Основы машинного обучения

Лекция 15

Градиентный бустинг

Евгений Соколов

esokolov@hse.ru

НИУ ВШЭ, 2024

Идея бустинга

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

- Обучение N-й модели:

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

Бустинг для MSE

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

- Обучение N-й модели:

$$\frac{1}{\ell}\sum_{i=1}^{\ell}(a_{N-1}(x_i) + b_N(x_i) - y_i)^2 \to \min_{b_N(x)}$$

Бустинг для MSE

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

- Обучение N-й модели:

$$\frac{1}{\ell} \sum_{i=1}^{\ell} \Big(b_N(x_i) - \big(y_i - a_{N-1}(x_i) \big) \Big)^2 \to \min_{b_N(x)}$$

Бустинг для MSE

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

- Обучение N-й модели:

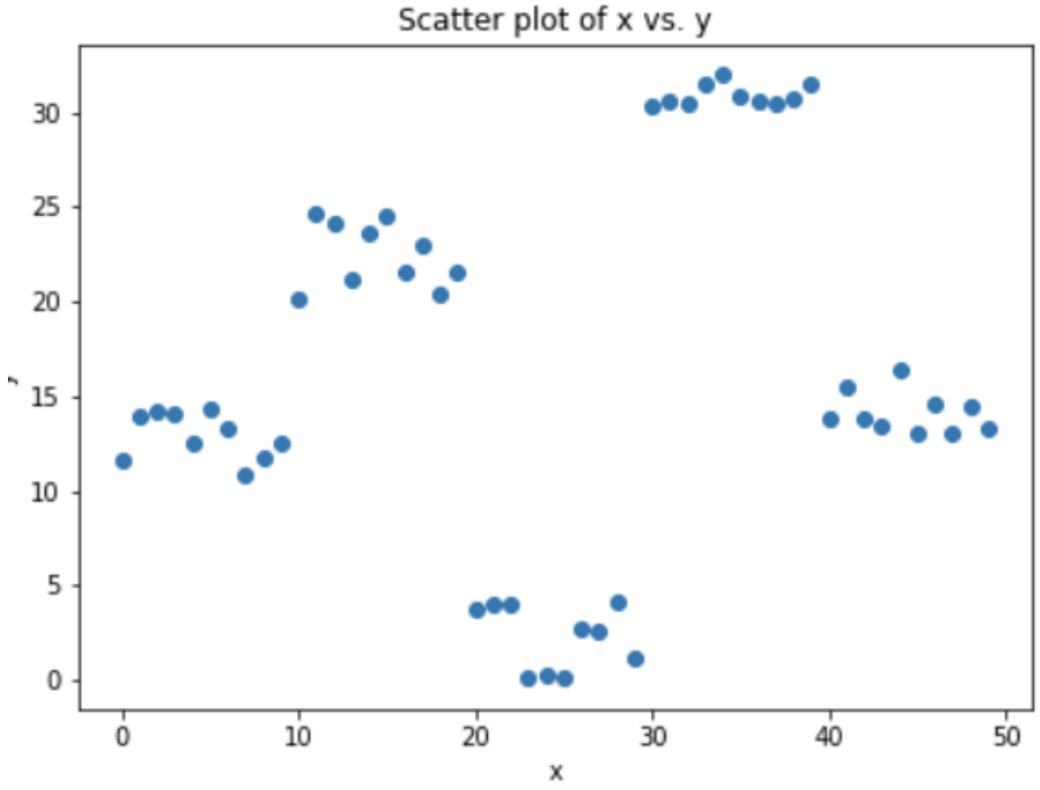
$$\frac{1}{\ell}\sum_{i=1}^{\ell}\Big(b_N(x_i) - \underbrace{\big(y_i - a_{N-1}(x_i)\big)}_{s_i^{(N)}}\Big)^2 \to \min_{b_N(x)}$$

