УДК 004.85

НОВЫЙ МЕТОД ПОСТРОЕНИЯ ДЕРЕВЬЕВ РЕШЕНИЙ С ПРОИЗВОЛЬНЫМИ ФУНКЦИЯМИ ПОТЕРЬ

А.В. Константинов
Санкт-Петербургский политехнический университет Петра Великого
Россия, 195251, Санкт-Петербург, Политехническая ул., 29
E-mail: andrue.konst@gmail.com

Л.В. Уткин
Санкт-Петербургский политехнический университет Петра Великого
Россия, 195251, Санкт-Петербург, Политехническая ул., 29
E-mail: lev.utkin@gmail.com

Ключевые слова: машинное обучение, дерево решений, дифференцируемые функции потерь, классификация, регрессия.

Аннотация: Деревья решений часто применяются для обработки табличных данных, задач классификации и регрессии, и лежат в основе множества ансамблевых моделей машинного обучения. Классические алгоритмы построения деревьев решений опираются на эвристические методы для выбора правил расщепления узлов дерева при обучении, что не гарантирует даже локальную оптимальность разбиения, и позволяет строить модели лишь для некоторых заранее определённых функций потерь. В данной работе рассматривается новый алгоритм построения деревьев решений, в основе которого лежит метод расщепления узлов, минимизирующий переменные компоненты разложения функции потерь в ряд до второго порядка. Таким образом, алгоритм позволяет строить деревья для оптимизации произвольных дважды дифференцируемых функций потерь. Также рассмотрены два варианта регуляризации. Применение нового алгоритма продемонстрировано на примере задачи классификации с бинарной перекрёстной энтропией в качестве функции потерь.

1. Введение

Деревья решений позволяют решать различные задачи машинного обучения, в том числе задачи классификации и регрессии. В составе ансамблей, таких как случайные леса [1] и чрезвычайно рандомизированные деревья [2], или модели градиентного бустинга [3], деревья решений широко применяются для обработки табличных данных. В основе алгоритма построения деревьев решений CART [4] лежит идея жадного рекурсивного расщепления узлов построенного дерева на основе эвристического правила, зависящего от типа задачи, поскольку для каждого потенциального разбиения вычисляется значение некоторого критерия, например, для задачи классификации, индекса Джини или энтропии. Для расщепления данного узла выбирается то потенциальное разбиение, которое имеет наименьшее значение рассчитанного критерия. В листья построенного дерева записывается агрегированное значение целевых значений всех наблюдений обучающей выборки, которые в него попадают, например среднее, или наиболее часто встречаемый класс в случае классификации. Использование эвристического правила, напрямую не связанного с оптимизируемой в

ходе обучения функцией потерь, может приводить к менее подходящим структурам деревьев. Кроме того, наиболее успешно применяемые ансамблевые методы основаны на идее градиентного бустинга: итеративного построения набора деревьев, сумма предсказаний которых минимизирует дифференцируемую функцию потерь, где каждое следующее дерево строится для реализации шага градиентного спуска. Такие методы позволяют обучать ансамбли используя лишь производные функции потерь, что даёт возможность объединять такие модели, например, с нейронными сетями, и осуществлять обучение от начала до конца градиентными методами. Однако отдельные деревья решений не позволяют осуществлять обучение на основе градиента нестандартной функции потерь. В алгоритме XGBoost [3] используется разложение функции потерь в ряд для расчёта значений в листьях, однако оно производится вокруг предсказаний ранее построенного ансамбля, что не позволяет строить одно дерево, оптимизирующее функцию по градиенту.

В данной работе рассматривается новый алгоритм построения деревьев решений для минимизации произвольных дважды дифференцируемых функций потерь, который не требует выбора специфического критерия расщепления, а также функции агрегации наблюдений для расчёта значений в листьях. На каждом шаге построения дерева используются не значения целевой переменной, которые требуется приблизить, как в классических алгоритмах, а лишь первые и вторые производные функции потерь для каждого обучающего наблюдения, позволяющие уточнить предсказание расщепляемого листа.

