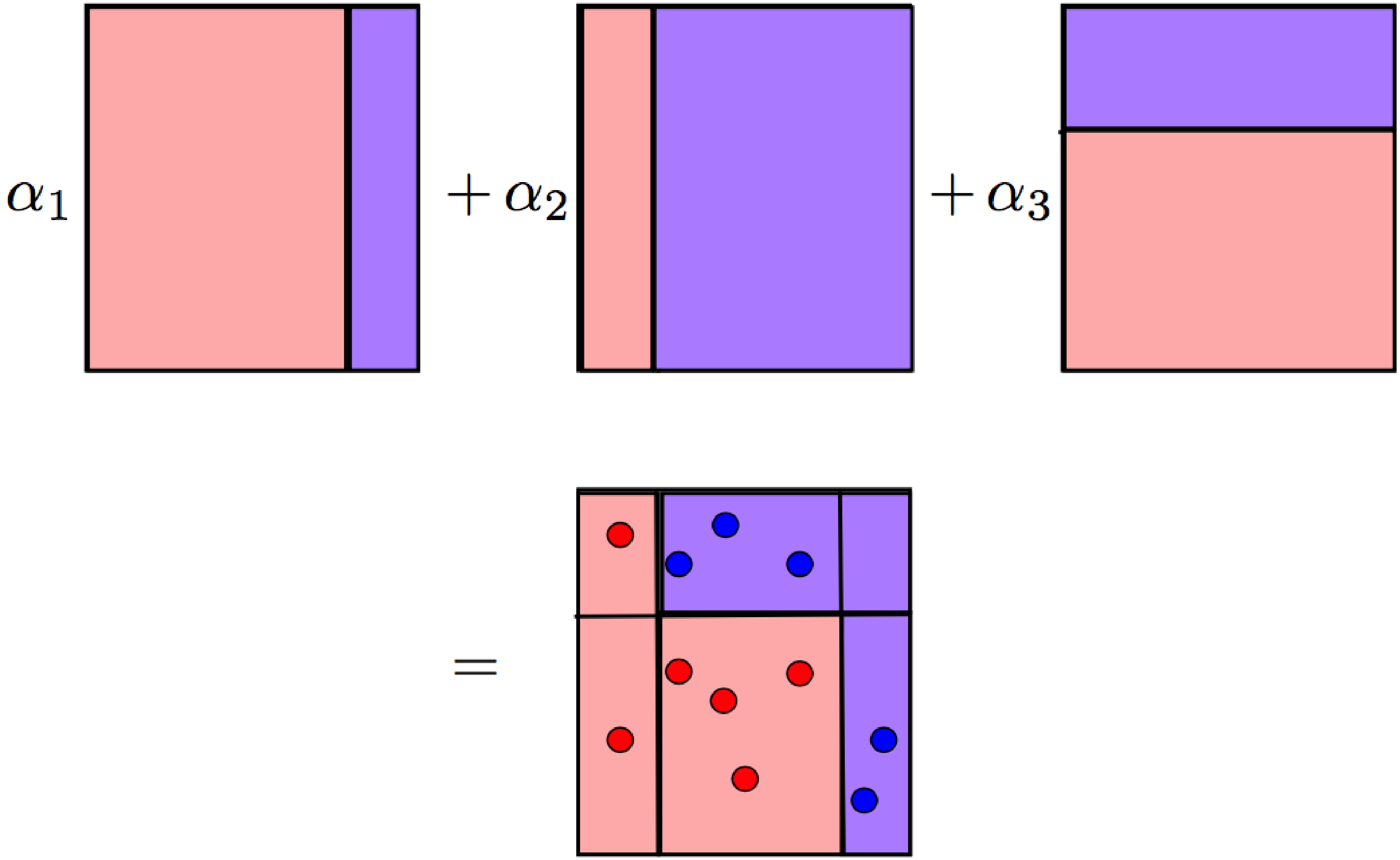
Бустинг

- Бустинг (англ. boosting — улучшение, усиление) — это процедура последовательного направленного построения ансамбля моделей машинного обучения, когда каждый следующий алгоритм стремится компенсировать ошибки предыдущих алгоритмов. Изначально понятие бустинга возникло в связи с вопросом: возможно ли, имея множество относительно слабых (незначительно отличающихся от случайных) и простых моделей, построить сильную модель?