Бустинг для MSE

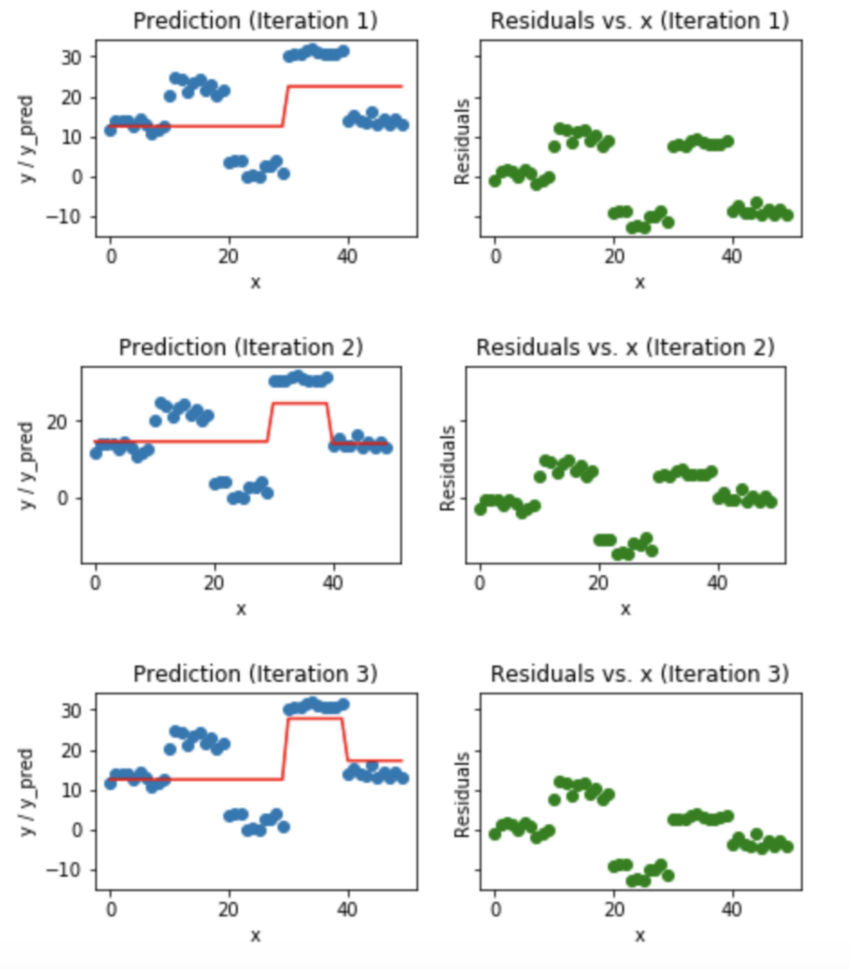
$$\frac{1}{\ell}\sum_{i=1}^{\ell}\left(b_N(x_i) - s_i^{(N)}\right)^2 \to \min_{b_N(x)}$$

