Труды Карельского научного центра РАН
№ 1. 2013. С. 117–136

УДК 512.2

СЛУЧАЙНЫЕ ЛЕСА: ОБЗОР

С. П. Чистяков

*Институт прикладных математических исследований
Карельского научного центра РАН*

В статье представлен обзор современного состояния исследований в области случайных лесов – статистического метода, предназначенного для решения задач классификации и регрессии. Приведен исторический экскурс развития деревьев решений и ансамблей классификаторов и описаны основные понятия (загрязненность, расщепление, баггинг, бустинг и др.), используемые при их построении. Рассматриваются вопросы, касающиеся состоятельности метода и сравнения его с другими методами классификации. Представлены возможности использования случайных лесов для задач нахождения наиболее информативных признаков, кластеризации, выделения аномальных наблюдений и определения прототипов классов. Кратко рассмотрены некоторые неклассические разновидности деревьев решений и случайных лесов, а именно: косоугольные деревья, случайные леса выживаемости, квантильные леса регрессий, логические случайные леса, вероятностные случайные леса и потоковые случайные леса. Также приведен обзор соответствующего программного обеспечения с акцентом на пакет R — свободно распространяемое программное обеспечение для статистических вычислений и графики, доступное на платформах Linux, Windows, Mackintosh.

Ключевые слова: деревья решений, ансамбли классификаторов, баггинг, случайные леса, классификация, регрессия, кластеризация, пакет R.

S. P. Chistiakov. RANDOM FORESTS: AN OVERVIEW

This paper presents an overview of the state-of-the-art in the studies of random forests — a statistical method designed to deal with problems of classification and regression. We tell about the history of decision trees and classifier ensembles and describe the corresponding basic ideas (impurity, split, bagging, boosting, etc.). Some issues of the consistency of the method are considered. Applicability of random forests to the problems of finding most informative features, clustering, finding outlier observations and class prototypes is surveyed. Several non-classical variants of decision trees and random forests is considered, namely: oblique trees, survival random forests, quantile regression forests, logical random forests, probabilistic random forests and streaming random forests. We also survey the corresponding software with the emphasis on R package — open source environment for statistical computing and graphics which is freely available for the computing platforms Linux, Windows, Mackintosh.

Key words: decision trees, classifier ensembles, bagging, random forests, classification, regression, clustering, R package.

Введение

Понятие *случайный лес* впервые было введено в научный обиход в работах [6, 7], см. также [8]. В этих статьях рассматривалось множество корневых лесов с помеченными вершинами, на котором задавалось равномерное распределение вероятностей. Позднее появилась монография [81], в которой изучались случайные леса с распределениями, отличными от равномерного. Таким образом, с точки зрения теории вероятностей, случайные леса являются частным случаем известного понятия *случайный элемент* (см. [5]). Однако в 2001 г. в статье [25] был предложен новый метод классификации и регрессии, также получивший название *случайный лес*. В этом смысле термин *случайный лес* широко используется в таких дисциплинах как машинное обучение (machine learning), распознавание образов, дисциплине известной как "Data Mining"[1] и, в меньшей степени, в прикладной статистике. Настоящий обзор посвящен этому методу.

Метод основан на построении большого числа (ансамбля) деревьев решений (это число является параметром метода), каждое из которых строится по выборке, получаемой из исходной обучающей выборки с помощью бутстрепа (т. е. выборки с возвращением). В отличие от классических алгоритмов построения деревьев решений [21, 91] в методе случайных лесов при построении каждого дерева на стадиях расщепления вершин используется только фиксированное число случайно отбираемых признаков обучающей выборки (второй параметр метода) и строится полное дерево (без усечения), т. е. каждый лист дерева содержит наблюдения только одного класса. Классификация осуществляется с помощью голосования классификаторов, определяемых отдельными деревьями, а оценка регрессии — усреднением оценок регрессии всех деревьев. Известно (см., например, [66]), что точность (вероятность корректной классификации) ансамблей классификаторов существенно зависит от разнообразия (diversity) классификаторов, составляющих ансамбль или, другими словами, от того, насколько коррелированы их решения. А именно, чем более разнообразны классификаторы ансамбля (меньше коррелированность их решений), тем выше вероятность корректной классификации. В случайных лесах решения составляющих их деревьев слабо коррелированны вследствии двойной "иньекции случайности" в алгоритм построения случайного леса – на стадии бутстрепа и на стадии случайного отбора признаков, используемых при расщеплении вершин деревьев.