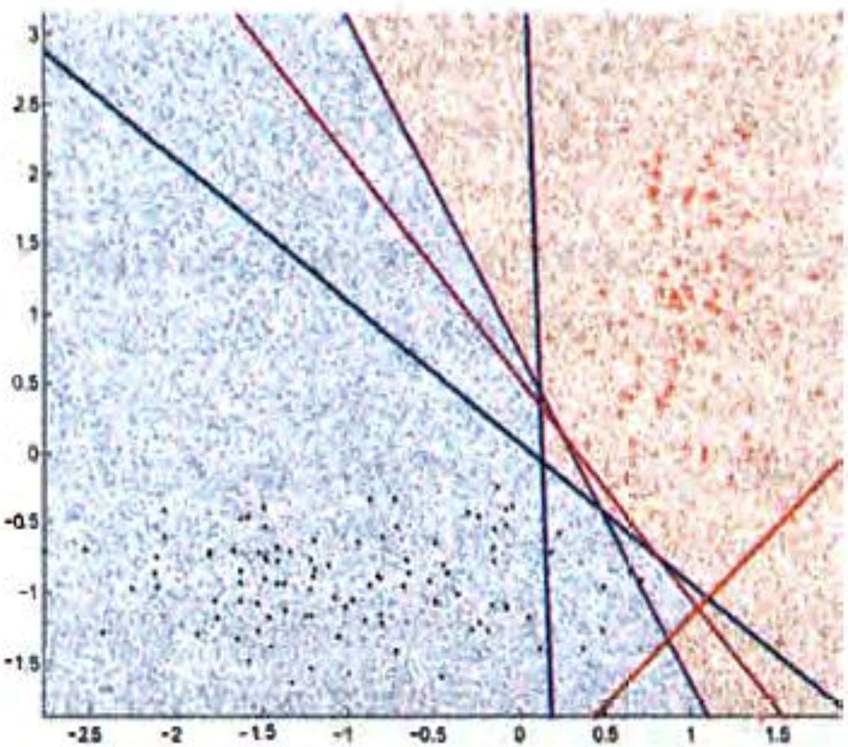
Пример работы Adaboost для ансамбля из трех простых деревьев (пней)

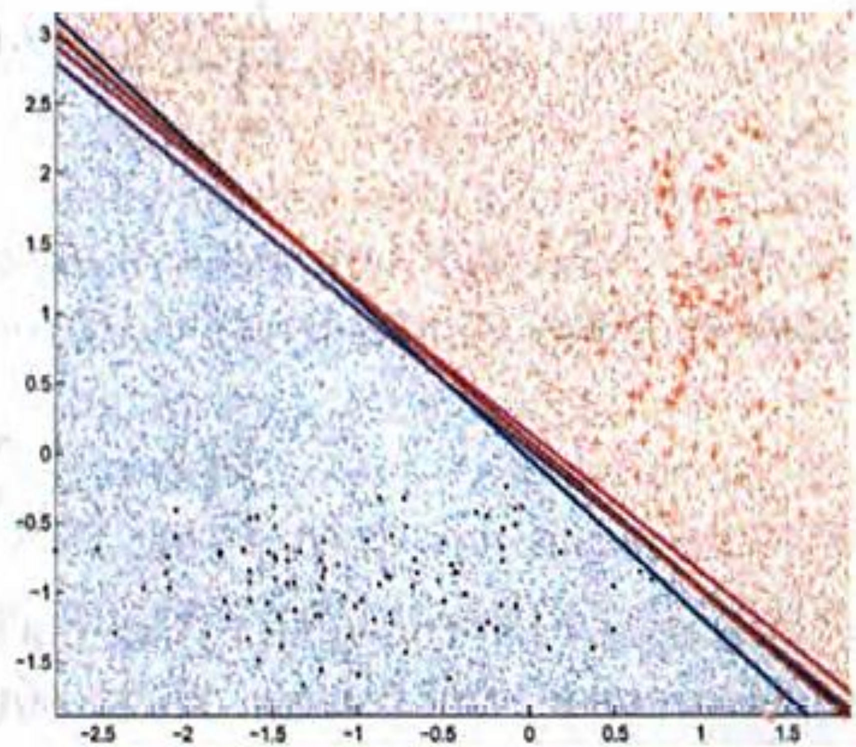


Пример работы Adaboost (итоговый классификатор)



Сравнение результатов бустинга для слабых и сильных моделей





Градиентный бустинг

- Градиентный бустинг - класс алгоритмов, представляющих бустинг как процесс градиентного спуска. В основе алгоритма лежит последовательное уточнение функции, представляющей собой линейную комбинацию базовых классификаторов, с тем чтобы минимизировать дифференцируемую функцию потерь. Градиентный бустинг - это один из самых универсальных и сильных методов машинного обучения, известных на сегодняшний день. В частности, на градиентном бустинге над деревьями решений основан алгоритм ранжирования выдачи компании Яндекс.