- $s_i^{(N)} = y_i - a_{N-1}(x_i)$ — остатки

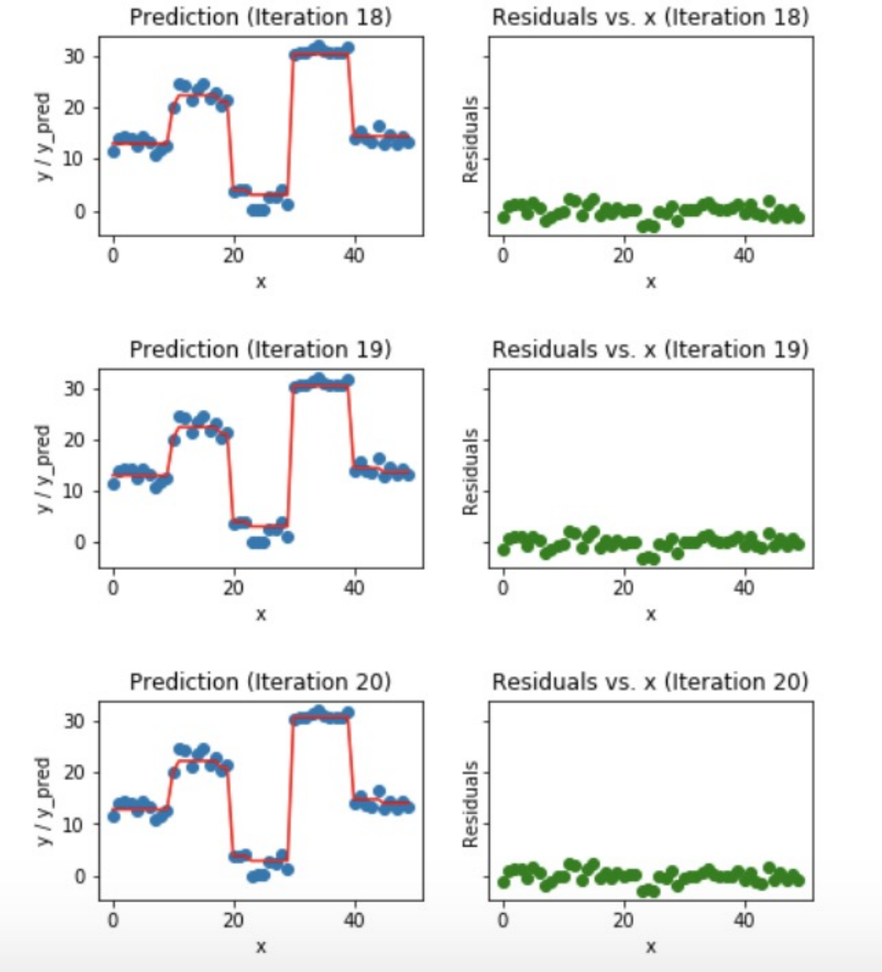
Визуализация



Визуализация



Визуализация



Градиентный бустинг в общем виде

Задача обучения базовой модели

- Обучение N-й модели:

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

- Как посчитать, куда и как сильно сдвигать $a_{N-1}(x_i)$, чтобы уменьшить ошибку?

Задача обучения базовой модели

- Обучение N-й модели:

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

- Как посчитать, куда и как сильно сдвигать $a_{N-1}(x_i)$, чтобы уменьшить ошибку?

- Посчитать производную

Задача обучения базовой модели

- Обучение N-й модели:

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

- Посчитаем производную:

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

Задача обучения базовой модели

- Посчитаем производную:

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

- Знак показывает, в какую сторону сдвигать прогноз на x_i, чтобы уменьшить ошибку композиции на нём

- Величина показывает, как сильно можно уменьшить ошибку, если сдвинуть прогноз

- Если ошибка почти не сдвинется, то нет смысла что-то менять

Градиентный бустинг

- Обучение N-й модели:

$$\frac{1}{\ell}\sum_{i=1}^{\ell}\left(b_N(x_i) - s_i^{(N)}\right)^2 \to \min_{b_N(x)}$$

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

Градиентный бустинг

- Обучение N-й модели:

$$\frac{1}{\ell}\sum_{i=1}^{\ell}\left(b_N(x_i) - s_i^{(N)}\right)^2 \to \min_{b_N(x)}$$

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

- Как бы градиентный спуск в пространстве ответов на обучающей выборке
- Базовая модель будет делать корректировки на объектах так, чтобы как можно сильнее уменьшить ошибку композиции
- Сдвиги учитывают особенности функции потерь

Градиентный бустинг для MSE

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

$$= -(a_{N-1}(x_i) - y_i) = y_i - a_{N-1}(x_i)$$

Градиентный бустинг для MSE

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

$$= -(a_{N-1}(x_i) - y_i) = y_i - a_{N-1}(x_i)$$

$$\frac{1}{\ell}\sum_{i=1}^{\ell}\Big(b_N(x_i) - \big(y_i - a_{N-1}(x_i)\big)\Big)^2 \to \min_{b_N(x)}$$

Градиентный бустинг для MSE

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

- $y_i = 10, a_{N-1}(x_i) = 5$: $s_i = 5$
- $y_i = 10, a_{N-1}(x_i) = 15$: $s_i = -5$