Метод быстро получил признание как в статистическом сообществе, так и в среде исследователей, использующих методы распознавания образов в своей работе и в настоящее время является одним из наиболее популярных методов классификации и непараметрической регрессии. Причиной этого явилась не только высокая точность классификации (а по мнению автора и не столько), обеспечиваемая методом, но и другие его достоинства. Именно:

- метод гарантирует защиту от переподгонки (overfitting)[2] даже в случае, когда количество признаков значительно превышает количество наблюдений. Это свойство выделяет метод "случайный лес" среди множества других методов классификации и является чрезвычайно ценным для решения многих прикладных задач;
- для построения случайного леса по обучающей выборке требуется задание всего двух параметров, которые требуют минимальной настройки (tuning);
- метод Out-Of-Bag (OOB), предложенный Брейманом [23], обеспечивает получение естественной оценки вероятности ошибочной классификации случайных лесов на основе наблюдений, не входящих в обучающие бутстреп выборки, используемые для построения деревьев (эти наблюдения называются OOB выборками);
- случайные леса могут использоваться не только для задач классификации и регрессии, но и для задач выявления наиболее информативных признаков, кластеризации, выделения аномальных наблюдений и определения прототипов классов;
- обучающая выборка для построения случайного леса может содержать признаки, измеренные в разных шкалах: числовой, порядковой и номинальной, что недопустимо для многих других классификаторов;

[1] Общепринятого перевода этого термина на русский язык не существует. Иногда используемый дословный перевод "добыча данных" не выдерживает никакой критики.

[2] Под переподгонкой понимается ситуация, при которой классификатор хорошо классифицирует наблюдения обучающей выборки, но непригоден для классификации наблюдений, не входящих в нее.

- метод допускает легкую параллелизацию (т. е. программную реализацию, пригодную для параллельных вычислений), что весьма существенно при больших объемах обучающей выборки.

Автору представляется, что развитие метода случайных лесов проходило в следующих направлениях:

- исследование свойств самого метода, т. е. аналитическая и экспериментальная работа по оценке точности, сравнению с другими ансамблевыми методами классификации и т. д.;

- развитие возможностей метода, ориентированных на решение задач, непосредственно не связанных с задачами классификации и регрессии (см. выше);

- разработка на основе метода других родственно связанных методов, таких как случайные леса выживаемости, квантильные регрессионные леса, логические случайные леса, вероятностные случайные леса и потоковые случайные леса;

- использование схемы построения случайных лесов для построения ансамблей классификаторов, не являющихся деревьями — ансамблей наивных байесовских классификаторов и мультиномиальных логистических моделей;

- разработка алгоритмов и программных средств, реализующих метод.

В соответствии с этой классификацией исследований и построен данный обзор.

Первый раздел представляет собой краткий экскурс в историю возникновения метода. Рассмотрены элементы метода — деревья решений, ансамбли классификаторов, бэггинг (агрегированный бутстреп) и метод случайных подпространств. В этом разделе автор счел целесообразным также кратко описать основные понятия, используемые при построении деревьев решений — загрязненность (impurity), расщепление (split) и усечение (pruning) деревьев решений.

Во втором разделе приведен алгоритм построения случайного леса и его определение, принадлежащие Брейману. Приведены некоторые теоретические и экспериментальные результаты относительно состоятельности метода. Обсуждаются достоинства и недостатки метода.

В третьем разделе рассмотрены возможности использования случайных лесов для задач отбора наиболее информативных признаков, кластеризации, выявления аномальных наблюдений и определения прототипов классов.

В четвертом разделе рассмотрены разновидности случайных лесов — случайные леса выживаемости (random survival forests) [63], квантильные регрессионные леса (quantile regression forests) [73], логические регрессионные леса (logic forest) [117] и другие. В этом разделе также кратко рассмотрены ансамбли классификаторов, строящихся по той же схеме, что и случайные леса, а именно, ансамбли наивных байесовских классификаторов (naive bayes classificator) и ансамбли мультиномиальных логистических моделей [84].

Пятый раздел посвящен обзору программного обеспечения для построения деревьев решений и случайных лесов с акцентом на пакет R — свободно распространяемое программное обеспечение для статистических вычислений и графики, доступное на платформах Linux, Windows, Mackintosh [85]. Пакет R не является пакетом в классическом понимании. Это среда для статистических вычислений и графики с интерпретирумым языком программирования (который также носит название R). За рубежом R получил широкое распространение как в среде профессиональных статистиков, так и в среде исследователей, регулярно применяющих статистические методы в своей работе. В настоящее время R содержит более 3000 пакетов практически по всем статистическим методам, методам распознавания образов, машинного обучения и Data Mining разработанные различными институтами, группами исследователей и отдельными исследователями по всему миру.

В заключении кратко обсуждаются возможные направления дальнейших исследований.

Обзор ни в коей мере не носит учебно-методический характер и предполагает, что читатель знаком с основными понятиями и методами распознавания образов и прикладной статистики (в частности, предполагается знакомство с деревьями решений). Разделы, посвященные деревьям решений, могут быть найдены практически в любом учебнике по машинному обучению и Data Mining. Обзор предназначен для читателей, чьи профессиональные интересы связаны с прикладной статистикой и распознаванием образов, однако автор надеется, что он также окажется полез-

ным и для исследователей в других областях науки.

Деревья решений

В этом разделе приведен краткий экскурс, посвященный истории возникновения и развития идей, синтез которых привел к появлению метода случайных лесов. Отправной точкой в этом изложении являются деревья решений. Идея, лежащая в основе деревьев решений, состоит в разбиении множества возможных значений вектора признаков (независимых переменных) на непересекающиеся множества и подгонке простой модели для каждого такого множества. Понятие дерева решений опирается на понятие дерева из теории графов и понятие обучающей выборки из распознавания образов. Ниже кратко описаны эти понятия.

