УДК 004.62

ДЕРЕВЬЯ РЕШЕНИЙ В ЗАДАЧАХ КЛАССИФИКАЦИИ: ОСОБЕННОСТИ ПРИМЕНЕНИЯ И МЕТОДЫ ПОВЫШЕНИЯ КАЧЕСТВА КЛАССИФИКАЦИИ

Полин Я.А., Зудилова Т.В., Ананченко И.В., Войтюк Т.Е.

Национальный исследовательский университет информационных технологий, механики и оптики, Санкт-Петербург, e-mail: polin.ya@mail.ru, zudilova@ifmo.spb.ru, anantchenko@yandex.ru, taire2006@yandex.ru

В статье рассматриваются аспекты применения деревьев и леса решений для задач классификации. Большинство современных информационных систем, приложений, рекомендательных систем используют деревья решений в задачах, где необходимо определить принадлежность объекта к некоторому классу из непересекающегося множества классов. К таким задачам можно отнести системы распознавания текста, речи, когнитивного поиска, анализ изображений, жестов, выявление спама и другие нелинейные задачи. Рассмотрена математическая модель задачи классификации с использованием деревьев решений и леса решений. Приводится обзор особенностей деревьев решений, которые целесообразны для подобных задач, а также такие их достоинства и недостатки, как автоматический отбор признаков, интерпретируемость механизма принятия решений, управляемость и автономность вершин, зависимость от сбалансированности числа обучающих примеров разных классов, переобучение, экспоненциальное уменьшение обучающей выборки, разбалансировка. Описаны методы, которые можно использовать для устранения проблемы, возникающей при уменьшении тренировочной выборки, а именно случайное перемешивание и добавление схожих примеров в выборку. Затронут вопрос тестирования классификаций с помощью метода скользящего контроля. Уделяется внимание вопросам балансировки обучающей выборки и применения методов, позволяющих избежать переобучения.

Ключевые слова: качество классификации, деревья решений, классификаторы, алгоритмы, машинное обучение

DECISION TREES IN CLASSIFICATION PROBLEMS: APPLICATION FEATURES AND METHODS FOR IMPROVING THE QUALITY OF CLASSIFICATION

Polin Ya.A., Zudilova T.V., Ananchenko I.V., Voytiuk T.E.

ITMO University, Saint-Petersburg, e-mail: anantchenko@yandex.ru, zudilova@ifmo.spb.ru, polin.ya@mail.ru, taire2006@yandex.ru

The article considers the aspects of applying tree and forest solutions to classification problems. Most modern information systems, applications, and recommendation systems use decision trees in tasks where it is necessary to determine whether an object belongs to a certain class from a disjoint set of classes. These tasks include text recognition, speech recognition, cognitive search, image analysis, gestures, spam detection, and other non-linear tasks. A mathematical model of the classification problem using decision trees and decision forests is considered. An overview of the features of decision trees that are convenient for such problems, as well as their advantages and disadvantages, such as automatic feature selection, interpretability of the decision-making mechanism, manageability and autonomy of vertices, dependence on the balance of the number of training examples of different classes, retraining, exponential reduction of the training sample, and unbalancing. We describe methods that can be used to solve the problem that occurs when reducing the training sample, namely, random mixing and adding similar examples to the sample. The question of testing classification using the sliding control method is raised. Attention is paid to balancing the training sample and applying methods that avoid overtraining.

Keywords: quality of classification, decision trees, classifiers, algorithms, machine learning

Большинство современных информационных систем, приложений, рекомендательных систем используют деревья решений в тех случаях, когда необходимо отнести объект к определенному классу из некоторого их числа, которые при этом не пересекаются. Если рассматриваемая переменная является дискретной, то полученное дерево называют классификационным, а если непрерывной, то дерево будет регрессионным. Широко известными примерами применения деревьев регрессий и классификационных деревьев можно назвать устройства и системы распознавания речи, текста, изображений (в том числе и анализ изображений или даже видео), жестов, алгоритмы и инструменты когнитивного поиска, системы обнаружения спама в веб-приложениях или соцсетях.