Градиентный бустинг для асимметричной функции

$$L(y,z) = \frac{1}{2}\left([z < y](z-y)^2 + 5[z \ge y](z-y)^2\right)$$

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

$$= [z < y](y-z) + 5[z \ge y](y-z)$$

Градиентный бустинг для асимметричной функции

$$s_i^{(N)} = [z < y](y - z) + 5[z \geq y](y - z)$$

- $y_i = 10, a_{N-1}(x_i) = 5$: $s_i = 5$
- $y_i = 10, a_{N-1}(x_i) = 15$: $s_i = -25$

Градиентный бустинг для логистической функции потерь

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

$$= -\frac{\partial}{\partial z} \log(1 + \exp(-y_i z))\Big|_{z=a_{N-1}(x_i)} =$$

$$= \frac{y_i}{1 + \exp\big(y_i a_{N-1}(x_i)\big)}$$

Градиентный бустинг для логистической функции потерь

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

$$= -\frac{\partial}{\partial z} \log(1 + \exp(-y_i z))\Big|_{z=a_{N-1}(x_i)} =$$

$$= \frac{y_i}{1 + \exp\big(y_i a_{N-1}(x_i)\big)}$$

$$\frac{1}{\ell} \sum_{i=1}^{\ell} \left(b_N(x_i) - \frac{y_i}{1 + \exp\big(y_i a_{N-1}(x_i)\big)} \right)^2 \to \min_{b_N(x)}$$

Градиентный бустинг для логистической функции потерь

$$\frac{1}{\ell}\sum_{i=1}^{\ell}\left(b_N(x_i) - \frac{y_i}{1+\exp\big(y_i a_{N-1}(x_i)\big)}\right)^2 \to \min_{b_N(x)}$$

- Отступ большой положительный: $\frac{y_i}{1+\exp\left(y_i a_{N-1}(x_i)\right)} \approx 0$
- Отступ большой отрицательный: $\frac{y_i}{1+\exp\left(y_i a_{N-1}(x_i)\right)} \approx \pm 1$

Градиентный бустинг для логистической функции потерь

$$\frac{1}{\ell}\sum_{i=1}^{\ell}\left(b_N(x_i) - \frac{y_i}{1+\exp\big(y_i a_{N-1}(x_i)\big)}\right)^2 \to \min_{b_N(x)}$$

- $y_i = +1, a_{N-1}(x_i) = -0.7$: $s_i = 0.67$
- $y_i = +1, a_{N-1}(x_i) = 2$: $s_i = 0.12$

Резюме

- Чтобы учесть особенности функции потерь, можно посчитать её производные в точке текущего прогноза композиции
- Базовую модель будем обучать на эти производные (со знаком минус)

Гиперпараметры и регуляризация в бустинге

Градиентный бустинг

$$a_N(x) = a_{N-1}(x_i) + b_N(x_i)$$

- Обучение N-й модели:

$$\frac{1}{\ell}\sum_{i=1}^{\ell}\left(b_N(x_i) - s_i^{(N)}\right)^2 \to \min_{b_N(x)}$$

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

Глубина деревьев

- Градиентный бустинг уменьшает смещение базовых моделей
- Разброс может увеличиться
- Поэтому в качестве базовых моделей стоит брать неглубокие деревья

Гиперпараметры

- Глубина базовых деревьев
- Число деревьев N

Проблемы бустинга

- Сдвиги показывают направление, в котором надо сдвинуть композицию на всех объектах обучающей выборки
- Базовые модели, как правило, очень простые
- Могут не справиться с приближением этого направления

