Бэггинг [M.191]

Наряду с бустингом бэггинг является одним из самых современных методов повышения эффективности моделей. Бэггинг формирует набор классификаторов, которые комбинируются путем голосования или усреднения. Термин «бэггинг» (bagging) происходит от английского словосочетания bootstrap aggregating, которое можно перевести как «улучшающее объединение».

В основе работы бэггинга лежит технология, получившая название «возмущение и комбинирование» (perturb and combine).

«Возмущение и комбинирование»

Модели, которые в процессе обучения адаптируют свое состояние в соответствии с обучающим множеством, такие как деревья решений и нейронные сети, являются неустойчивыми: даже небольшие изменения в обучающем множестве (замена или удаление одного примера) могут привести к существенным изменениям в состоянии модели — в структуре дерева решений или в распределении весов нейронной сети. Иными словами, внося даже незначительные изменения в обучающие данные, мы всегда будем получать другую модель. Но исходная и измененная модели будут функционировать примерно одинаково и со сравнимой точностью: незначительные изменения в обучающих данных не приведут к изменению основных закономерностей, которые должны быть обнаружены моделью. Сказанное можно пояснить с помощью рисунка 1.

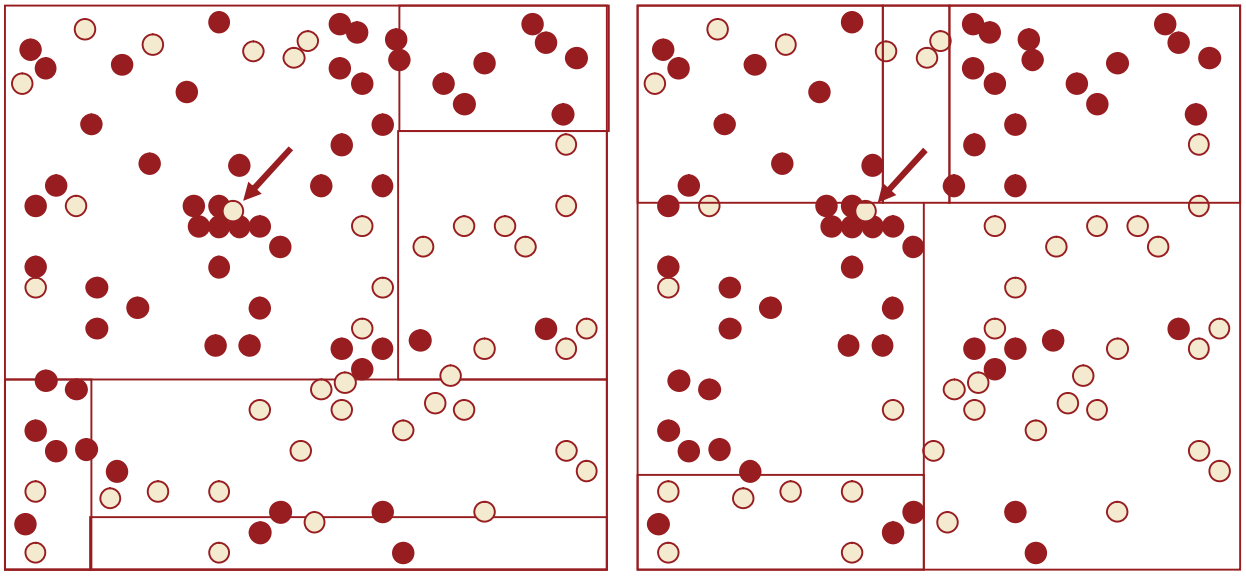

Рисунок 1 – Изменение одного примера отражается на конечной модели

Слева представлено разбиение объектов двух классов (светлых и темных точек) с помощью дерева решений. Изменение всего лишь одной точки (помеченной стрелкой) приведет к совершенно другому разбиению и другой структуре дерева решений, но в целом точность модели останется примерно на том же уровне. Неустойчивость деревьев решений во многом обусловлена *конкурирующими узлами*, то есть узлами, работающими примерно одинаково. Поэтому даже небольшое изменение в данных может привести к тому, что процесс обучения пойдет по другому узлу и будет построено другое дерево решений.

Неустойчивость моделей, особенно деревьев решений, используется для создания ансамблей моделей с помощью технологии «возмущения и комбинирования». Под *возмущением*

понимается внесение некоторых изменений, часто случайного характера, в обучающие данные и построение нескольких альтернативных моделей на измененных данных с последующим комбинированием результатов. Для возмущения используются следующие приемы:

- извлечение выборок из обучающего множества. В этом случае путем сэмплинга из исходного обучающего множества извлекается несколько выборок, и на каждой из них обучается отдельная модель;
- выборка из выборок, формирование внутри выборок подгрупп;
- добавление шума;
- адаптивное взвешивание;
- случайный выбор между конкурирующими узлами (разбиениями).

Добавление в процесс обучения элемента случайности часто называют *рандомизацией* (randomization).

