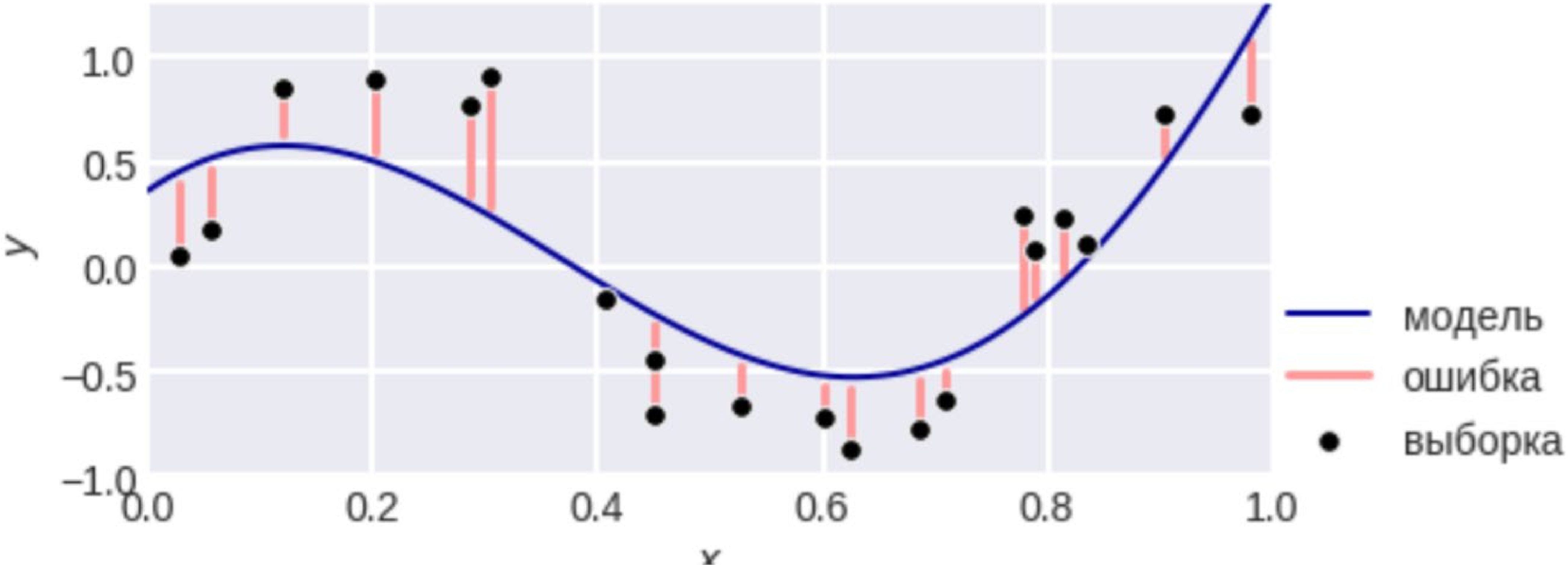
Градиентный бустинг

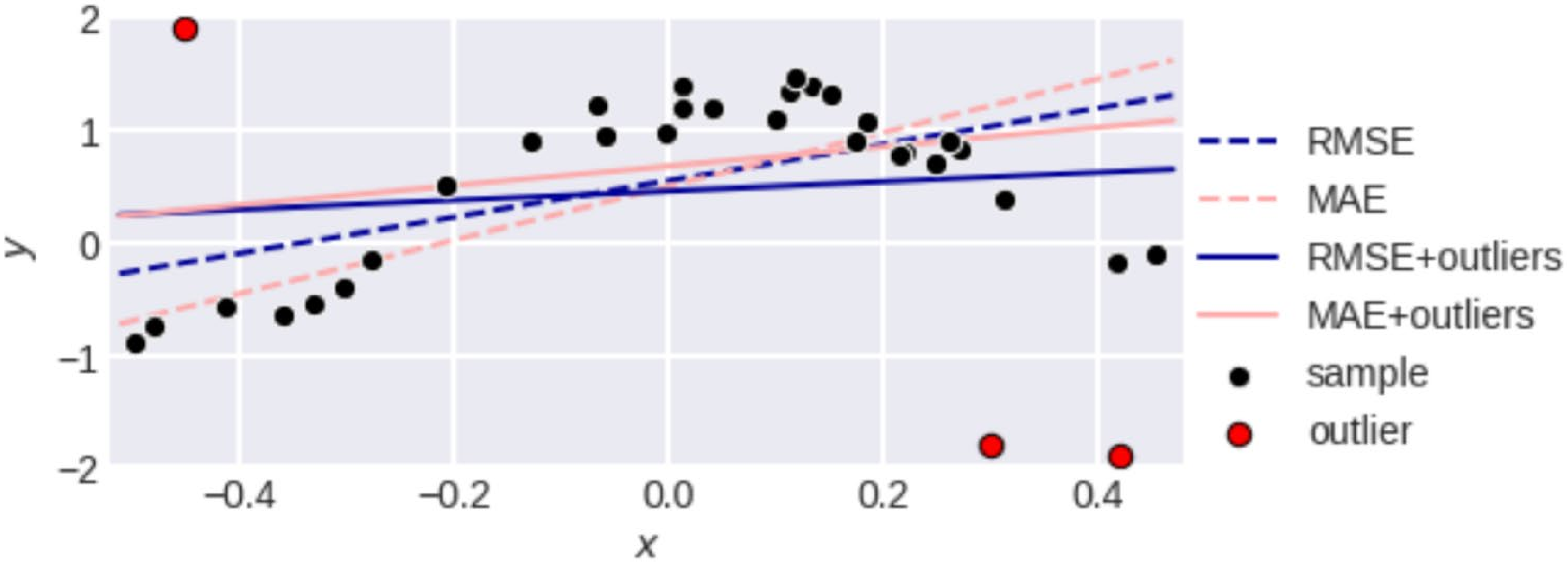
Метрики регрессии



Метрики регрессии

- $MaxError(y, \hat{y}) = \max(|y_i - \widehat{y_i}|)$
- $MeanAbsoluteError(y, \hat{y}) = MAE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}|y_i - \widehat{y_i}|$
- $MeanSquaredError(y, \hat{y}) = MSE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y_i})^2$
- $RootMSE(y, \hat{y}) = RMSE(y, \hat{y}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y_i})^2}$
- $MeanAbsolutePercentError(y, \hat{y}) = MAPE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \widehat{y_i}}{y_i}\right|$
- $R^2(y, \hat{y}) = 1 - \frac{MSE(model)}{MSE(baseline)} = 1 - \frac{\sum_{i=1}^{n}(y_i - \widehat{y_i})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$

