Машинное обучение ФИВТ

DS-поток

Лекция 4

Бэггинг, ансамбли моделей и случайный лес

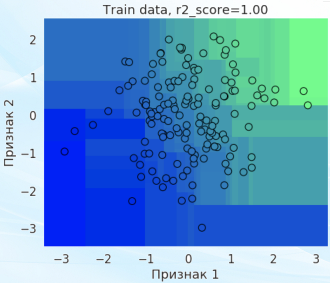
Основные свойства решающих деревьев

Плюсы

- Восстанавливают сложные закономерности

Минусы

- Очень легко переобучаются.
  Неустойчивы к малейшим изменениям в данных.
- Восстанавливаемая зависимость довольно ужасна.



$\Rightarrow$ Сами деревья не очень хороши.

Идея

Один в поле не воин...

Идея

Лес - много деревьев

Идея

А есть ли смысл брать деревья одинаковыми?

Нужны разные деревья

"Танцующий лес", нац. парк Куршская коса, Калининградская обл.

Идея

Возьмем композицию вида:

$$f = \frac{1}{T}\sum_{t=1}^{T} b_t$$

где b_t — решающее дерево.

Чтобы сделать деревья b_t разными:

- b_t обучаем на некоторой подвыборке.
- b_t обучаем на случайном подпространстве признаков.

Ансамбли моделей и случайные леса

Bias-variance tradeoff

Беггинг

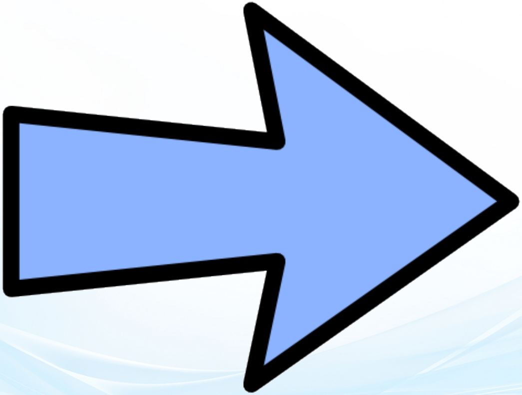
Случайный лес

Out-of-Bag ошибка

Связь с метрическими моделями

Важность признаков

Bias-variance tradeoff

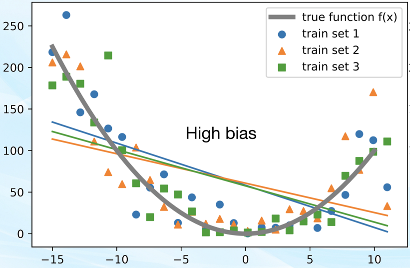


Bias-variance tradeoff

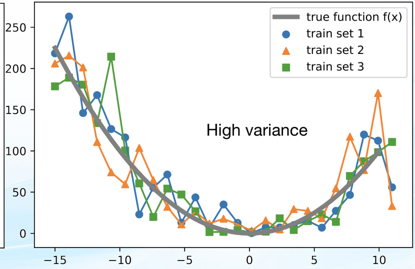
Шум = шум в данных.

Смещение (bias) = среднее отклонение модели от истинной зависимости.

Разброс (variance) = среднеквадратичный разброс ответов обученных моделей относительно среднего ответа.
Показывает насколько сильно может измениться предсказание обученной модели в зависимости от выборки.



Большое смещение, маленький разброс



Маленькое смещение, большой разброс

Bias-variance tradeoff

Общий случай

Есть более общие формулы этого разложения для других функций, состоящие из трех компонент с похожим смыслом.

Т.е. для многих распространненых функций потерь ошибка метода обучения может быть разложена на шум, смещение и дисперсию.

Подробнее для общий вид разложения можно прочитать тут: *Domingos, Pedro (2000). A Unified Bias-Variance Decomposition and its Applications*

Ансамбли моделей и случайные леса

Беггинг

Bagging = Boostrap Aggregating

Пусть есть выборка (X, Y).

Сгенерируем T бутстрепных подвыборок из нее.

На каждой из них обучим отдельную модель $\hat{y}_t = \mu_t(X_t^*, Y_t^*)$.
$\mu_t : (\mathscr{X} \times \mathscr{Y})^n \to \mathscr{F}$ - метод обучения, который сопоставляет выборке некоторую модель из семейства $\mathscr{F}$.