Основная идея

Идея бэггинга проста. Сначала на основе исходного множества данных путем случайного отбора формируется несколько выборок. Они содержат такое же количество примеров, что и исходное множество. Но, поскольку отбор производится случайно, набор примеров в этих выборках будет различным: одни примеры могут быть отобраны по нескольку раз, а другие — ни разу. Затем на основе каждой выборки строится классификатор, и выходы всех классификаторов комбинируются (агрегируются) путем голосования или простого усреднения. Ожидается, что полученный результат будет намного точнее любой одиночной модели, построенной на основе исходного набора данных. Обобщенная схема процедуры бэггинга представлена на рисунок 2 (на примере дерева решений).

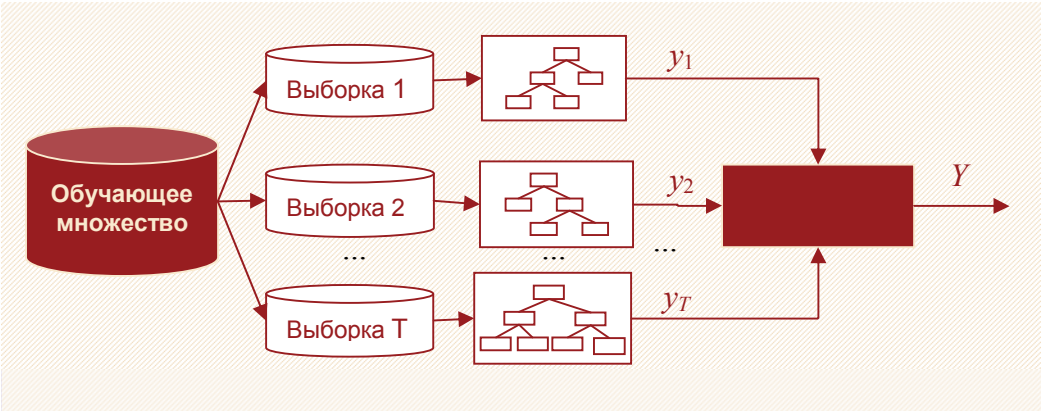

Рисунок 2 – Схема процедуры бэггинга

Таким образом, бэггинг включает следующие шаги.

1. Из обучающего множества извлекается заданное количество выборок одинакового размера.
2. На основе каждой выборки строится модель.
3. Определяется общий результат путем голосования или усреднения выходов моделей.

Предположим, что задан набор из N обучающих примеров, каждый из которых относится к одному из $k = k_1, k_2, \ldots, K$ классов, а также алгоритм обучения, строящий классификаторы. Бэггинг представляет собой итерационную процедуру, где количество итераций (испытаний) T задается как константа (хотя иногда для автоматического определения числа итераций рекомендуется использовать тестовое множество, чтобы избежать переобучения). Классификатор, полученный на итерации t, обозначим C_t, а итоговый классификатор,

полученный с помощью бэггинга, — C^*. Для каждого примера x $C_t(x)$ и C*(x) являются классами, предсказанными классификаторами C_t и C^* соответственно.

Пусть на каждой итерации $t = t_1, t_2,\ldots, T$ из исходного обучающего множества производится выборка, состоящая из N примеров. Она имеет тот же размер, что и исходное множество, но некоторые примеры могут в нее не попасть, а некоторые попадут несколько раз. Система обучения создает классификаторы C_t из выборок, а конечный классификатор C^* создается путем агрегирования T классификаторов. При классификации примера x голоса для класса k записываются каждым классификатором, для которого $C_t(x) = k$, и затем для $C^*(x)$ назначается класс, получивший большинство голосов.

Схема бэггинга, представленная на рисунке 2, применяется и к регрессионным моделям. В этом случае результат — среднее значение, рассчитанное по выходам всех моделей ансамбля.

Остановка процедуры бэггинга производится на основе следующих критериев.

- На некоторой итерации t ошибка ε_t классификатора C_t становится равной 0 или больше либо равной $0{,}5$. В этом случае процедура бэггинга останавливается, а последний классификатор удаляется: $T = t - 1$. Таким образом, бэггинг не пускает в ансамбль «плохие» классификаторы.
- Число итераций достигло заданного пользователем предела T. Как и для большинства других итеративных алгоритмов Data Mining, однозначного ответа на вопрос, каково достаточное число итераций, не существует. Оно подбирается эмпирическим путем.
- Бэггинг, хотя и в меньшей степени, чем отдельные модели, склонен к переобучению. Поэтому критерием для остановки процедуры может служить возрастание ошибки на тестовом множестве.

Почему бэггинг работает?

Результаты бэггинга часто удивляют аналитиков: точность предсказания построенных с его помощью комбинированных классификаторов оказывается значительно выше, чем точность отдельных моделей. Причины этого неочевидны. Действительно, если все выборки извлечены из одного обучающего множества, то можно ожидать, что построенные на их основе модели будут практически идентичными. Однако данное предположение ошибочно. Причина заключается все в той же неустойчивости моделей, особенно деревьев решений: даже небольшие изменения в обучающих данных легко могут привести к выбору различных атрибутов в определенном узле дерева, что вызовет существенные изменения при дальнейшем ветвлении. В результате одни и те же тестовые примеры будут распознаваться различными деревьями по-разному.