Граф $G = (V, E)$ состоит из конечного непустого множества V, элементы которого называются вершинами и множества пар вершин E, называемых ребрами. Путем в графе называется последовательность ребер вида $(v_1, v_2), (v_2, v_3), \ldots, (v_{m-1}, v_m)$. Если $v_1 = v_m$, то такой путь называется циклом. Если пара вершин v, w, образующая ребро (v, w), является упорядоченной, то такое ребро называется ориентированным или дугой, ведущей из вершины v в вершину w. Если все ребра графа ориентированы, то такой граф называется ориентированным. Дерево представляет собой связный граф без циклов. Под корневым деревом понимается дерево, в котором одна вершина выделена и называется корнем. Далее рассматриваются только ориентированные корневые деревья, в которых дуги направлены по направлению от корня. Заметим, что такие деревья удовлетворяют следующим условиям:

- существует только одна вершина, называемая корнем, в которую не ведет ни одна дуга;
- в каждую вершину (исключая корень) ведет только одна дуга;
- существует единственный путь от корня к любой вершине.

Если (v, w) — некоторая дуга, то вершина v называется родителем w, а вершина w — потомком вершины v. Вершина, не имеющая потомков, называется терминальной вершиной или листом. Дерево называется бинарным, если каждая его вершина (за исключением терминальных вершин) имеет ровно двух потомков.

Понятие обучающей выборки является ключевым в распознавании образов. Под обучающей выборкой понимается независимая выборка $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^{l}$ из некоторого (неизвестного) распределения $P(\mathbf{x}, y) = P(\mathbf{x})P(y|\mathbf{x})$. Здесь $\mathbf{x}_i,\ i = 1, 2, \ldots, l$ — векторы признаков (называемые прецедентами), координаты которых представляют значения n признаков (независимых переменных), измеренных на некотором объекте (образе). Множество всех возможных значений векторов признаков $\mathcal{X}$ называется пространством образов. Признаки могут быть измерены в различных шкалах — числовой, порядковой или номинальной. Соответствующие y_i представляют собой значения зависимой переменной. Если y_i может принимать только конечное число значений, т. е. $y_i \in \{\omega_1, \omega_2, \ldots, \omega_c\},\ c \geqslant 2$, то мы имеем задачу классификации. В этом случае y_i называется меткой класса и определяет принадлежность соответствующего объекта к одному из c классов, а сам признак называется классовым; если же y_i измерены в числовой шкале, то мы имеем задачу регрессии; в этом случае признак называется откликом;

Деревом решений называется дерево, с каждой вершиной t которого связаны [1][3]:

1. Некоторое подмножество $\mathcal{X}_t \subset \mathcal{X}$; с корневой вершиной связывается все пространство образов $\mathcal{X}$;
2. Подвыборка $\mathcal{D}_t \subset \mathcal{D}$ обучающей выборки $\mathcal{D}$, такая, что $\mathcal{D}_t = \{(\mathbf{x}, y) \in \mathcal{D} : \mathbf{x} \in \mathcal{X}_t\}$; таким образом с корневой вершиной связывается вся выборка $\mathcal{D}$;
3. Некоторая функция (правило) $f_t : \mathcal{X} \to \{0, 1, \ldots, k_t - 1\}$ (здесь $k_t \geqslant 2$ — количество потомков вершины t), определяющая разбиение множества $\mathcal{X}$ на k непересекающихся подмножеств. С терминальными вершинами не связывается никакая функция.

Обозначим $t_{i(t)},\ i = 0, 1, \ldots, k_t - 1$ вершину, являющуюся i–м потомком вершины t. Множество $\mathcal{X}_t$ и правило f_t определяют множества $\mathcal{X}_{t_{i(t)}}$ следующим образом:

$$\mathcal{X}_{t_{i(t)}} = \mathcal{X}_t \cap \{\mathbf{x} \in \mathcal{X} : f_t(\mathbf{x}) = i\}. \qquad (1)$$

Цель построения дерева решений состоит либо в классификации векторов $\mathbf{x}$ из распределения $P(\mathbf{x})$, либо в оценке условного математического ожидания отклика при данном значении $\mathbf{x}$. Процесс принятия решений начинается с корневой вершины и состоит в последовательном применении правил, связанных с вершинами

[3] В этой работе использовался термин "древообразные классификаторы".

дерева. Результатом этого процесса является определение терминальной вершины t такой, что $\mathbf{x} \in \mathcal{X}_t$. В случае классификации вектор $\mathbf{x}$ относится к классу, являющемуся мажорантным (наиболее часто встречающимся) в подвыборке $\mathcal{D}_t$, соответствующей данной терминальной вершине, а в случае регрессии оценка условного математического ожидания отклика представляет собой среднее значение отклика в этой подвыборке (в этом случае дерево решений часто называют деревом регрессий).