Проблемы бустинга

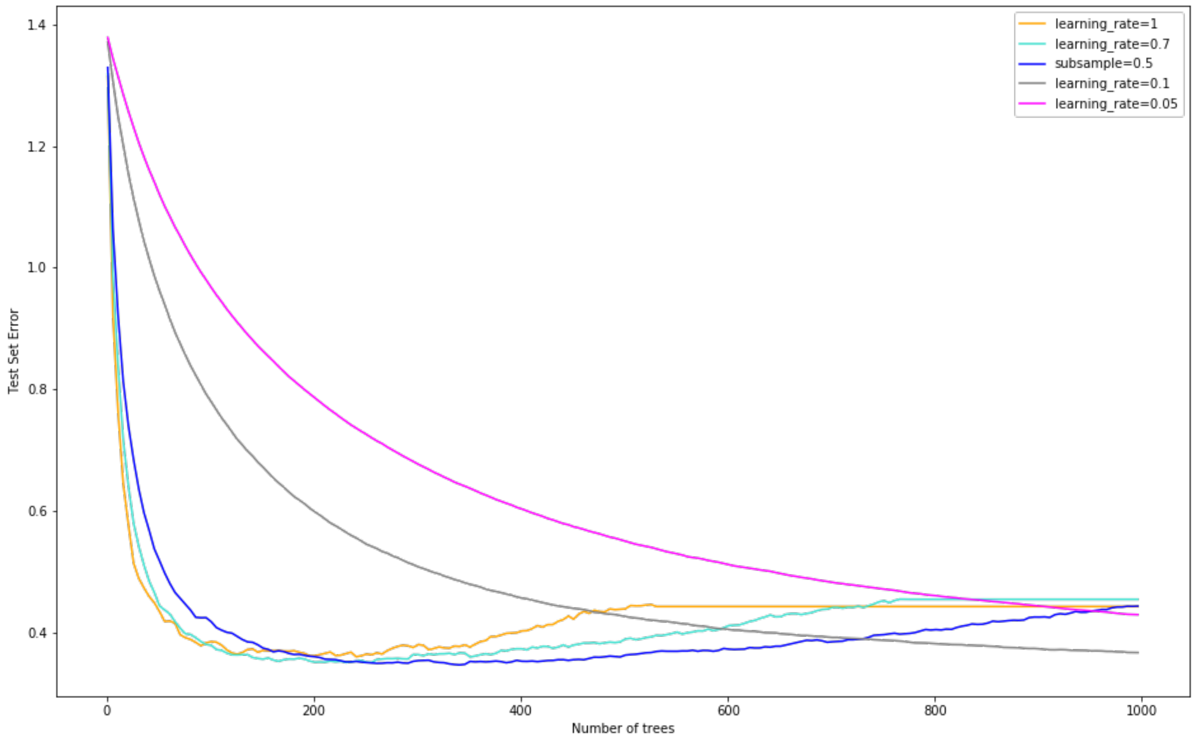
- Сдвиги показывают направление, в котором надо сдвинуть композицию на всех объектах обучающей выборки
- Базовые модели, как правило, очень простые
- Могут не справиться с приближением этого направления
- Выход: добавлять деревья в композицию с небольшим весом

Длина шага

$$a_N(x) = a_{N-1}(x_i) + \eta b_N(x_i)$$

- $\eta \in (0, 1]$ — длина шага
- Можно сказать, что это регуляризация композиции
- Снижает вклад каждой модели в композицию
- Чем меньше η, тем больше надо деревьев