Итоговая модель строится как композиция:

$$\hat{y} = \frac{1}{T}\sum_{t=1}^{T}\hat{y}_t = \frac{1}{T}\sum_{t=1}^{T}\mu_t(X_t^*, Y_t^*)$$

Модели $\hat{y}_t$ не обязаны быть моделями из одного вида моделей
Например, $\hat{y}_1$ может быть линейной моделью, а $\hat{y}_2$ - деревом.

Беггинг

Рассмотрим случай, когда $\mu_1 = \mu_2 = ... = \mu_n = \mu$.

Т.е. рассматриваем одну и ту же модель обучения.

Тогда компоненты композиции одинакого распределены.

Смещение из bias-variance разложения для композиции:

$$\mathrm{E}\Big(\mathrm{E}\big(\hat{y}(X) - f(X,\epsilon)|X\big)\Big)^2 = \mathrm{E}\Big(\mathrm{E}\big(\hat{y}_1(X) - f(X,\epsilon)|X\big)\Big)^2$$

Разброс из bias-variance разложения для композиции:

$$\mathrm{E}\Big(\mathrm{D}\big(\hat{y}(X)|X\big)\Big) = \frac{1}{T}\mathrm{E}\Big(\mathrm{D}\big(\hat{y}_1(X)|X\big)\Big) + \frac{T-1}{T}\mathrm{E}\; cov\big(\hat{y}_1(X), \hat{y}_2(X)|X\big)$$

Беггинг: вывод

Если базовые модели

- слабо коррелированы
- имеют низкое смещение
- имеют высокий разброс

то беггинг-композиция имеет низкое смещение и низкий разброс.

Когда модели менее коррелированы?

Когда они достаточно разные.

Как сделать модели разными?

- Использовать разные виды моделей и разные гиперпараметры.
- Обучать модели на разных признаках.
- Делать разную предобработку данных.

Ансамбли моделей и случайные леса

Bias-variance tradeoff

Беггинг

Случайный лес

Out-of-Bag ошибка

Связь с метрическими моделями

Важность признаков

Случайный лес

Возьмем в качестве базовых моделей решающие деревья.

Свойства решающего дерева с большой глубиной:

- bias - низкий
- variance - высокий

Деревья разные $\Rightarrow$ при объединении получим хорошую композицию.

Напоминание: Деревья могут быть сильно разными даже при небольшом изменении выборки.

Как сделать деревья разными?

- По объектам: Каждое дерево обучается на бутстрепной выборке.
- По признакам: Деревья в лесу являются *рандомизированными*.

Случайный лес

В обычном дереве:

В каждой вершине разбиение ищется по всем признакам.

В рандомизированном дереве:

В каждой вершине разбиение ищется по случайному подмножеству признаков некоторого размера d_0.
Множество признаков выбирается для каждой вершины заново!

Поиск разбиения в вершине в рандомизированном дереве:

1. Выбрать случайное подмножество признаков размера d_0 из всех признаков.
2. Перебрать все признаки из d_0 с соответствующими порогами, посчитать критерий ошибки.
3. Выбрать оптимальный признак из d_0 выбранных и его порог.

Случайный лес

Пусть d - количество признаков.

Рекомендации:

- В задаче классификации

  Взять $d_0 = \left\lfloor \sqrt{d} \right\rfloor$.

  Строить каждое дерево до тех пор,

  пока в каждом листе не окажется по 1 объекту.

- В задаче регрессии

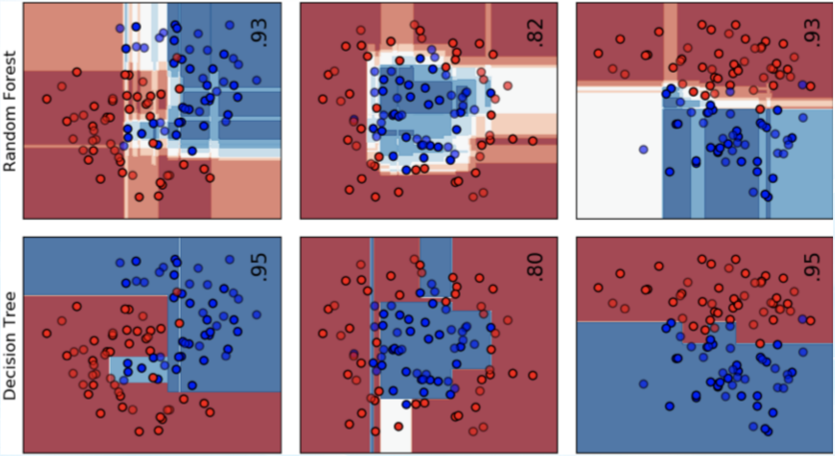
  Взять $d_0 = \left\lfloor d/3 \right\rfloor$.