Существует несколько причин, обусловивших широкое применение деревьев решений в самых различных областях несмотря на их, вообще говоря, невысокую точность классификации. Основными из них, по мнению автора, являются следующие:

1. Обучающая выборка для построения деревьев решений может содержать признаки, измеренные как в числовой, так и в номинальной шкале, что недопустимо для многих других классификаторов, предполагающих существование некоторой объективной метрики, измеряющей близость наблюдений;

2. Иерархическая структура деревьев решений сводит принятие решения (классификацию) к последовательности более простых и интуитивно понятных решений;

3. Деревья решений служат удобной моделью представления знаний в экспертных системах;

4. Алгоритмы построения (индукции) деревьев решений инвариантны к масштабированию значений признаков.

Хотя деревья решений как инструмент классификации известны достаточно давно, однако их широкое применение началось с разработкой алгоритма CART (Classification And Regression Trees) и публикацией классической работы [21], идеи которой дали начало многочисленным исследованиям. Алгоритм CART является базовым (generic) алгоритмом, на основе которого может быть построено множество конкретных алгоритмов, приводящих к построению различных деревьев решений. Алгоритм основан на идее рекурсивного разбиения обучающей выборки на две более однородные подвыборки с помощью одного из признаков. Для реализации этой идеи необходимо определить понятие меры однородности.

Загрязненность

Обычно вместо меры однородности рассматривается противоположная по смыслу мера загрязненности (impurity). Пусть t — некоторая вершина дерева решений, $D(t)$ — подвыборка, связанная с этой вершиной и $i(t)$ — загрязненность вершины. Естественно потребовать, чтобы загрязненность вершины была равна 0, если $D(t)$ содержит прецеденты только одного класса и была бы максимальной в случае, если $D(t)$ содержит одинаковое число прецедентов каждого класса. Все приведенные ниже меры загрязненности удовлетворяют этому условию. Одной из наиболее используемых является мера загрязненности вершины, основанная на понятии энтропии (entropy impurity):

$$i(t) = -\sum_{i=1}^{c} P(\omega_j) \log_2 P(\omega_j), \qquad (2)$$

где $P(\omega_j)$ есть доля примеров класса ω_j в подвыборке $D(t)$ и полагается $0 \log 0 = 0$. Другой популярной мерой является индекс Джини (Gini) [44], определяемый как

$$i(t) = 1 - \sum_{i=1}^{c} P^2(\omega_j). \qquad (3)$$

Индекс Джини представляет собой частоту ошибочной классификации при случайном назначении меток классов наблюдениям подвыборки $D(t)$. Реже применяется мера загрязненности, основанная на частоте ошибочной классификации (misclassification impurity):

$$i(t) = 1 - \max_j P(\omega_j) \ . \qquad (4)$$

В этом случае $i(t)$ представляет собой частоту ошибочной классификации если все наблюдения выборки $D(t)$ относятся к мажорантному (наиболее часто встречающемуся) классу. Исследования показали [21], что выбор меры загрязненности не оказывает существенного влияния на точность классификации и более важным является выбор критериев остановки и усечения дерева решений, рассматриваемых ниже.

Расщепление вершин

Правило разбиения множества $\mathcal{X}$, связанное с каждой вершиной дерева решений, называется расщеплением (split). Количество подмножеств, на которые разбивается $\mathcal{X}$, в принципе может быть разным для разных вершин, однако, большинство алгоритмов основано на

построении бинарных деревьев, т. е. деревьев, в которых расщепление осуществляется на два подмножества. Это связано с тем, что для любого дерева решений можно построить эквивалентное ему (с точки зрения принимаемых решений) бинарное дерево и, кроме того (что весьма существенно), значительно облегчается программная реализация алгоритма. Бинарное расщепление вершины t можно рассматривать как функцию $f_t : \mathcal{X} \to \{0, 1\}$, $\mathbf{x} \in \mathcal{X}$, где в случае $f(\mathbf{x}) = 0$ вектор $\mathbf{x}$ относится к первому (левому) потомку, а в случае $f(\mathbf{x}) = 1$ – ко второму (правому). Обычно эта функция имеет простой вид и зависит от значений только одного признака. Именно, если некоторый признак x измерен в числовой шкале, то расщепление состоит в выборе x_s, минимизирующего используемую меру загрязненности и определении

$$f_t(\mathbf{x}) = \begin{cases} 0 & \text{при } x < x_s, \\ 1 & \text{при } x \geqslant x_s. \end{cases} \qquad (5)$$

В этом случае, если признак x в выборке принимает m различных значений, то существует $m - 1$ возможное расщепление, сохраняющее порядок значений признака x в подвыборках. Легко видеть, что если все признаки обучающей выборки измерены в числовой шкале и используются только расщепления указанного вида, то области решений будут представлять собой многомерные параллелепипеды, а часть границы области решений, соответствующая данному расщеплению, будет представлять собой часть гиперплоскости, параллельной соответствующей координатной оси в пространстве $\mathbf{R}^n$. Аналогичные расщепления используются и для признаков, измеренных в порядковой шкале.

Если признак x измерен в номинальной шкале и варьирует на уровнях из множества C, то возможные расщепления имеют вид

$$f_t(\mathbf{x}) = \begin{cases} 0 & \text{при } x \in C_1 \\ 1 & \text{при } x \in C_2 \end{cases}, \qquad (6)$$

где C_1 — произвольное непустое подмножество C и $C_2 = C - C_1$.