Метрики регрессии



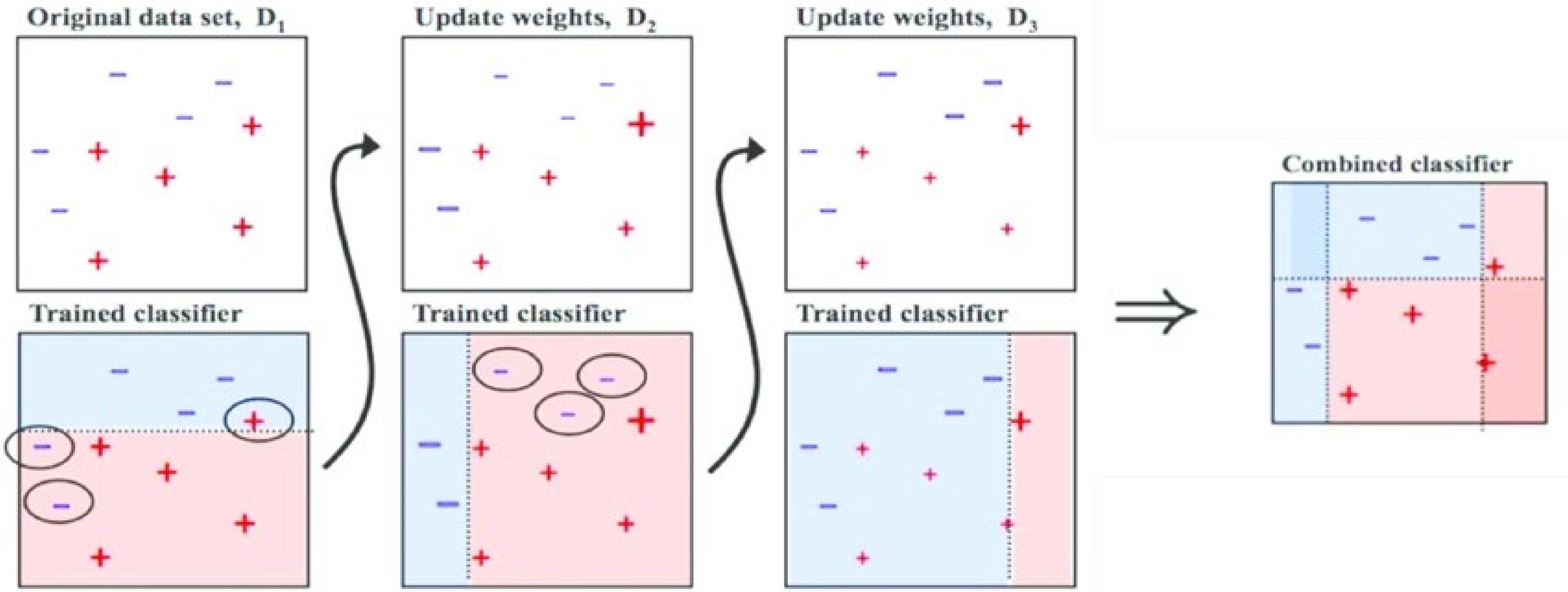
Бустинг

- Основан на одной из самых прорывных идей ML за последнее время
- Крайне популярен
- Стандарт для победы в ML соревнованиях

Основная идея - объединение большого количества “слабых” моделей в одну “сильную”

“Слабая” модель - любая модель, точность которой чуть лучше случайного угадывания