Градиентный бустинг в задаче регрессии

$$\frac{1}{2}\sum_{i=1}^{\ell}(a(x_i)-y_i)^2 \to \min_a$$

$$a_N(x) = \sum_{n=1}^{N} b_n(x),$$

$$b_1(x) := \arg\min_{b\in\mathcal{A}} \frac{1}{2}\sum_{i=1}^{\ell}(b(x_i)-y_i)^2$$

$$s_i^{(1)} = y_i - b_1(x_i)$$

$$b_2(x) := \arg\min_{b\in\mathcal{A}} \frac{1}{2}\sum_{i=1}^{\ell}(b(x_i)-s_i^{(1)})^2$$

Градиентный бустинг в задаче регрессии

Каждый следующий алгоритм тоже будем настраивать на остатки предыдущих:

$$s_i^{(N)} = y_i - \sum_{n=1}^{N-1} b_n(x_i) = y_i - a_{N-1}(x_i), \qquad i = 1, \ldots, \ell;$$

$$b_N(x) := \arg\min_{b \in \mathcal{A}} \frac{1}{2} \sum_{i=1}^{\ell} (b(x_i) - s_i^{(N)})^2$$

Заметим, что остатки могут быть найдены как антиградиент функции потерь

$$s_i^{(N)} = y_i - a_{N-1}(x_i) = -\frac{\partial}{\partial z} \frac{1}{2}(z - y_i)^2 \Big|_{z = a_{N-1}(x_i)}$$

Таким образом, выбирается такой базовый алгоритм, который как можно сильнее уменьшит ошибку композиции это свойство вытекает из его близости к антиградиенту функционала на обучающей выборке.

Градиентный бустинг (общий случай)

- Пусть дана некоторая дифференцируемая функция потерь L(y, z). Построим взвешенную сумму базовых алгоритмов:

$$a_N(x) = \sum_{n=0}^{N} \gamma_n b_n(x)$$

- Способы выбора $b_0(x)$:
- самый популярный класс (в задачах классификации):

$$b_0(x) = \arg\max_{y \in \mathbb{Y}} \sum_{i=1}^{\ell} [y_i = y]$$

средний ответ (в задачах регрессии):

$$b_0(x) = \frac{1}{\ell} \sum_{i=1}^{\ell} y_i$$

Градиентный бустинг (продолжение)

Допустим, мы построили композицию $a_{N-1}(x)$ из $N-1$ алгоритма, и хотим выбрать следующий базовый алгоритм $b_N(x)$ так, чтобы как можно сильнее уменьшить ошибку:

$$\sum_{i=1}^{\ell} L(y_i, a_{N-1}(x_i) + \gamma_N b_N(x_i)) \to \min_{b_N, \gamma_N}$$

$$\sum_{i=1}^{\ell} L(y_i, a_{N-1}(x_i) + s_i) \to \min_{s_1, \ldots, s_\ell}$$

$$s_i = -\left.\frac{\partial L}{\partial z}\right|_{z=a_{N-1}(x_i)}$$

вектор сдвигов $s = (s_1, \ldots, s_\ell)$ совпадает с антиградиентом:

$$\left(-\left.\frac{\partial L}{\partial z}\right|_{z=a_{N-1}(x_i)}\right)_{i=1}^{\ell} = -\nabla_z \sum_{i=1}^{\ell} L(y_i, z_i)\Big|_{z_i=a_{N-1}(x_i)}$$