В работе [110] рассматривались расщепления, определяемые линейными комбинациями значений признаков:

$$f_t(\mathbf{x}) = \begin{cases} 0 & \text{при } \sum_{i=1}^{n} a_i x_i < x_s \\ 1 & \text{при } \sum_{i=1}^{n} a_i x_i \geqslant x_s \end{cases}, \qquad (7)$$

что приводит к так называемым косоугольным деревьям решений (oblique trees). В этом случае минимизация загрязненности осуществляется по параметрам $a_1, a_2, \ldots, a_n$ и x_s. Некоторые алгоритмы построения деревьев решений, использующих для расщепления значения нескольких признаков, представлены в [16, 30, 77, 78]. Пока эти алгоритмы не получили широкого распространения, что, повидимому, связано со сложностью дальнейшей интерпретации получаемых деревьев. Одним из подходов к получению расщеплений (рекомендованный Брейманом), использующий значения нескольких признаков, является метод главных компонент, при котором из исходной обучающей выборки сначала выделяются главные компоненты, а на втором этапе по преобразованной выборке строится дерево решений.

Оптимальное расщепление

Как уже отмечалось, расщепление подвыборки естественно осуществлять таким образом, чтобы максимально уменьшить загрязненность. Уменьшение загрязненности вершины t для бинарных деревьев определяется как

$$\Delta i(t) = i(t) - P_L i(t_L) - P_R i(t_R), \qquad (8)$$

где P_L и P_R есть доли примеров подвыборки $D(t)$, соответствующие левому и правому потомку (t_L и t_R). Наилучшим расщеплением вершины t естественно считать разбиение, которое максимизирует величину $\Delta i(t)$.

В общем случае уменьшение загрязненности определяется как

$$\Delta i(t) = i(t) - \sum_{k=1}^{B} P_k i(t_k), \qquad (9)$$

где B — количество потомков вершины t, P_k — доля примеров подвыборки $D(t)$, соответствующая вершине t_k и $\sum_{k=1}^{B} P_k = 1$. Непосредственное использование этой формулы, однако, приводит к тому, что чаще для расщепления будут выбираться признаки, имеющие больше уровней по сравнению с остальными. Поэтому обычно используется следующая формула

$$\Delta i_B(t) = \frac{\Delta i(t)}{-\sum_{k=1}^{B} P_k \log_2 P_k}, \qquad (10)$$

и выбирается расщепление, максимизирующее величину $\Delta i_B(t)$.

Критерии остановки расщепления

Процесс расщепления вершин имеет естественный предел, когда становится невозможно уменьшить загрязненность очередной вершины. Все такие вершины объявляются терминальными, а соответствующее дерево

полным. Обычно такая ситуация характеризуется тем, что каждая терминальная вершина содержит примеры только одного класса, а само дерево содержит большое количество вершин. Хорошо известно, что полное дерево, как правило, обладает низкой точностью классификации [21]. Это связано с так называемой проблемой переподгонки (overfitting), заключающейся в том, что статистическая модель фактически описывает только саму выборку и непригодна в качестве модели всей генеральной совокупности. В некоторой степени задачу предотвращения построения больших деревьев играют критерии остановки расщепления. Существует несколько способов принятия решения об остановке расщепления. Наиболее простой способ состоит в задании минимального числа для количества наблюдений в подвыборках, соответствующих терминальным вершинам (или минимальной доли наблюдений обучающей выборки). Второй способ заключается в установлении верхнего порога для загрязненности вершины, т. е. задается некоторое число a и расщепление вершины не происходит, если уменьшение загрязненности при расщеплении не превышает a. Третий способ заключается в применении кросс-проверки, а именно, сравнивается количество ошибочно классифицированных наблюдений до и после расщепления вершины и если сокращения ошибок не происходит, то вершина не расщепляется. В случае бинарных деревьев альтернативой приведенным способам является статистический подход [75], основанный на статистике

$$\chi^2 = \sum_{i=1}^{2} \frac{(n_{iL} - n_{ie})^2}{n_{ie}}, \qquad (11)$$

где n_{iL} – количество наблюдений класса ω_i, отнесенных в результате расщепления в левую подвыборку, и n_{ie} – ожидаемое количество наблюдений в левой подвыборке при случайном расщеплении вершины t. Если максимальное значение (по всем возможным расщеплениям) статистики χ^2 не превышает критического значения, соответствующего выбранному уровню значимости, то расщепления не происходит и вершина объявляется терминальной.