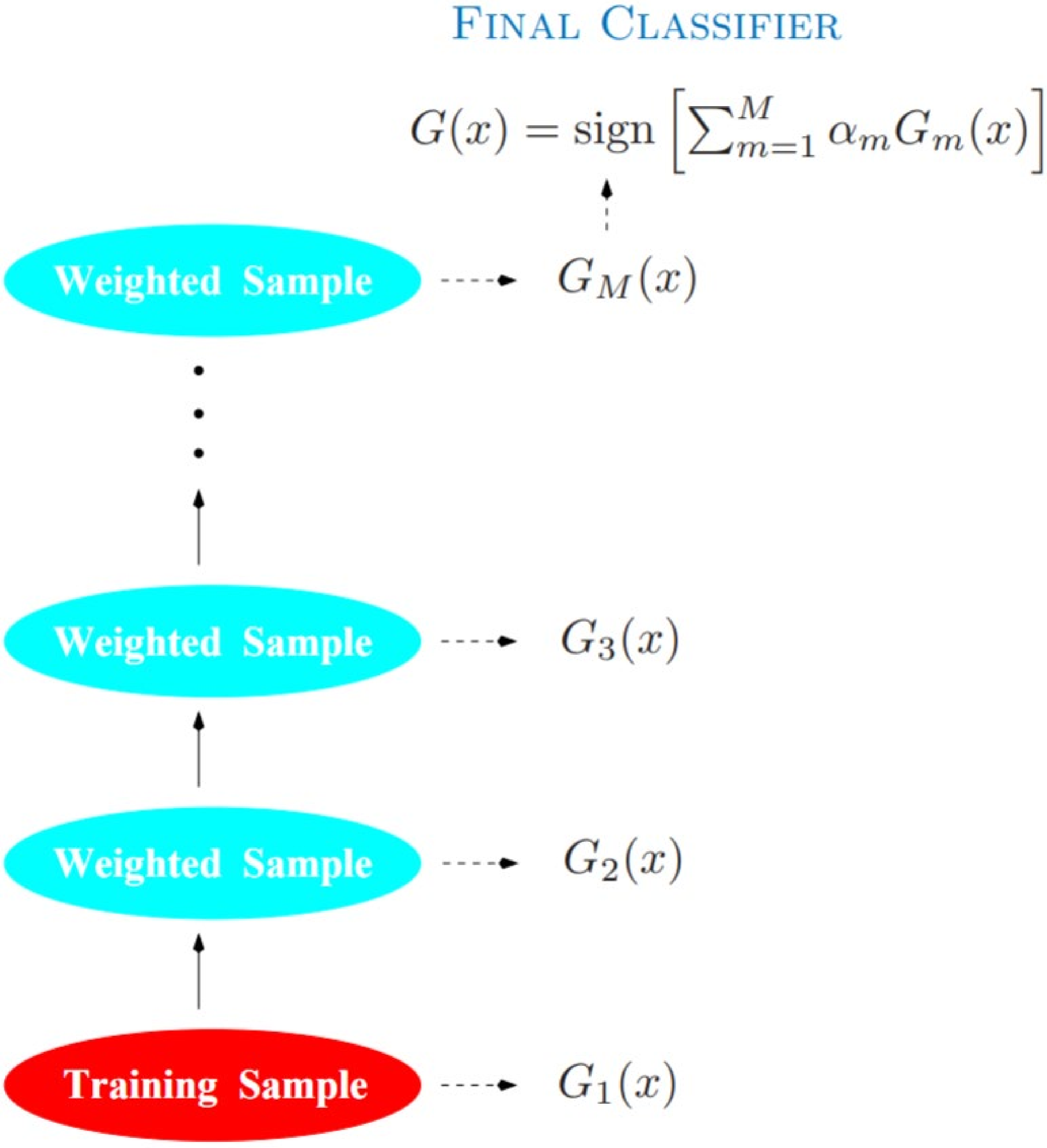
Adaboost



Original data set, D_1
Trained classifier
Update weights, D_2
Trained classifier
Update weights, D_3
Trained classifier
Combined classifier

Adaboost