Параметры алгоритмов в машинном обучении настраиваются автоматически на основе некоторой обучающей выборки, а иными словами, некоторого множества объектов с известными атрибутами, характеристиками или соответствующими им ответами. Качество прогнозов и решения поставленной перед алгоритмами задачи напрямую зависит от качества обучения на начальной выборке.

В контексте рассматриваемых классификационных моделей алгоритмом можно назвать функцию, на вход которой поступает рассматриваемый объект, который необходимо отнести к определенному классу,

а на выходе возвращается определенный класс, которому исследуемый объект соответствует. Если же говорить о понятийном аппарате классификационных моделей, именуемых деревьями решений, то тут важно обозначить, что по своей структуре деревья образуются вершинами и листьями. Вершины, называемые также узлами дерева, являются условиями, по которым проверяется соответствие объекта некоторому заранее определенному атрибуту. Листья являются конечными, терминальными элементами деревьев и содержат значения (то есть классы, иначе называемые найденными решениями). В процессе обучения дерева корректируются и подстраиваются параметры узлов дерева, а также значений в листьях с целью создания качественного классификационного алгоритма. Обучающими примерами для задачи формирования правильно работающего дерева становится некоторая заранее подготовленная группа объектов с известными классами.

Математически это можно записать следующим образом. Если задано конечное множество объектов $X = \{x_1,\ldots, x_L\}$ и множество алгоритмов $A = \{a_1,\ldots, a_D\}$, а также некоторая бинарная функция, характеризующая потери l, такая что: $A\times X \to \{0,1\}$, $l(a, x) = 1$ тогда и только тогда, когда алгоритм ошибается на некотором объекте x. Количество ошибок алгоритма на выборке определяется по формуле

$$n(a, X) = \sum_{x\in X} l(a, x). \quad (1)$$

При необходимости понять частотность ошибок алгоритма на некоторой выборке, можно воспользоваться следующей формулой:

$$v(a, X) = n(a, x)/|X|. \quad (2)$$

Этот параметр частоты ошибок, совершаемых алгоритмом на контрольной выборке, называется качеством классификации.

Цель данной работы заключается в рассмотрении преимуществ и недостатков популярного метода решения задач классификации – деревьев решений, а также в определении возможных методов, позволяющих устранить или уменьшить недостатки деревьев решений.

1. Достоинства и недостатки использования деревьев решений в качестве классификаторов

а) Отбор признаков в вершинах узлов происходит автоматически

При использовании деревьев решений информативные признаки отбираются автоматически в процессе обучения. Этот факт делает метод классификации с помощью деревьев решений более привлекательным относительно других методов машинного обучения. Кроме того, в этом методе отсутствует дополнительный отбор признаков и существует возможность формировать собственные произвольные наборы признаков.

б) Интерпретируемость механизма принятия решений

В тех случаях, когда процесс работы алгоритма должен быть понятен для бизнеса или для верхнеуровневой оценки логики принятия решений в процессе его работы, есть возможность сформировать правила, по которым принимаются решения в понятной эксперту форме. Это свойство, вкупе с естественным языком предикатов в узлах деревьев, также может помочь найти ошибку в логике работы машинного алгоритма.

в) Управляемость и автономность вершин

В случаях, когда объекты тестовой выборки классифицируются неправильно, есть возможность переобучать те вершины, которые совершают ошибку. Также эта особенность позволяет обучать другим алгоритмам только части деревьев, используя более эффективные алгоритмы. Такой подход дает возможность управлять обучением, а также изменять результаты классификации определенных объектов, не затрагивая работу всего дерева.