В основе эффективности методов комбинирования моделей, в том числе бэггинга, лежит идея декомпозиции ошибки ансамбля на *смещение* и *дисперсию*. Предположим, что имеется бесконечное число независимых обучающих множеств одинакового размера, которые используются для построения бесконечного числа классификаторов. Обучающие примеры обрабатываются всеми классификаторами, и их общий выход определяется большинством голосов. В данной ситуации ошибки все еще имеют место, поскольку идеальных обучающих процедур не существует. Значение ошибки зависит от того, насколько хорошо метод машинного обучения соответствует решаемой задаче, а также от качества самих данных.

Допустим, ожидаемая ошибка оценивается как средняя ошибка комбинированного классификатора на бесконечном числе независимо выбранных тестовых примеров. Значение ошибки *конкретного* обучающего алгоритма называется *смещением* (систематической ошибкой) ε_b. Смещение есть мера стабильности ошибки данного алгоритма обучения, которая не может быть исключена даже применением бесконечного числа обучающих множеств. Конечно, на практике такая ошибка не может быть вычислена точно, а только оценивается приблизительно.

Вторым источником ошибки при обучении модели является то, что обучающее множество используется частично — оно всегда конечно — и, следовательно, не полностью представляет реальную популяцию наблюдений. Ожидаемое значение этого компонента ошибки по всем возможным обучающим наборам заданного размера и всем возможным тестовым наборам называется *дисперсией* метода обучения для этой задачи ε_D.

Таким образом, *общая ожидаемая ошибка классификатора* состоит из суммы дисперсии и смещения: $\varepsilon = \varepsilon_b + \varepsilon_D$. В этом и заключается смысл декомпозиции ошибки на дисперсию и смещение. Комбинирование нескольких классификаторов позволяет уменьшить ошибку за счет дисперсии. При этом, чем больше классификаторов используется, тем меньше дисперсия.

Смещение определяется как среднеквадратическая ошибка, ожидаемая при усреднении по моделям, построенным на основе всех обучающих множеств одного размера, а дисперсия — это компонент ожидаемой ошибки отдельной модели, которая была построена с помощью отдельного обучающего множества. Теоретически можно показать, что усреднение по множеству моделей, построенных на основе независимых обучающих множеств, всегда уменьшает ожидаемое значение среднеквадратической ошибки.

На практике возникает проблема: где взять большое количество обучающих множеств. Бэггинг решает эту задачу, используя единственное обучающее множество, а именно «тасует» его, как колоду карт (рисунок 3).

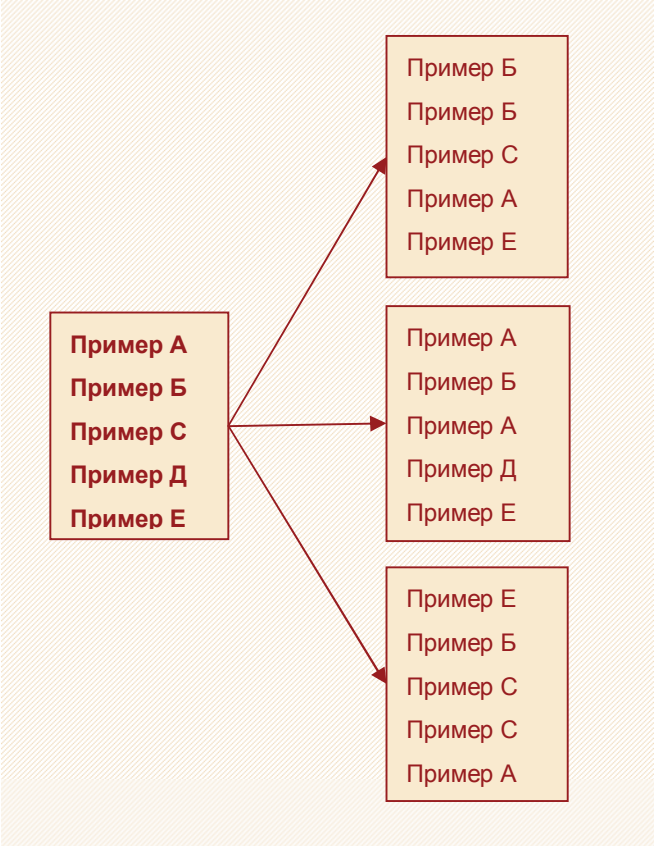

Рисунок 3 – Отбор примеров с заменой, используемый в бэггинге

Разница между бэггингом и идеальной процедурой, описанной выше, заключается в способе формирования обучающих множеств. Вместо получения независимых множеств из предметной области бэггинг просто производит перевыборку исходного множества данных. Такие множества отличаются друг от друга, но не являются независимыми, поскольку все они основаны на одном и том же множестве. Тем не менее бэггинг позволяет создавать комбинированные модели, которые, как правило, работают значительно лучше, чем отдельная модель.