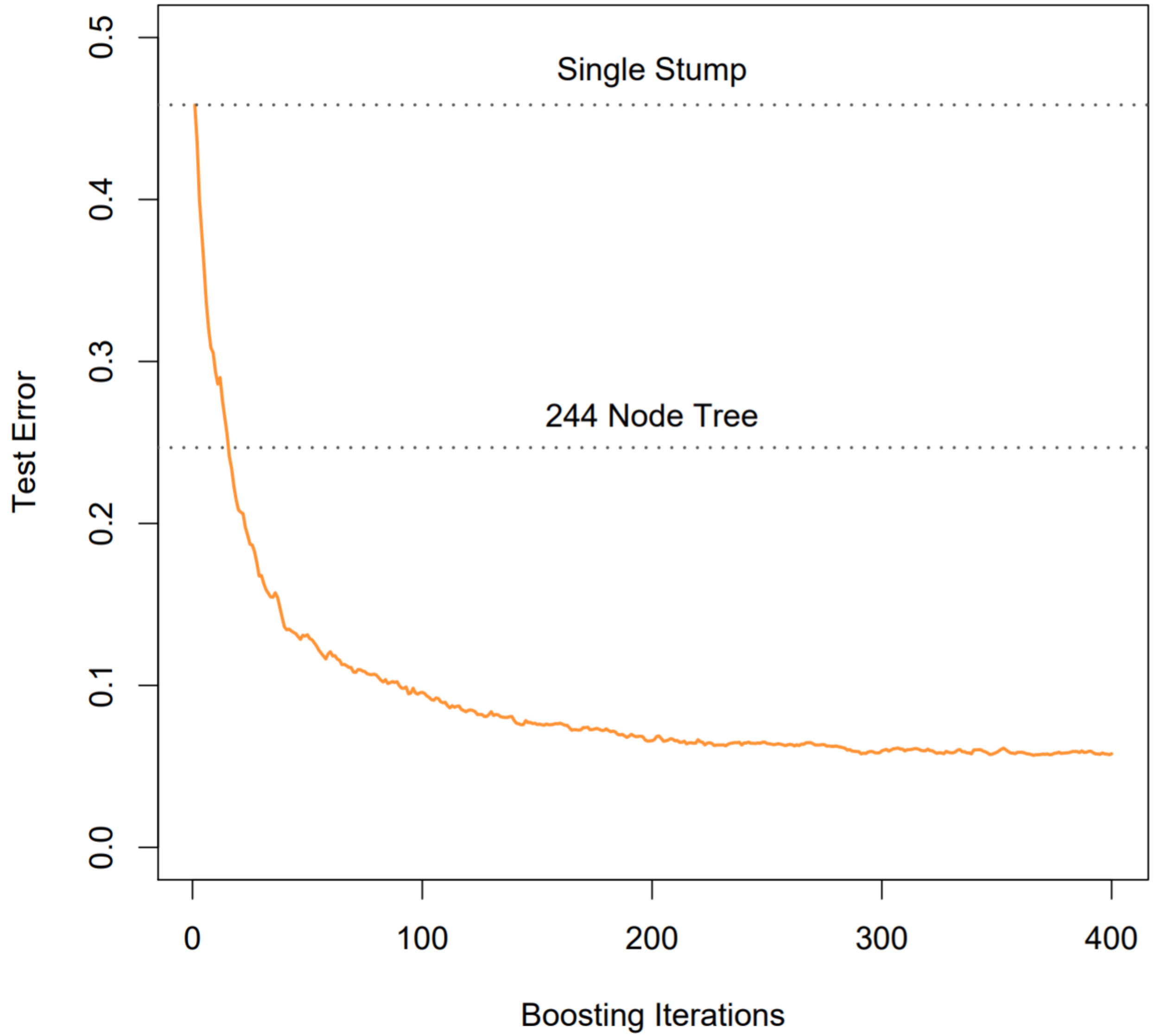
- Задача классификации
- $Y \in \{-1,1\}$
- $\overline{err} = \frac{1}{N}\sum_{i=1}^{N} I(y_i \neq G(x_i))$
- $G(x) = sign\left(\sum_{m=1}^{M} \propto_m G_m(x)\right)$