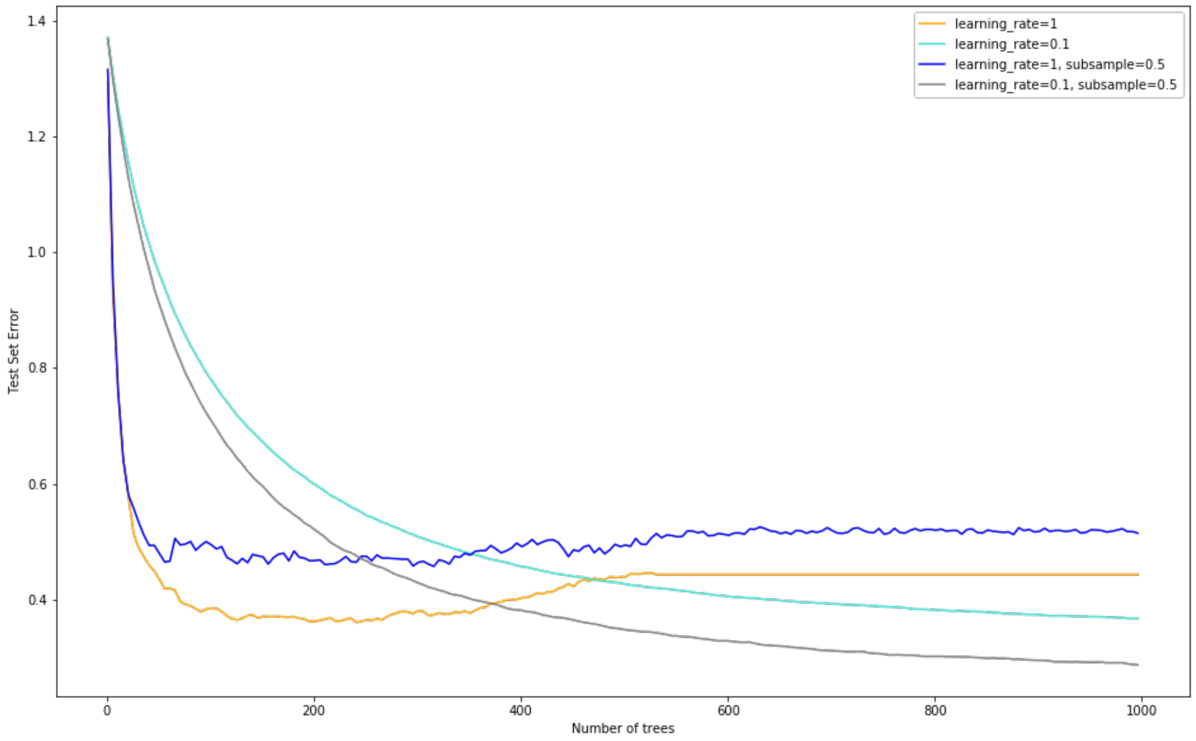
Длина шага


learning_rate=1
learning_rate=0.7
subsample=0.5
learning_rate=0.1
learning_rate=0.05
Test Set Error
1.4
1.2
1.0
0.8
0.6
0.4
0
200
400
600
800
1000
Number of trees

Рандомизация

- Можно обучать деревья на случайных подмножествах признаков
- Бустинг уменьшает смещение, поэтому итоговая композиция всё равно получится качественной
- Может снизить переобучение
- Можно обучать деревья на подмножествах объектов — способ борьбы с шумом в данных

Рандомизация



Гиперпараметры

- Глубина базовых деревьев
- Число деревьев N
- Длина шага
- Размер подвыборки для обучения
- и т.д.

Резюме

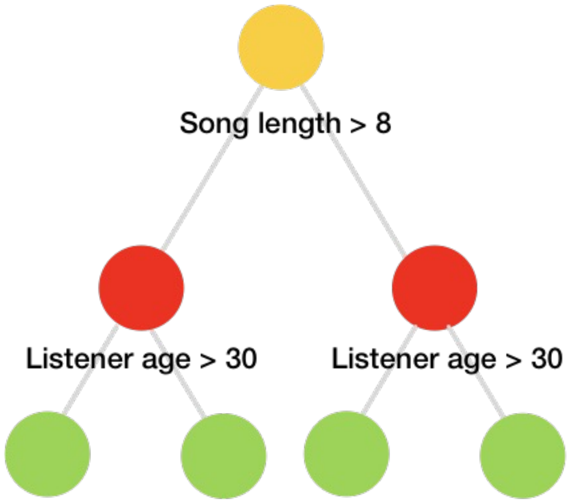
- Чтобы снизить переобучение, можно добавлять модели в композицию с небольшими весами