2. Формальная постановка задачи

Пусть задана дважды дифференцируемая функция потерь $l(y, \hat{y})$, а также обучающая выборка $D = \{(x_i, y_i)\}_{i=1}^{N}$ из совместного распределения $(X, Y) \propto \mathcal{P}$, где каждый вектор признаков x_i состоит из M компонент $(x_i^{(1)}, \dots, x_i^{(M)}) \in \mathbb{R}^M$, а y_i – соответствующее значение целевой переменной. Целью является построение модели $f(x)$, минимизирующей эмпирический функционал риска:

$$\hat{\mathcal{L}}(f) = \frac{1}{N}\sum_{i=1}^{N}\big[l\big(y_i, f(x_i)\big)\big] \approx \mathbb{E}_{X,Y\sim\mathcal{P}}\big[l\big(Y, f(X)\big)\big].$$

Искомая функция f^* также должна быть из $\mathcal{F}$ – класса функций, представимых в виде деревьев решений. Пусть каждому узлу из множества листьев L дерева $f \in \mathcal{F}$ сопоставлена оценка значения целевой переменной в листе c_n, определяющая значение функции для любой точки x, попадающей в область листа R_n. При этом области листьев попарно не пересекаются, а объединение всех областей даёт $\mathbb{R}^M$. Дерево решений f может быть записано, как:

$$f(x) = \sum_{n\in L} \mathbb{I}[x \in R_n] \cdot c_n,$$

где $\mathbb{I}$ – индикаторная функция.

Пусть $Path(n)$ – набор узлов дерева на пути к узлу n, а $d\big(q, Path(n)\big)$ определяет для узла q, в какое из поддеревьев, левое или правое, идёт следующий узел пути $Path(n)$. Каждая область R_n определяется следующим образом:

$$\mathbb{I}[x \in R_n] = \prod_{q\in Path(n)} \mathbb{I}\big[x^{(k_q)} \le \theta_q \oplus d\big(q, Path(n)\big)\big],$$

где θ_n – пороговое значение, k_n – номер признака разбиения.

На каждом шаге построения дерева один из листовых узлов расщепляется на два, порождая две новых области, определяемые пороговым значением и номером признака разбиения, а также две оценки значений целевой переменной в новых листьях. Требуется определить правило разбиения узла n и оценки значений целевой переменной, чтобы минимизировать значение функции потерь для получаемого дерева.

3. Новый метод построения дерева решений

Рассмотрим узел n, который является на данном этапе листом и требуется расщепить. До расщепления для всех точек, попадающих в узел, то есть из области R_n, функция, соответствующая дереву, постоянна, и равна оценке значения целевой переменной в листе:

$$\forall x \in R_n \ (f(x) = c_n).$$

При построении разбиения будем рассматривать только точки обучающей выборки, попадающие в область R_n, и считать, что к функции значения в листе добавляется новая ступенчатая функция ϕ_n, уточняющая значения в левом и правом поддеревьях:

$$\phi_n(x) = \mathbb{I}\left[x^{(k_n)} \le \theta_n\right] \cdot u_n + \mathbb{I}\left[x^{(k_n)} > \theta_n\right] \cdot v_n,$$

где u_n, v_n – значения оценки целевой переменной слева и справа соответственно.

Ключевая идея нового метода состоит в том, чтобы новая функция, для всех точек, попадающих в узел, задавалась как сумма предыдущего значения в узле и уточняющей функции:

$$\tilde{f}(x) = \left(c_n + \phi_n(x)\right).$$

Определим параметры разбиения так, чтобы минимизировать ошибку новой функции узла $\tilde{f}(x)$. Для этого сперва разложим функцию потерь в ряд в окрестности предыдущей неизменной оценки значения целевой переменной в листе c_n, при $x \in R_n$:

$$l\left(y, \tilde{f}_n(x)\right) = l(y, c_n) + \ \phi_n(x) \cdot \left.\frac{\partial l(y,z)}{\partial z}\right|_{z=f(x)} + \frac{1}{2}\phi_n^2(x) \cdot \left.\frac{\partial^2 l(y,z)}{\partial z^2}\right|_{z=f(x)} + o\left(\phi_n^2(x)\right).$$

Обозначим для краткости первую и вторую производные как:

$$g_n(x,y) = \left.\frac{\partial l(y,z)}{\partial z}\right|_{z=f_n(x)}; h_n(x,y) = \left.\frac{\partial^2 l(y,z)}{\partial z^2}\right|_{z=f_n(x)}.$$