Adaboost

1. Инициализируем веса $\omega_i = \frac{1}{N}, i = 1,2, \dots, N$
2. Для каждой итерации от $m = 1$ до M:
   1. Обучить классификатор $G_m(x)$ на данных с весами ω_i
   2. Вычислить ошибку $err_m = \frac{\sum_{i=1}^{N} \omega_i I\left(y_i \neq G_m(x_i)\right)}{\sum_{i=1}^{N} \omega_i}$
   3. Вычислить веса $\propto_m = \log\left((1 - err_m)/err_m\right)$
   4. Обновить веса $\omega_i \leftarrow \omega_i \cdot \exp\left[\propto_m \cdot I\left(y_i \neq G_m(x_i)\right)\right], i = 1,2, \dots, N$
3. Результат $G(x) = sign\left[\sum_{m=1}^{M} \propto_m G_m(x)\right]$

Adaboost



Single Stump
244 Node Tree
Test Error
0.0
0.1
0.2
0.3
0.4
0.5
0
100
200
300
400
Boosting Iterations

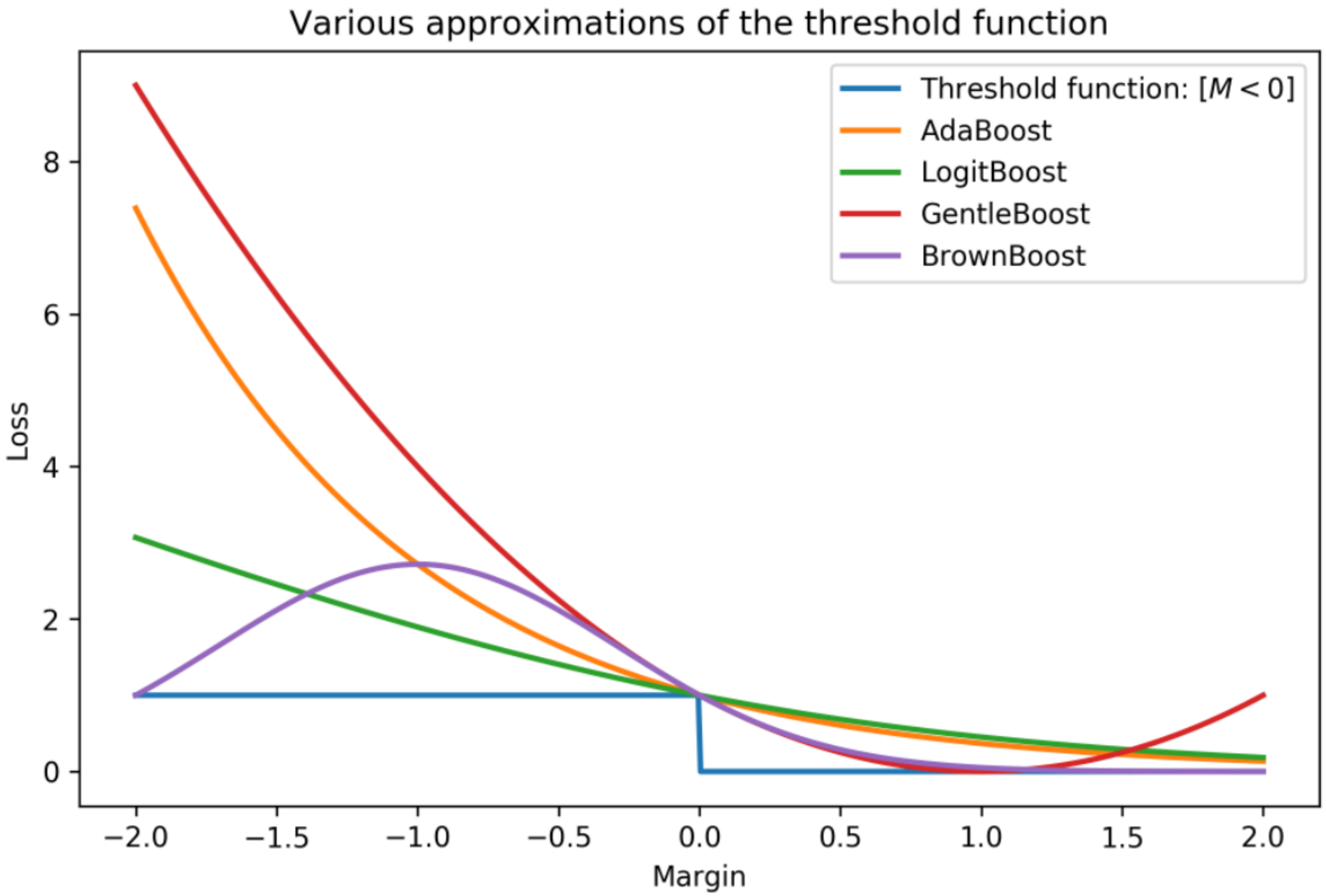
Adaboost

- $\overline{err} =$

  $\frac{1}{N}\sum_{i=1}^{N} I(y_i \neq G(x_i)) =$

  $\frac{1}{N}\sum_{i=1}^{N} I(y_i \cdot G(x_i) < 0) =$

  $\frac{1}{N}\sum_{i=1}^{N} \max(0, 1 - y_i \cdot G(x_i))$



GBM

- $x: \{x_1, \dots x_n\}, y: \{y_1, \dots y_n\}, F: x \to y$
- Найдем аппроксимацию $\hat{F}(x)$ для отношения F, которая минимизирует функцию потерь $L\big(y, F(x)\big)$

$$\hat{F} = arg \min_F L\big(y, F(x)\big)$$