- Также может помочь обучение моделей на подвыборках

Вариации бустинга

ODT

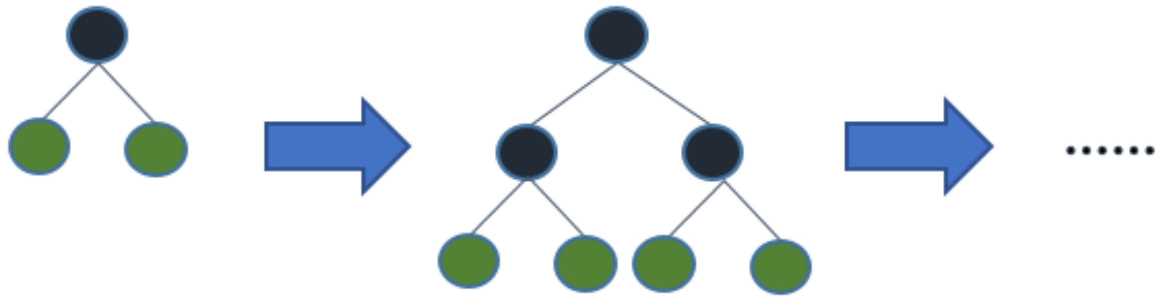
- Oblivious decision trees
- Ограничение: на одном уровне дерева используется один и тот же предикат



https://catboost.ai/

Способ построения дерева

- Level-wise: дерево строится рекурсивно до тех пор, пока не достигнута максимальная глубина

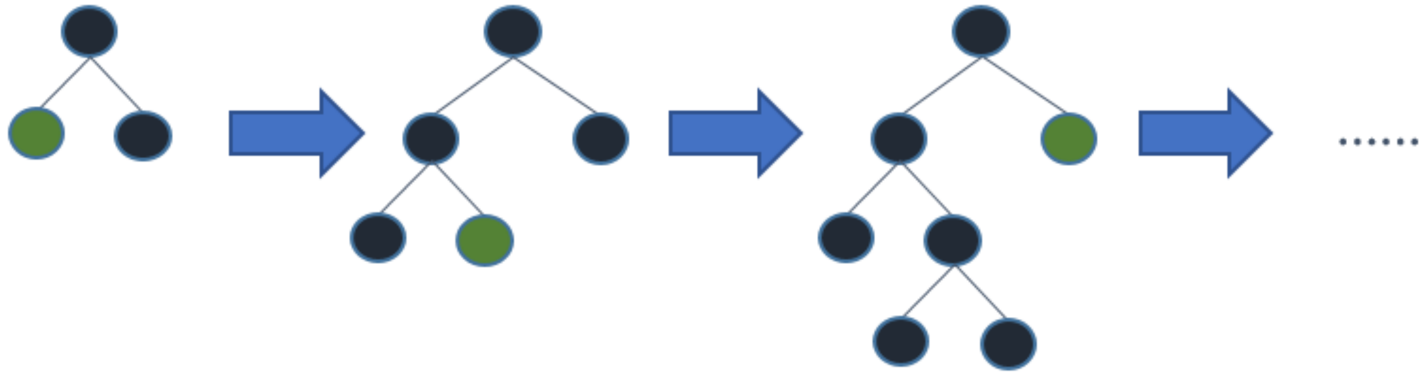


Level-wise tree growth

https://lightgbm.readthedocs.io/

Способ построения дерева

- Level-wise: дерево строится рекурсивно до тех пор, пока не достигнута максимальная глубина
- Leaf-wise: среди текущих листьев выбирается тот, чьё разбиение сильнее всего уменьшает ошибку



Leaf-wise tree growth

https://lightgbm.readthedocs.io/

Выбор лучшего порога для предиката

- $[x_j < t]$ — как выбрать t?
- Вариант 1: перебрать все известные значения признака
- Вариант 2: построить гистограмму для признака и искать пороги среди границ на гистограмме
- Вариант 3: просемплировать объекты с близкими к нулю значениями производной

Регуляризация деревьев

- Базовая регуляризация: введение длины шага и семплирования признаков
- Штрафы за число листьев в дереве
- Штрафы за величину прогнозов в листьях дерева

Улучшенное обучение

- Мы обучаем деревья на сдвиги, ошибка измеряется с помощью MSE
- Когда дерево построено, можно подобрать оптимальные значения в листьях с точки зрения исходной функции потерь

Имплементации

- XGBoost
- LightGBM: leaf-wise growth, поиск порогов на основе производных
- CatBoost: ODT