Тогда новая задача минимизации функции потерь с регуляризацией Ω, при условии справедливости разложения в ряд:

$$\tilde{\mathcal{L}}_n = \sum_{i=1}^{N} \mathbb{I}[x_i \in R_n] \cdot \left[\phi_n(x_i) \cdot g_n(x_i, y_i) + \frac{1}{2}\phi_n^2(x_i) \cdot h_n(x_i, y_i)\right] + \Omega(u_n, v_n;\ k_n, \theta_n).$$

Введём обозначения для краткости:

$$\mathbb{U}_i = \mathbb{I}\left[x_i^{(k_n)} \le \theta_n\right];\ \mathbb{V}_i = \mathbb{I}\left[x_i^{(k_n)} > \theta_n\right];\ \mathbb{I}_i = \mathbb{I}[x_i \in R_n].$$

Тогда функция потерь может быть разбита на слагаемые:

$$\tilde{\mathcal{L}}_n = \tilde{\mathcal{L}}_n^U + \tilde{\mathcal{L}}_n^V + \Omega,$$

$$\tilde{\mathcal{L}}_n^U = \sum_{i=1}^{N} \mathbb{I}_i \cdot \left[\mathbb{U}_i \left(u_n \cdot g_n(x_i, y_i) + \frac{1}{2}u_n^2 \cdot h_n(x_i, y_i)\right)\right],$$

$\tilde{\mathcal{L}}_n^V$ определяется аналогично. Пусть

$$G_n^U = \sum_{i=1}^{N} \mathbb{I}_i \cdot \mathbb{U}_i \cdot g_n(x_i, y_i);\ H_n^U = \sum_{i=1}^{N} \mathbb{I}_i \cdot \mathbb{U}_i \cdot h_n(x_i, y_i),$$

и аналогично для G_n^V, H_n^V.

Для обеспечения справедливости замены исходной функции потерь на её аппроксимацию может потребоваться ограничить ϕ_n. Для этого может быть введена так называемая скорость обучения $0 < \gamma \leq 1$, а итоговая функция уточнения будет выглядеть следующим образом:

$$\tilde{\phi}_n(x) = \gamma \cdot \phi_n(x).$$

Альтернативным способом введения ограничения на ϕ_n может быть l_1 или l_2-регуляризация. Пусть регуляризация задана в виде:

$$\Omega = M \cdot \left[\lambda_1 \cdot \|(u_n, v_n)^T\|_1 + \frac{1}{2}\lambda_2 \cdot \|(u_n, v_n)^T\|_2^2\right],$$

где $M = \sum_{i=1}^{N} \mathbb{I}[x_i \in R_n]$ – число попавших в лист точек.

Поскольку Ω также может быть разбита на сумму двух слагаемых, зависящих отдельно от u_n, и от v_n, минимизировать $\tilde{\mathcal{L}}_n^U$ можно также отдельно, по аналогии с работой [5]:

$$u_n = -\frac{G_n^U + A(G_n^U;\ M \cdot \lambda_1)}{M \cdot \lambda_2 + H_n^U},$$

где A – функция, отвечающая за l_1 регуляризацию:

$$A(z;t) = \begin{cases} t, z < -t \\ -z, |z| \leq t \\ -t, z > t. \end{cases}$$

Таким образом, при фиксированных параметрах разбиения θ_n, k_n, определяющих $\mathbb{U}_i, \mathbb{V}_i$, могут быть найдены оптимальные значения u_n, v_n. Далее аналогично классическим алгоритмам, таким как CART, осуществляется частичный или полный перебор признаков и порогов разбиения, рассчитывается значение приближения функции потерь и выбирается оптимальные параметры. Отметим, что при разбиении значение в листьях зависит от предыдущего значения в узле c_n, так как градиент функции потерь вычисляется в точке c_n. В отличие от [3], функция потерь раскладывается в ряд вокруг более точной аппроксимации в листе, что делает разложение в ряд справедливым при больших значениях скорости обучения, и позволяет достраивать дерево даже при изменении функции потерь, что возможно, например, при комбинировании нейронных сетей с деревьями решений, где нейронная сеть выступает в роли функции потерь. При этом в предложенном алгоритме для такого построения не требуется вычислять значение функции потерь для всех кандидатов на разбиение, что может быть вычислительно затратным.