- Будем искать как взвешенную сумму «слабых» функций $h_i(x)$

$$\hat{F}(x) = \sum_{i=1}^{M} \gamma_i h_i(x) + const$$

GBM

- По принципу минимизации эмпирического риска, начнем с константного приближения

$$F_0(x) = arg \min_{\gamma} \sum_{i=1}^{n} L(y_i, \gamma)$$

- И жадно будем улучшать фукнцию

$$F_m(x) = F_{m-1}(x) + arg \min_{h_m \in H} \left[\sum_{i=1}^{n} L\big(y_i, F_{m-1}(x_i) + h_m(x_i)\big) \right]$$

GBM

Входные данные:

- Размеченная выборка $\{(x_i, y_i)\}_{i=1}^{n}$
- Дифференцируемая функция потерь $L\big(y, F(x)\big)$
- Число итераций M
- Семейство алгоритмов H: $h(x, \theta)$
- Гиперпараметры

GBM

1. Инициализируем модель константным значением $F_0(x) = arg\min\limits_{\gamma} \sum_{i=1}^{n} L(y_i, \gamma)$
2. Для каждой итерации от $m = 1$ до M:
   1. Вычислим псевдо-остатки $r_{im} = -\left[\frac{\partial L\left(y_i, F(x_i)\right)}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}, i = 1, \ldots, n$
   2. Обучим $h_m(x)$ на $\{(x_i, r_{im})\}_{i=1}^{n}$
   3. Найдем вес $\gamma_m = arg\min_{\gamma} \sum_{i=1}^{n} L\left(y_i, F_{m-1}(x_i) + \gamma h_m(x_i)\right)$
   4. Обновим модель $F_m(x) = F_{m-1}(x) + \gamma h_m(x)$
3. Результат $F_M(x)$