Передовые инженерные школы

МИНОБРНАУКИ РОССИИ

УНИВЕРСИТЕТ ИННОПОЛИС

онлайн университет

Программная инженерия. Разработка ПО (Python для продвинутых специалистов. Машинное обучение)

Модуль: Предобработка данных и машинное обучение

Лекция 10: Градиентный бустинг (XGB, LightGBM, Catboost)

Дата: 20.06.2025

План

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА
УНИВЕРСИТЕТА ИННОПОЛИС

Что мы уже изучили:

- как готовить данные для моделей машинного обучения
- линейная регрессия (для задачи регрессии) и логистическая регрессия (для задачи классификации)
- алгоритм машинного обучения: дерево решений
- ансамбли моделей

Что планируем сегодня изучить:

- реализации XGBoost, LightGBM, CATBoost

Градиентный бустинг

На каждой итерации градиентного бустинга вычисляется вектор сдвигов s, который показывает, как нужно скорректировать ответы композиции на обучающей выборке, чтобы как можно сильнее уменьшить ошибку

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

L() - функция потерь

a_{N-1} - композиция алгоритмов

Градиентный бустинг

После этого новый базовый алгоритм обучается путем минимизации среднеквадратичного отклонения(MSE) от вектора сдвигов s:

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

XGBoost (Extreme Gradient Boosting)

ПЕРЕДОВАЯ
ИНЖЕНЕРНАЯ ШКОЛА
УНИВЕРСИТЕТА ИННОПОЛИС

В XGBoost обучение происходит с добавлением второй производной. Вторая производная (гессиан) используется для более точного определения структуры дерева и оптимального обновления весов в листьях.

XGBoost (Extreme Gradient Boosting)

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА УНИВЕРСИТЕТА ИННОПОЛИС

Первая производная (градиент) показывает направление, в котором нужно двигаться, чтобы уменьшить ошибку.

Вторая производная (гессиан) показывает скорость изменения градиента — то есть насколько "круто" меняется функция потерь.

Аналогия:

Градиент — это скорость автомобиля (насколько сильно нужно нажать на газ/тормоз).

Гессиан — это ускорение (как быстро меняется скорость при нажатии на педаль).

XGBoost (Extreme Gradient Boosting)

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА УНИВЕРСИТЕТА ИННОПОЛИС

- MSE (регрессия), градиент = $\hat{y}$ - y, вторая производная 1
- LogLoss (классификация), градиент = p - y, вторая производная p(1 - p) (зависит от уверенности модели)

- Если гессиан большой (например, в LogLoss при $p \approx 0.5$), XGBoost делает меньший шаг, чтобы не "перепрыгнуть" минимум.
- Если гессиан маленький ($p \approx 0$ или $p \approx 1$), шаг обновления будет больше.

XGBoost. Регуляризация

ПЕРЕДОВАЯ
ИНЖЕНЕРНАЯ ШКОЛА
УНИВЕРСИТЕТА ИННОПОЛИС

Регуляризация – добавляются штрафы за

- **количество листьев**
- **за норму коэффициентов в узле**

То есть следит, чтобы не появились мусорные признаки и дерево не стало сложным

XGBoost. Регуляризация

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА УНИВЕРСИТЕТА ИННОПОЛИС

1. **Добавляет штраф за большие веса**
   - Влияет на расчёт весов листьев:

$$w_j = -\frac{\sum g_i}{\sum h_i + \lambda}$$

     где:
     - g_i — градиенты (первые производные функции потерь),
     - h_i — гессианы (вторые производные).
   - Чем больше λ, тем сильнее "сжимаются" веса w_j (аналог Ridge-регуляризации).
2. **Уменьшает переобучение**
   - Снижает влияние отдельных признаков, делая модель более гладкой и устойчивой к шуму.
3. **Влияет на критерий информативности (Gain)**
   - При выборе разбиения в дереве учитывается штраф за сложность:

$$\text{Gain} = \frac{1}{2}\left[\frac{(\sum g_{\text{left}})^2}{\sum h_{\text{left}} + \lambda} + \frac{(\sum g_{\text{right}})^2}{\sum h_{\text{right}} + \lambda} - \frac{(\sum g_{\text{parent}})^2}{\sum h_{\text{parent}} + \lambda}\right] - \gamma$$

   - Чем выше λ, тем консервативнее дерево (меньше разбиений).

XGBoost. Регуляризация

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА УНИВЕРСИТЕТА ИННОПОЛИС

При построении дерева используется критерий информативности, зависящий от оптимального вектора сдвига. В XGBoost это Gain, который вычисляется так:

$$Gain = \frac{1}{2}\left[\frac{(\sum g_L)^2}{\sum h_L + \lambda} + \frac{(\sum g_R)^2}{\sum h_R + \lambda} - \frac{(\sum g)^2}{\sum h + \lambda}\right] - \gamma$$

Как это работает?

1. Дерево перебирает все возможные разбиения и выбирает то, которое **максимизирует** `Gain`.
2. Чем больше `(∑g)² / (∑h + λ)`, тем лучше разбиение (оно сильнее уменьшает ошибку).
3. `γ` — штраф за добавление нового узла (если `Gain < γ`, разбиение не делается).

XGBoost. Пропуски

ПЕРЕДОВАЯ
ИНЖЕНЕРНАЯ ШКОЛА
УНИВЕРСИТЕТА ИННОПОЛИС

На каждом шаге разбиения (split) XGBoost рассматривает пропуски как отдельное направление.

Алгоритм сравнивает два варианта:

- Отправить все пропуски в левую дочернюю вершину.

- Отправить все пропуски в правую дочернюю вершину.