Усечение деревьев

Недостатком подхода, основанного на способах остановки расщепления вершин для предотвращения построения больших деревьев, является то, что решение об остановке принимается без учета ситуации, которая могла бы сложиться при продолжении расщепления. Именно, не исключено, что дальнейшее расщепление потомков некоторой вершины могло бы существенно повысить точность классификации. В связи с этим ключевое значение приобрел альтернативный подход, основанный на построении и дальнейшем усечении (pruning) полных деревьев решений [49, 76, 79, 91]. Усечение означает процедуру замены в построенном полном дереве некоторой вершины и связанного с ней поддерева терминальной вершиной. Большинство методов усечения основано на оценке чувствительности поддеревьев по отношению к некоторой мере и удалению поддеревьев, которые оказывают минимальное влияние на эту меру. Эмпирическое сравнение различных методов усечения деревьев решений проведено в [45, 76]. Общим недостатком всех методов усечения относится то, что они не могут гарантировать нахождение оптимального решения, а также высокая вычислительная сложность.

Другим популярным алгоритмом построения деревьев решений является алгоритм C4.5 [92], предшественником которого является алгоритм ID3 [90]. Алгоритм ID3 предполагает, что все признаки обучающей выборки измерены в номинальной шкале. Если обучающая выборка содержит признаки, измеренные в числовой шкале, то они предварительно дискретизуются. Дискретизация означает разбиение области значений непрерывного признака на совокупность непересекающихся интервалов, каждый из которых затем трактуется как уровень номинального признака. Количество подвыборок, на которые расщепляется выборка, соответствуюшая некоторой вершине, равно количеству уровней признака, выбранного для расщепления в этой вершине. В алгоритме C4.5 номинальные признаки используются аналогично ID3, а непрерывные признаки аналогично CART.

Построение деревьев решений по обучающей выборке относится к методам рекурсивного разбиения. Подробное введение в эти методы содержится в работе [105]. В заключение заметим, что наряду с классическими деревьями решений разрабатывается также подход, основанный на теории нечетких множеств и приводящий к построению нечетких деревьев решений (fuzzy decision trees) [33, 112].

Ансамбли классификаторов

Ансамбль классификаторов представляет собой множество классификаторов, чьи решения комбинируются некоторым образом для получения окончательной классификации наблюдений. Обычно синтез решений от-

дельных классификаторов, составляющих ансамбль, осуществляется путем их голосования (возможно, взвешенного). Основная причина, обусловившая интерес к данной тематике, состоит в том, что при определенных условиях ансамбли классификаторов обладают точностью, значительно превосходящей точность отдельных классификаторов и робастны (устойчивы) по отношению к "зашумлению" обучающей выборки. Необходимым и достаточным условием высокой точности ансамбля классификаторов является то, чтобы составляющие его классификаторы были сами достаточно точны и различны (diverse), т. е. совершали ошибки на различных прецедентах [54]. Существует несколько групп методов построения ансамблей классификаторов [42]:

1. Манипулирование примерами обучающей выборки;
2. Манипулирование признаками;
3. Инъекция случайности в индуктивный алгоритм;
4. Манипулирование метками классов;
5. Байесовское голосование.

Случайные леса синтезируют методы первых трех групп, поэтому мы остановимся только на них. Первая группа методов состоит либо в прогоне базового индуктивного алгоритма на различных подвыборках исходной обучающей выборки, либо в итеративном перевзвешивании наблюдений. Наиболее простой способ формирования подвыборок предложен Брейманом [22]. Метод основан на формировании обучающей выборки для каждого классификатора ансамбля с помощью бутстрепа (bootstrap), т. е. случайной выборки (того же объема, что и исходная обучающая выборка) с возвращением из исходной обучающей выборки и использовании метода голосования для агрегирования решений отдельных классификаторов. Метод получил название баггинг (bagging) или агрегированный бутстреп (bootstrap aggregating). Многочисленные экспериментальные исследования (см., например, [20, 22] показали, что использование баггинга существенно повышает точность классификации в случае неустойчивости базового классификатора, когда небольшие возмущения обучающей выборки приводят к существенным изменениям в классификации. Заметим, что деревья решений представляют собой пример неустойчивого классификатора. Теоретический анализ баггинга деревьев решений, проведенный в [15], показал, что баггинг приводит к сокращению средней квадратичной ошибки классификации. Дальнейшие исследования показали, что это верно не для всех базовых моделей классификаторов. К методам этой группы относится также метод, основанный на формировании множества дизъюнктных подвыборок обучающей выборки и использовании методики кросс-проверки [80] для формирования ансамбля классификаторов.

Метод, основанный на итеративном перевзвешивании наблюдений обучающей выборки, называемый бустингом (boosting), был предложен в работе [47]. Идея бустинга состоит в том, что классификаторы ансамбля строятся последовательно и на каждой итерации происходит коррекция (перевзвешивание) наблюдений обучающей выборки (на первой итерации веса всех наблюдений равны). Коррекция осуществляется таким образом, чтобы соответствующий классификатор делал меньше ошибок на тех наблюдениях, на которых часто делали ошибки классификаторы, построенные на предыдущих итерациях алгоритма. Кроме того, каждому классификатору приписывается некоторый вес исходя из количества допущенных им ошибок.

Отметим, что идея последовательной коррекции классификационных алгоритмов для компенсации предыдущих ошибок классификации развивалась и в рамках оптимизационных методов алгебраического подхода к распознаванию образов [2, 3, 9](см., также обзор [4]).