г) Качество классификации зависит от баланса числа различных классов в обучающем множестве

Последовательность алгоритма создания дерева решений такова, что для начала процесса обучения в узел дерева выбирается максимально информативный предикат, который разбивает полученное множество на две части. Существует формула оценки информативности условия, размещенного в вершине:

$$l = \frac{|L|}{|L|+|R|} * H(L) + \frac{|R|}{|L|+|R|} * H(R), \quad (3)$$

где L и R – множества примеров, попадающие в результате разбиения по условию в левый и правый узлы дерева соответственно;

Оценка же информативности рассматриваемых тренировочных выборок производится по функциям $H(L)$ и $H(R)$, которые оцениваются с помощью энтропии Шеннона и индекса Джини.

Индекс Джини вычисляется по формуле

$$H(S) = 1 - \sum_{k=1}^{K} p^2(k \mid S), \quad (4)$$

где $p(k|S)$ – доля экземпляров класса k в S;

K – количество классов;
S – некоторое множество обучающих объектов.

Особенность использования дерева решений такова, что в процессе обучения оно использует классы с большим числом тренировочных объектов, но пропускает те, которые содержат малое число примеров. В то время как для качественной классификации важно иметь баланс между классами в обучающей выборке. Можно рассматривать эту особенность дерева решений, с одной стороны, как достоинство, но в то же время это является и ее недостатком. В случае возникновения диспропорции в классах обучающей выборки, процесс обучения модели выполняется некорректно. А в качестве положительного аспекта этой особенности можно сказать, что вариация баланса между тренировочными объектами позволяет управлять обучением и корректировать его в нужную сторону.

д) Обучающая выборка уменьшается по экспоненциальной зависимости

В процессе обучения тренировочное множество последовательно разделяется на подмножества, проходя через каждую последующую вершину. Благодаря этому на каждом последующем вложенном уровне дерева находится меньше обучающих объектов, чем на предыдущем. При условии деления выборки на две равные части при каждой итерации прохождения через узел дерева, можно оценить размер выборки на самом низком уровне дерева (самая вложенная вершина). Если предположить, что уровень дерева – *a*, то размер выборки на вложенном листе станет в 2^a раз меньше, чем ее начальный. Таким образом, при достижении более вложенных листов меньший размер обучающих примеров влечет за собой переобучение, и, следовательно, для корректного обучения дерева стоит подавать на вход модели значительно большие тренировочные множества.

2. Использование лесов решений для задач классификации

Использование леса решений для задач классификации может решить проблему переобучения. Итоговый ответ леса решений – это результат классификации нескольких деревьев, определенных голосованием (большинство предсказывают одно и то же). Положим, некоторое множество деревьев из леса решений, из которых каждое определяет объект *x* из множества *X* к одному из классов *c*, принадлежащего некоторому множеству классов *Y*. Применяется метод простого голосования, при котором для каждого класса *c* подсчитывается число деревьев, которые относят обозначенные объекты к данному классу. Если мы полагаем, что $f_c^t = 1$, то дерево из лесов решений определяет рассматриваемый объект к классу *c*. Количество деревьев в данном случае можно вычислить по формуле

$$G_c(x) = \frac{1}{T_c}\sum_{t=1}^{T_c}(x), c \in Y. \quad (5)$$

Итоговым ответом, который дает классификатор леса решений, засчитывается класс, определенный большинством деревьев из леса: $a(x) = \arg\max_{c\in Y} Gc(x)$. Независимость ошибок классификации при таком подходе является основой качественного распознавания объектов. В пограничном случае, при котором все деревья из леса ошибаются на одних примерах, использование леса решений не дает никаких дополнительных преимуществ перед использованием одного дерева решений. Верхняя граница ошибки обобщения, для оценки подобных случаев, может быть рассчитана по формуле

$$GE = p\frac{1-s^2}{s^2}, \quad (6)$$

где *GE* – ошибка обобщения;
p – средняя попарная корреляция между ошибками деревьев в лесу;
s – качество классификации каждого отдельного дерева [1].