Выбирается тот вариант, который даёт наибольший прирост качества (gain) по метрике (например, коэффициенту Джини или MSE).

XGBoost. Параллелизация при построении деревьев

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА УНИВЕРСИТЕТА ИННОПОЛИС

Основная вычислительная нагрузка в XGBoost — перебор всех возможных разбиений (splits) для каждого узла дерева.
XGBoost распараллеливает перебор по столбцам (признакам):
Каждый CPU-поток обрабатывает свой набор признаков → ускорение в N раз (где N — число ядер).

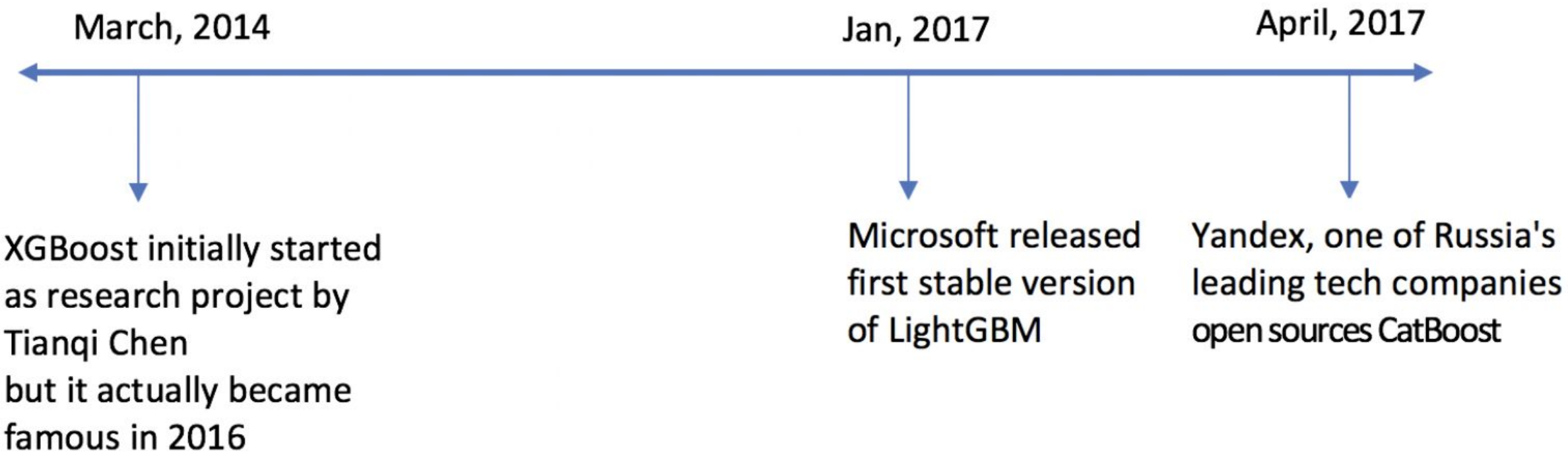
Пример

Допустим, у нас 4 признака и 4 CPU-ядра:

- Поток 1: перебирает разбиения для Возраста
- Поток 2: перебирает разбиения для Зарплаты
- Поток 3: перебирает Стаж
- Поток 4: перебирает Кредитная история

XGBoost (Extreme Gradient Boosting)

ПЕРЕДОВАЯ
ИНЖЕНЕРНАЯ ШКОЛА
УНИВЕРСИТЕТА ИННОПОЛИС



LightGBM

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА УНИВЕРСИТЕТА ИННОПОЛИС

LightGBM был разработан инженерами машинного обучения компании Microsoft. Первая версия вышла в апреле 2017.

Особенности LightGBM:

- повершинный способ построения деревьев
- односторонний отбор на основе градиентов (GOSS – Gradient-based One-Side Sampling)
- поиск точек расщепления на основе гистограмм
- связывание взаимоисключающих признаков (EFB – Exclusive Feature Bundling)

CatBoost

ПЕРЕДОВАЯ
ИНЖЕНЕРНАЯ ШКОЛА
УНИВЕРСИТЕТА ИННОПОЛИС

в 2017 году Яндекс выпустила новую библиотеку градиентного бустинга CatBoost, которая обаладает следующими особенностями:

- динамический бустинг
- новый подход к обработке категориальных признаков

Бустинг

ПЕРЕДОВАЯ
ИНЖЕНЕРНАЯ ШКОЛА
УНИВЕРСИТЕТА ИННОПОЛИС

Практика

Ссылки

ПЕРЕДОВАЯ ИНЖЕНЕРНАЯ ШКОЛА УНИВЕРСИТЕТА ИННОПОЛИС

1. Ансамбли в машинном обучении. URL: https://education.yandex.ru/handbook/ml/article/ansambli-v-mashinnom-obuchenii
2. Random Forest, метод главных компонент и оптимизация гиперпараметров: пример решения задачи классификации на Python. URL: https://habr.com/ru/companies/ruvds/articles/488342/
3. Объединение моделей для методов ансамблевого обучения. URL: https://github.com/xsolare/neyronki/blob/master/README.md#25
4. Открытый курс машинного обучения. Тема 5. Композиции: бэггинг, случайный лес. URL: https://habr.com/ru/companies/ods/articles/324402/
5. Градиентный бустинг. URL: https://education.yandex.ru/handbook/ml/article/gradientnyj-busting
6. ДЕРЕВЬЯ И БУСТИНГ. URL: https://logic.pdmi.ras.ru/~sergey/teaching/mlspsu22/16-boosting.pdf

Передовые инженерные школы

МИНОБРНАУКИ РОССИИ

УНИВЕРСИТЕТ ИННОПОЛИС

онлайн университет

Спасибо за внимание