Из методов второй группы отметим работу [32], в которой исходное множество признаков было разбито на несколько дизъюнктных подмножеств и строился ансамбль нейронных сетей, каждая из которых включала признаки только из одного подмножества. К методам этой (и третьей) группы можно отнести метод случайных подпространств [56–58], рассматриваемый ниже.

Методы третьей группы основаны на том, что случайность вводится непосредственно в базовый алгоритм. Так, в работах [40, 41] алгоритм построения деревьев решений, составляющих ансамбль, был рандомизирован следующим образом. На каждом шаге расщепления вычислялось 20 наилучших расщеплений и затем осуществлялся случайный выбор одного из них. Во второй из этих работ было проведено сравнительное экспериментальное исследование трех методов построения ансамблей деревьев решений — бустинга, баггинга и рандомизации на 33 выборках. Результаты показали, что в случае незначительно "зашумленной" обучающей выборки рандомизация срав-

нима (и, возможно, несколько превосходит) с баггингом, но не так точна как бустинг. В случае сильно "зашумленных" выборок баггинг намного точнее бустинга и иногда превосходит рандомизацию.

Экспериментальное сравнение различных алгоритмов построения ансамблей деревьев решений на 57 реальных обучающих выборках было проведено в [14]. Точность решений соответствующих классификаторов оценивалась методом кросс-проверки с дальнейшей проверкой на статистическую значимость. По результатам кросс-проверки алгоритмы ранжировались в порядке уменьшения точности (для каждой выборки). Результаты этих исследований могут быть суммированы следующим образом:

- лучший алгоритм был статистически значимо лучше баггинга только на 8 из 57 выборок;
- проверка средних рангов алгоритмов по всем выборкам показала, что бустинг, случайные леса и рандомизированные деревья статистически значимо точнее баггинга.

В работе [100] было проведено сравнительное экспериментальное исследование двух методов построения ансамблей – баггинга и бустинга. В качестве базового классификатора использовался метод ближайшего среднего (сравнение с эталоном). Исследование показало, что баггинг "работает" намного лучше бустинга в случае, если количество признаков сравнимо с объемом обучающей выборки, а бустинг предпочтительнее баггинга если объем обучающей выборки намного превышает количество признаков. Причиной этого (как установлено в статье) является то, что в случае баггинга разнообразие классификаторов ансамбля уменьшается при увеличении объема обучающей выборки, а в случае бустинга увеличивается.

Случайные леса

Первые работы, связанные с построением ансамблей деревьев решений, были основаны на эвристических процедурах и относятся к началу 90-х гг. прошлого века [68, 97, 98, 115]. Применялся также подход, основанный на бустинге (boosting) деревьев решений [43].

Более строгий подход, названный методом случайных подпространств, был развит в работах Хо [56–58]. Суть подхода заключается в том, что для построения каждого дерева используется только фиксированная доля случайно отобранных признаков. Ансамбли построенных деревьев Хо назвал случайными лесами решений (Random Decision Forests).

В важной для формирования метода случайных лесов работе [12] для построения деревьев было предложено использовать для расщепления каждой вершины деревьев только фиксированную долю случайно отбираемых признаков, т. е. фактически использовать метод случайных подпространств на каждой итерации построения дерева.

И, наконец, Брейманом был предложен метод, известный как случайные леса. Фактически случайный лес Бреймана — это ансамбль деревьев решений, каждое из которых строится на основе бутстреп выборки из исходной обучающей выборки (баггинг) [22], причем для расщепления вершин аналогично [12] используется только доля случайно отбираемых признаков. Кроме того, строится полное дерево (без усечения). Классификация деревьев в ансамбле осуществляется большинством голосов.

Алгоритм индуктивного построения и использования случайного леса впервые был разработан Брейманом и Катлер [35] и реализован в ряде коммерческих пакетов. Вся информация, касающаяся деталей алгоритма, технические доклады, руководства, обзор и примеры применения метода случайных лесов может быть найдена по адресу:
URL: http://www.stat.berkeley.edu/users/breiman/RandomForests.

Алгоритм индукции случайного леса может быть представлен в следующем виде [99]:

1. Для $i = 1, 2, \ldots, B$ (здесь B — количество деревьев в ансамбле) выполнить
   - Сформировать бутстреп выборку S размера l по исходной обучающей выборке $D = \{\mathbf{x}_i, y_i\}_{i=1}^{l}$;
   - По бутстреп выборке S индуцировать неусеченное дерево решений T_i с минимальным количеством наблюдений в терминальных вершинах равным n_{min}, рекурсивно следуя следующему подалгоритму:
     (а) из исходного набора n признаков случайно выбрать p признаков;
     (б) из p признаков выбрать признак, который обеспечивает наилучшее расщепление;
     (в) расщепить выборку, соответствующую обрабатываемой вершине, на две подвыборки;

2. В результате выполнения шага 1 получаем ансамбль деревьев решений $\{T_i\}_{i=1}^{B}$;

3. Предсказание новых наблюдений осуществлять следующим образом:

   (а) для регрессии:

$$\hat{f}_{rf}^{B}(\mathbf{x}) = \frac{1}{B}\sum_{i=1}^{B} T_i(\mathbf{x}) \ ;$$

   (б) для классификации:
   пусть $\hat{\omega}_i(\mathbf{x}) \in \{\omega_1, \omega_2, \ldots, \omega_c\}$ – класс, предсказанный деревом решений T_i, т. е. $T_i(\mathbf{x}) = \hat{\omega}_i(\mathbf{x})$; тогда $\hat{\omega}_{rf}^{B}(\mathbf{x})$ — класс, наиболее часто встречающийся в множестве $\{\hat{\omega}_b(\mathbf{x})\}_{i=1}^{B}$.