4. Пример использования нового метода

Предложенный алгоритм подходит для различных задач машинного обучения, включая регрессию и классификацию. В качестве примера рассмотрим простой набор данных с двумя признаками, задающий две соединённые спирали, каждой из которых соответствует свой класс. Для решения данной задачи классификации в качестве функции потерь используется бинарная перекрёстная энтропия, задающаяся как:

$$l^{BCE}(y, p) = y \cdot \log(p) + (1 - y) \cdot \log(1 - p),$$

где p – предсказанная вероятность, зависящая от $\hat{y}$, целевого значения, предсказываемого деревом решений, и вычисляемая как:

$$p = \sigma(\hat{y}) = \frac{1}{1 + e^{-\hat{y}}}.$$

Производные, необходимые для применения алгоритма, зависят от p:

$$g_n(x, y) = p - y;\ h_n(x, y) = p \cdot (1 - p).$$

На рис. 1 приведено (а) обучающее множество, которое использовалось для построения дерева решений, где цветом обозначены классы; (б) предсказания дерева решений, построенного классическим алгоритмом CART; (в) предсказания дерева, построенного предложенным методом без l_2-регуляризации и (г) – с коэффициентом регуляризации 0,1. Для всех деревьев глубина не превосходит 10. Видно, что при использовании регуляризации в части областей предсказанные оценки вероятностей отличны от 0 и 1 и более близки друг к другу для соседних областей, в которых не содержалось большого числа обучающих примеров. В данном примере, при использовании оценки качества ROC-AUC, и её расчёте на отдельном множестве с 20000 тестовых наблюдений, значение оценки качества CART-дерева равно 0,92; дерева с предложенным алгоритмом без регуляризации 0,94; с l_2-регуляризацией 0,98.

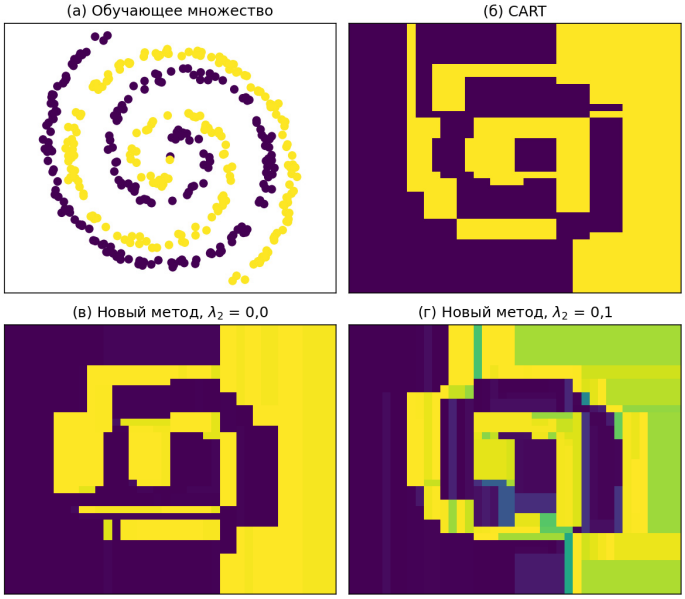

Рис. 1. Пример данных и построенных деревьев решений.

Таким образом, предложенный метод построения деревьев решений позволяет решать задачи машинного обучения на основе лишь первой и второй производной функции потерь, рассчитанной для всех обучающих примеров. Дальнейшим направлением исследований является адаптация алгоритма для работы с векторными целевыми переменными, а также комбинация данного алгоритма с ансамблевыми методами, такими как градиентный бустинг деревьев решений и применение данного алгоритма для обучения композиции деревьев с нейронными сетями.

Список литературы

1. Breiman L. Random forests // Machine learning. 2001. Vol. 45. P. 5-32.
2. Geurts P., Ernst D., Wehenkel L. Extremely randomized trees // Machine learning. 2006. Vol. 63. P. 3-42.
3. Chen T., Guestrin C. Xgboost: A scalable tree boosting system // Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. 2016. P. 785-794.
4. Breiman L, Friedman J, Olshen RA, Stone CJ. Classification and regression trees // New York: Chapman & Hall. 1984.
5. Konstantinov A.V., Utkin L.V. Interpretable ensembles of hyper-rectangles as base models // Neural Computing and Applications. 2023. Vol. 35, No. 29. P. 21771-21795.