Существует другой краевой случай – когда деревья леса обучаются с использованием одних и тех же тренировочных выборок с помощью одинаковых методов. В результате такого подхода деревья формируются идентично или с минимальными отличиями. Для того, чтобы избежать таких ситуаций, используется один из вариантов группы жадных алгоритмов – случайный лес [2]. В тех же случаях, когда необходимо достигнуть независимости ошибок внутри леса решений, используются отдельные специализированные методы (Bagging, XGBoost и другие). Рассмотрим самые популярные из них – Bagging и XGBoost).

При применении метода Bagging [3] каждое дерево из леса обучается на своем собственном сформированном случайно из общей выборки подмножества. Из негативных эффектов данного метода можно отметить исчезновение эффекта при увеличении численности тренировочного множества. При увеличении числа объектов все подвыборки становятся похожими друг на друга, в связи с тем, что они формируются из единого вероятностного распределения и влияние случайных отклонений

при формировании таких подмножеств слабеет [4].

Метод Boosting использует в своем алгоритме назначение каждому примеру из тренировочной выборки (но в зависимости от степени сложности) определенные веса $(w_1^1,\ldots,w_m^1)$. Веса при этом формируются вероятностно. Для них справедлива следующая формула:

$$\sum_{j=1}^{m} w_j^1 = 1, w_j^1 \in [0,1]. \quad (7)$$

Начальное распределение весов устанавливается равномерным и равным $w_j^1 = \frac{1}{m}, j \in 1,\ldots,m.$

Процесс алгоритма осуществляется следующим образом: сначала проходит обучение первое дерево. Следом классифицируются тренировочные обучающие примеры и их веса перераспределяются. Если обучающие примеры были определены верным образом, их вес снижается, а у классифицированных неверно – повышается. Второе и последующие деревья формируются с учетом скорректированных в тренировочной выборке весов, и так происходит до требуемого количества деревьев или опираясь на необходимую для данной модели ошибку классификации. Стоит принять во внимание, что при расчете информативности признака по формуле вместо доли объектов используется отношение суммы весов объектов, принадлежащих данному классу, к сумме весов всех объектов подмножества *S*:

$$\mathrm{p}(k|S) = \frac{\sum_{i=1}^{|S|} I[y_i = k] * w_i^d}{\sum_{i=1}^{|S|} w_i^d}. \quad (8)$$

Особенность применения алгоритма XGBoost к лесу деревьев заключается в последовательном обучении. Каждый шаг сопровождается вычислениями отклонений от предсказаний уже обученного леса на обучающей выборке. Каждая следующая модель, добавляемая в XGBoost, должна предсказывать эти отклонения. Благодаря этому, добавление предсказания нового дерева к лесу позволяет уменьшать среднее отклонение всей модели и это оптимизирует весь лес решений. Критерием добавления новых деревьев в лес является уменьшение ошибки, либо же добавление новых деревьев прекращается, если выполняется одно из правил ранней остановки [5].

3. Методы тестирования в машинном обучении при использовании деревьев решений

Для оценки качества алгоритма в деревьях решений можно применять метод скользящего контроля. Он помогает протестировать качество классификатора [6].

Начальное множество обучающих примеров разбивается на два подмножества, называемые тренировочным и контрольным. Сначала алгоритм проходит обучение на тренировочном множестве, затем рассчитывается число ошибок на контрольной выборке, и это число выполняет роль меры качества имеющегося алгоритма.

При оценке классификаторов таким способом невозможно выявить ошибки в формировании обучающего множества, а этот факт может оказаться критичнее, чем другие его негативные особенности. По существу, вся проблема исходит из того, что тренировочное и контрольное подмножества формируются на одном распределении, не всегда полно и корректно передающем начальные исходные данные, что создает ложные зависимости при реализации алгоритма машинного обучения. В таком случае оценка качества результирующей классификации может быть осуществлена двумя способами: на выборке, взятой из отличного от существующего независимого источника (лучшее решение данной проблемы), или оценив среднее значение разных источников. При таком способе важно помнить, что величина оценки будет напрямую зависеть от распределения примеров в выборке и их соотношения. Бывают прецеденты, при которых из тестовой выборки удаляются объекты, не изменяющие результаты классификации. Так улучшается качество классификации на тестовых объектах, но при этом ухудшается качество классификации на «боевых» данных. Не рекомендуется использовать такой прием еще и по той причине, что так можно ненамеренно изменить алгоритмы под тестовую выборку и они перестанут работать на других значениях, которые не похожи на тестовые.

