Практика 6

Ансамблевые методы машинного обучения ч.2

Основная идея

- строятся базовые модели, отличающиеся друг от друга
- агрегация прогнозов в ансамбль с применением мета-алгоритма

Итог: борьба как с недообучением, так и переобучением

Адаптивный бустинг

- Модель – взвешенный ансамбль базовых моделей
- В процессе построения ансамбля замеряются и учитываются две характеристики:
  - Сложность объекта (зависит от ошибки ансамбля на объекте)
  - Вес базовой модели (насколько он лучше других)
- Модели строятся последовательно, на каждом шаге находим такую модель и такой вес, которые минимизируют функцию потерь
- Итоговый ансамбль применяет взвешенное голосование

AdaBoost для бинарной классификации

Пусть есть выборка примеров, каждый имеет свой неотрицательный вес с их общей суммой 1.

$a_0(x) = 0, w_i = \frac{1}{l}$. Далее цикл из m итераций:

1) Находим классификатор, который минимизирует взвешенную ошибку классификации:

$$err_w\big(b_m(x)\big) < 0.5; b_m = argmin \sum_{i=1}^{m} w_i \, [y_i \neq a(x_i)]$$

2) Добавляем его в ансамбль с новым весом $\frac{1}{2} ln \frac{1-err_w}{err_w}$.
3) Считаем ошибки всех примеров.
4) Перевзвешиваем и нормируем веса важности примеров. Используется экспоненциальная ф-я потерь $\mathrm{L}(y, f(x)) = \exp(-yf(x))$.

Адаптивный бустинг

Достоинства:

- Полностью теоретически обоснован
- Большой выбор слабых классификаторов

Недостатки:

- Чувствителен к выбросам
- Легко переобучить, требуются большие выборки
- Плохо распараллеливается
- Не интерпретируем

Адаптивный бустинг

Параметры:

- Размер ансамбля
- Базовый классификатор (обычно проще -> лучше)
- Learning rate

Градиентный метод

- Идея: для поиска минимума целевой функции используется итерационная процедура, определяющая приближенное решение
- Выбирается начальное приближение, затем на каждой итерации определяется направление минимизации (противоположно градиенту) и длина шага
  - Константная длина шага
  - Адаптивный шаг
- Критерий остановки – близость нормы градиента к 0, малое изменение целевой функции, число шагов

Метод Ньютона

- Раскладываем функцию в ряд тейлора
- Минимизируем квадратичную часть

- Нет проблемы выбора шага
- Быстро сходится
- Ищет локальный минимум
- Важен выбор начального приближения
- Вычислительная трудоемкость

Градиентный бустинг

- Следующий отклик = предыдущий – результат функции потерь с коэффициентом
- Чтобы найти градиент, можем обучить модель на векторе откликов (антиградиент функции потерь с предыдущего шага, «псевдоостатки»)
- Добавляем эту модель в ансамбль, продолжаем цикл
- Таким образом, каждая следующая модель обучается на ошибках предыдущей

Регуляризация

- Ранняя остановка
- Стохастический градиентный бустинг (случайные подвыборки меньшего размера для обучения базовых моделей на каждой итерации)
- Ограничение размера моделей или всего ансамбля
- Learning rate
- Регуляризация L1, L2

Современные алгоритмы

- XGBoost (2016)
  - Оптимизирован под рост в ширину
  - Ньютоновский бустинг – критерий поиска разбиений, учитывающий качество и регуляризацию всего ансамбля
- LightGBM (2017)
  - Оптимизирован под рост дерева в глубину
  - Есть жадный алгоритм для группировки категориальных признаков в непересекающиеся группы
- Catboost (2017)
  - Кодирует категориальные признаки
  - Упорядоченный семплинг

- Ускорение вычислений (гистограммный метод), распараллеливание, поддержка GPU
- Применение бэггинга
- Пользовательские метрики и функции потерь