Градиентный бустинг (продолжение)

$$b_N(x) = \arg\min_{b \in A} \sum_{i=1}^{\ell} (b(x_i) - s_i)^2$$

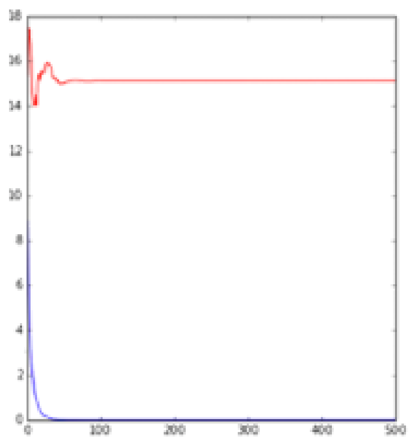
$$\gamma_N = \arg\min_{\gamma \in \mathbb{R}} \sum_{i=1}^{\ell} L(y_i, a_{N-1}(x_i) + \gamma b_N(x_i))$$

- Если базовые алгоритмы очень простые, то они плохо приближают вектор антиградиента. По сути, добавление такого базового алгоритма будет соответствовать шагу вдоль направления, сильно отличающегося от направления наискорейшего убывания.
- Если базовые алгоритмы сложные, то они способны за несколько шагов бустинга идеально подогнаться под обучающую выборку что будет являться переобучением, связанным с излишней сложностью семейства алгоритмов.

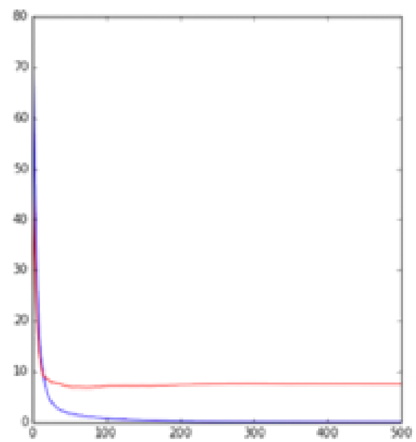
Сокращение шага

Решение проблемы - сокращение шага: вместо перехода в оптимальную точку в направлении антиградиента делается укороченный шаг

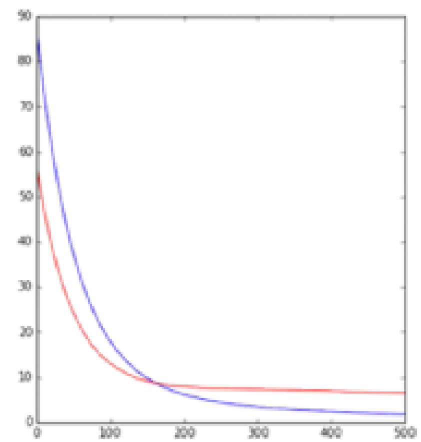
$$a_N(x) = a_{N-1}(x) + \eta \gamma_N b_N(x), \qquad \eta \in (0, 1]$$



$\eta = 1$



$\eta = 0.1$



$\eta = 0.01$

Функции потерь

- Регрессия: квадратичная $L(y,z) = \frac{1}{2}(y-z)^2$
- Классификация: логистическая $L(y,z) = \log(1+\exp(-yz)).$

$$L'_z(y,z) = -\frac{y}{1+\exp(-yz)}$$

$$s = \left(\frac{y_1}{1+\exp(-y_1 a_{N-1}(x_1))}, \dots \right.$$
$$\left. \dots, \frac{y_\ell}{1+\exp(-y_\ell a_{N-1}(x_\ell))}\right)$$

Градиентный бустинг над деревьями

Дерево разбивает все пространство на непересекающиеся области, в каждой из которых его ответ равен константе:

$$b_n(x) = \sum_{j=1}^{J_n} b_{nj}[x \in R_j],$$

где $j = 1, \dots, J_n$ — индексы листьев, R_j — соответствующие области разбиения, b_{nj} — значения в листьях. Значит, на N-й итерации бустинга композиция обновляется как

$$\sum_{i=1}^{\ell} L\left(y_i, a_{N-1}(x_i) + \sum_{j=1}^{J_N} \gamma_{Nj}[x \in R_j]\right) \to \min_{\{\gamma_{Nj}\}_{j=1}^{J_N}}.$$

Поскольку области разбиения R_j не пересекаются, данная задача распадается на J_N независимых подзадач:

$$\gamma_{Nj} = \arg\min_{\gamma} \sum_{x_i \in R_j} L(y_i, a_{N-1}(x_i) + \gamma), \qquad j = 1, \dots, J_N.$$