  Строить каждое дерево до тех пор,

  пока в каждом листе не окажется по 5 объектов.

Случайный лес

Сравнение с решающим деревом



Ансамбли моделей и случайные леса

Bias-variance tradeoff

Беггинг

Случайный лес

Out-of-Bag ошибка

Связь с метрическими моделями

Важность признаков

Out-of-Bag ошибка

Каждое дерево в лесе обучается по подмножеству объектов.

$\Rightarrow$ Объекты, не вошедшие в бутстрепную выборку (X_t^*, Y_t^*) для дерева $\hat{y}_t$, являются валидационными для данного дерева.

$\Rightarrow$ Можем для каждого объекта (x_i, Y_i) найти деревья, которые были обучены без него.

Напоминание из статистики:

$P(\text{объект попадает в бутстрепную выборку}) = 1 - e^{-1}$

Т.е. в среднем около трети объектов не попадут в бутстрепную выборку.

Out-of-Bag ошибка

Вычислим по их ответам out-of-bag-ошибку:

$$OOB = \sum_{i=1}^{n} \mathcal{L}\left(Y_i, \frac{1}{\sum_{t=1}^{T} I\{x_i \notin X_t^*\}} \sum_{t=1}^{T} I\{x_i \notin X_t^*\}\, \hat{y}_t(x_i)\right)$$

где $\mathcal{L}(y, z)$ - функция потерь.

Смысл: Усредняем ответы только деревьев, не обученных на (x_i, Y_i).

По мере увеличения числа деревьев T ошибка OOB стремится к leave-one-out оценке, но сильно проще для вычисления.

Ансамбли моделей и случайные леса

Bias-variance tradeoff

Беггинг

Случайный лес

Out-of-Bag ошибка

Связь с метрическими моделями

Важность признаков

Связь с метрическими моделями

Утверждение:

Случайные леса осуществляют предсказание для объекта на основе похожих объектов из обучения.
Схожесть объектов тем выше, чем чаще эти объекты попадают в один лист дерева в лесу.

Рассмотрим задачу регрессии с функцией потерь MSE.

Пусть $L_k(x)$ — лист k-го дерева, в который попал объект x.

Ответ k-го дерева на объекте x — средний ответ по всем обучающим объектам, попавшим в лист $L_k(x)$:

$$\hat{y}_k(x) = \frac{1}{|L_k(x)|} \sum_{x_i \in L_k(x)} Y_i =$$

$$= \sum_{i=1}^{n} \frac{I\{L_k(x) = L_k(x_i)\}}{\sum\limits_{j=1}^{n} I\{L_k(x) = L_k(x_j)\}} \; Y_i = \sum_{i=1}^{n} w_k(x, x_i) \; Y_i$$

Связь с метрическими моделями

$$\hat{y}_k(x) = \sum_{i=1}^{n} w_k(x, x_i)\; Y_i$$

Тогда ответ композиции равен:

$$\hat{y}(x) = \frac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{n} w_t(x, x_i) Y_i = \sum_{i=1}^{n}\left(\frac{1}{T}\sum_{t=1}^{T} w_t(x, x_i)\right)\; Y_i$$

Ответ случайного леса — сумма ответов всех объектов обучения с некоторыми весами.

Веса измеряют сходство объектов x и x_i на основе того, сколько раз они оказались в одном и том же листе.

$\Rightarrow$ Случайный лес позволяет ввести функцию расстояния на объектах.

Связь с метрическими моделями

Номер листа $L_k(x)$, в который попал объект, сам по себе является хорошим признаком.

Неплохо работает следующий подход:

- По выборке обучается случайный лес
- К выборке добавляются категориальные признаки $L_1(x), L_2(x), ..., L_T(x)$.
- На полученном обучается некоторая модель.

Новые признаки являются результатом нелинейного разбиения пространства и несут в себе информацию о сходстве объектов.

Ансамбли моделей и случайные леса

Bias-variance tradeoff

Беггинг

Случайный лес

Out-of-Bag ошибка

Связь с метрическими моделями

Важность признаков

Важность признаков

Будет рассказано позже.


ВСЁ!