4. Предлагаемые методы решения проблем при работе с деревьями решений

Приведем возможные способы решения проблемы, возникающей при уменьшении тренировочной выборки. Первый способ – случайное перемешивание. В качестве достоинств данного метода можно указать простоту реализации, а также минимальное значение используемой памяти. Для реализации данного способа после идентификации признака в узле дерева тренировочное множество делится на две группы объектов,

а функция, по которой происходит это разбиение, зависит от обозначенного признака и порога. Математически эту закономерность можно описать так:

$$s(x,\Theta)=\begin{cases}0, \text{ если } x(\Theta_1)<\Theta_2\\ 1 \text{ в иных случаях,}\end{cases} \quad (9)$$

где функция разбиения $S(x, \Theta)$, признака разбиения Θ_1 и порог Θ_2.

В зависимости от значения признака подмножества разделяются на левое и правое. Если функция равна нулю – определяются в левое подмножество, значение единица классифицирует объект в правое. Метод случайного перемешивания реализуется случайными изменениями подмножеств, в которое попадает объект. Для краевых случаев, когда левое подмножество становится больше правого по числу объектов, при этом не учитывая случайно перемешанные объекты, математическое ожидание тех из них, кто был перемещен из левого подмножества в правое, будет больше, чем для тех, кто перемещался из правого подмножество в левое. Результатом таких вероятностных изменений станет более равномерное распределение элементов множеств по вершинам, а также элиминация или уменьшение тех вершин, где число тренировочных примеров было недостаточно для корректного обучения модели.

Второй метод решения проблемы уменьшения тренировочного множества – это искусственное увеличение исходной выборки путем добавления схожих обучающих объектов. В данной работе уже упоминалось, что часть вершин деревьев решений при обучении может получить малое число тренировочных данных, в результате чего возникает переобучение. Решением этой проблемы может стать добавление новых, похожих на имеющиеся, примеров в тренировочную выборку. Недостатком данного способа может стать увеличение объема памяти, используемой при обучении, и если исходный набор данных достаточно большой, то в совокупности с временем добавления дополнительных данных этот способ может быть затратным по времени исполнения.

Заключение

В статье рассмотрена задача классификации с использованием алгоритма деревьев решений; описаны такие особенности: автоматический отбор признаков, интерпретируемость механизма принятия решений, управляемость и автономность вершин, зависимость от сбалансированности числа обучающих примеров разных классов, переобучение, экспоненциальное уменьшение обучающей выборки, разбалансировка. Предложены методы устранения недостатков, рассмотренных в статье.

Список литературы

1. Breiman L. Random Forests. Machine Learning. 2001. Vol. 45(1). P. 5–32.

2. Sedgewick R., Wayne K. Algorithms (4th Edition). Boston, Addison-Wesley Professional, 2011. 992 p.

3. Чистяков С.П. Случайные леса: Обзор // Труды Карельского научного центра РАН. 2013. № 1. С. 117–136.

4. Синяев И.Ф., Шестернева О.В. Исследование bagging подхода при построении ансамбля моделей для повышения точности классификации // Актуальные проблемы авиации и космонавтики. 2014. № 10. С. 300.

5. Chen Tianqi, Guestrin Carlos. XGBoost: A Scalable Tree Boosting System. In Krishnapuram Balaji; Shah Mohak, Smola Alexander J., Aggarwal Charu C., Shen Dou, Rastogi, Rajeev (eds.). Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016. ACM. P. 785–794.

6. Arlot Sylvain, Celisse Alain. A survey of cross-validation procedures for model selection. Statist. Surv. 2010. Vol. 4. P. 40–79.