

Ансамбли в машинном обучении

Суть в том, что мы собираем несколько моделей вместе, каждая из них делает своё предсказание, и на основе этой совокупности предсказаний мы делаем итоговое предсказание. Может быть ситуация, что итоговый результат будет лучше, чем предсказание лучшей из моделей.

Смещение и разброс

Мы можем ввести функция качества модели, она будет зависеть от шума, смещения и разброса(bias-variance decomposition).

Смещение (bias) - усреднённая разница между значением в истинной зависимости и предсказанной.

Разброс (variance, дисперсия) - разброс предсказаний алгоритма в зависимости от обучающей выборки.

Шум - минимально возможная ошибка.

Очевидно, что мы хотим уменьшать ошибку алгоритма, а так как с шумом мы ничего поделать не можем, то нужно уменьшать разброс или смещение.

Бэггинг (bagging, bootstrap aggregation)

Идея: возьмём нашу выборку, равновероятным образом возьмём объекты из неё и добавим во вторую выборку (переложим столько объектов, сколько их в выборке), в новой выборке какие-то объекты встретятся несколько раз, а какие-то не встретятся вообще, на второй выборке обучим какую-то модель, повторим процедуру k раз, получим k моделей, обученных на k выборок, тогда итоговым предсказанием будет являться среднее арифметическое предсказаний всех вспомогательных моделей.

$$a(x) = \frac{1}{k}(b_1(x) + \dots + b_k(x))$$

Генерация подвыборок с возвращением элементов - bootstrap.

к вспомогательных алгоритмов - базовые алгоритмы.

Наша итоговая модель - ансамбль.

С помощью математики (считанию мат. ожидания и т.д.) выяснили, что смещение композиции равно смещению алгоритма, то есть не меняется, а дисперсия получается в k раз меньше.

Поэтому всегда делают бэггинг на деревьях, например, а не просто использует одно дерево.

Random forest

Случайный лес - бэггинг на решающих деревьях. Алгоритм построения такого ансамбля:

- 1 Для построения i -го дерева:
 - а. Сначала, как в обычном бэггинге, из обучающей выборки X выбирается с возвращением случайная подвыборка X^i того же размера, что и X .
 - б. В процессе обучения каждого дерева **в каждой вершине** случайно выбираются $n < N$ признаков, где N — полное число признаков (метод случайных подпространств), и среди них ищется оптимальный сплит. Такой приём как раз позволяет управлять степенью скоррелированности базовых алгоритмов.
- 2 Чтобы получить предсказание ансамбля на тестовом объекте, усредняем отдельные ответы деревьев (для регрессии) или берём самый популярный класс (для классификации).
- 3 Profit. Мы построили **Random Forest (случайный лес)** — комбинацию бэггинга и метода случайных подпространств над решающими деревьями.

Какие параметры могут быть?

Глубина деревьев в случайном лесе

Нам нужно уменьшить или смещение или разброс. Так как бэггинг не влияет на смещение, а только уменьшает разброс, давайте постараемся уменьшить смещение. Чем более глубоко дерево, тем лучше оно запоминает выборку, тем меньше получается смещение, значит нужно использовать глубокие деревья.

Количество признаков у дерева

Чем больше признаков, тем более похожими получатся деревья между собой, тем меньше будет пользы от бэггинга. Чем меньше будет признаков, тем более слабыми будут деревья. Универсальный совет - берём корень фич для классификации и треть для регрессии.

Количество деревьев

Фактически, чем больше, тем лучше, так как мы уменьшаем разброс таким образом, но в какой-то момент прирост будет незначительным, так что лучше строить график зависимости ошибки от количества деревьев, чтобы выбрать оптимальное число. Ну и чем больше деревьев, тем дольше будет работать ансамбль.

Бустинг

Бустинг (boosting) - тоже ансамблевый метод, только теперь базовые алгоритмы будут обучаться последовательно, каждый следующий применяем алгоритм старается уменьшить ошибку после всех своих предшественников, получается, что мы уменьшаем смещение(хотя разброс может даже увеличиться). Если с помощью бустинга уменьшается смещение, то логично в качестве базовых алгоритмов рассматривать алгоритмы с высоким смещением, но низким разбросом, неглубокие деревья(2-3 уровня), например.

Самый практичный и распространённый вариант бустинга - градиентный бустинг.

Стекинг

Стекинг (stacking) - ансамблевый метод, в котором можно использовать базовые алгоритмы разных семейств, а затем финальная (как правило,

простенькая) модель (метамодель) на основе предсказаний базовых выдаёт финальный ответ.

Делим тренировочную выборку на N частей, на подвыборке, состоящей из $N - 1$ частей тренируем базовые модели, на оставшейся части тестируем (фактически, кросс-валидация), на полученных мета-факторах обучается мета-модель, которая и выдаёт результат.

Blending - если данных много, то можно разделить тестовую часть на две непересекающихся подвыборки, на первой тренировать базовые модели, а на второй обучать метамодель на основе предобученных базовых моделей.

Стекинг не влияет напрямую на разброс и смещение.