Одно из достоинств случайных лесов состоит в том, что для оценки вероятности ошибочной классификации нет необходимости использовать кросс–проверку или тестовую выборку. Оценка вероятности ошибочной классификации случайного леса осуществляется методом "Out–Of–Bag" (OOB), состоящем в следующем [23]. Известно, что каждая бутстреп выборка не содержит примерно 37 % наблюдений исходной обучающей выборки (поскольку выборка с возвращением, то некоторые наблюдения в нее не попадают, а некоторые попадают несколько раз). Классифицируем некоторый вектор $\mathbf{x} \in \mathcal{D}$. Для классификации используются только те деревья случайного леса, которые строились по бутстреп выборкам, не содержащим $\mathbf{x}$, и как обычно используется метод голосования. Частота ошибочно классифицированных векторов обучающей выборки при таком способе классификации и представляет собой оценку вероятности ошибочной классификации случайного леса методом OOB. Практика применения оценки OOB показала, что в случае, если количество деревьев достаточно велико, эта оценка обладает высокой точностью. Если число деревьев мало, то оценка имеет положительное смещение [31].

Случайные леса обладают целым рядом привлекательных качеств, что обусловило их широкое применение, а именно:

1. Случайные леса обеспечивают существенное повышение точности, так как деревья в ансамбле слабо коррелированы вследствие двойной инъекции случайности в индуктивный алгоритм — посредством баггинга и использования метода случайных подпространств при расщеплении каждой вершины;

2. Методически и алгоритмически сложная задача усечения полного дерева решений снимается, поскольку деревья в случайном лесу не усекаются (это также приводит к высокой вычислительной эффективности);

3. Отсутствует проблема переподгонки (даже при количестве признаков, превышающем количество наблюдений обучающей выборки и большом количестве деревьев). Тем самым снимается сложная проблема отбора признаков, необходимая для других ансамблевых классификаторов;

4. Простота применения: единственными параметрами алгоритма являются количество деревьев в ансамбле и количество признаков, случайно отбираемых для расщепления в каждой вершине дерева. Подробные рекомендации по выбору этих параметров можно найти в [69];

5. Легкость организации параллельных вычислений.

Состоятельность случайных лесов

Брейман дает следующее определение случайного леса [25].

Определение 1. Случайным лесом называется классификатор, состоящий из набора деревьев $\{h(\mathbf{x}, \Theta_k)\ k = 1, \ldots\}$, где Θ_k — независимые одинаково распределенные случайные векторы и каждое дерево вносит один голос при определении класса $\mathbf{x}$.

Определение, данное Брейманом, является достаточно общим. В рамках этого определения возможны различные модели случайных лесов, в зависимости от того, каким образом "вводится случайность" в алгоритм индукции деревьев решений [12, 25, 41, 56]. Так, в соответствии с этим определением, случайным лесом является классификатор, состоящий из ансамбля деревьев решений, каждое из которых строится с использованием баггинга. В этом случае Θ_k представляют собой независимые одинаково распределенные l–мерные случайные векторы, координаты которых являются независимыми дискретными случайными величинами, принимающими значения из множества $\{1, 2, 3, \ldots, l\}$ с равными вероятностями. Реализация случайного вектора Θ_k определяет номера прецедентов обучающей выборки, которые образуют бутстреп выборку, используемую при построении k–го дерева решений.

Исследование состоятельности случайных лесов, т. е. вопросов сходимости их решений при неограниченном возрастании объема обучающей выборки, было начато автором метода. В работе [24] исследовались общие вопросы сходимости решений ансамблей классификаторов. Для случайных лесов регрессий исследовалась среднеквадратичная сходимость. Именно, обозначим через $M(Y|\mathbf{X}=\mathbf{x})$ условное математическое ожидание непрерывного отклика при данном значении вектора признаков X. Среднеквадратичная сходимость означает, что

$$\lim_{l\to\infty} |\hat{f}_{rf}(\mathbf{x}) - M(Y|\mathbf{X}=\mathbf{x})|^2 = 0\ . \qquad (12)$$

В случае классификации состоятельность означает сходимость риска (вероятности ошибочной классификации) к байесовскому риску, т. е. риску оптимального байесовского классификатора.

Брейман отмечал, что хотя алгоритм индукции случайных лесов кажется простым, его теоретическая модель сложна для анализа. Поэтому в работе [27] рассматривалась упрощенная модель случайного леса, основанная на предположении, что шаг формирования бутстреп выборок отсутствует и каждая терминальная вершина содержит ровно одно наблюдение. В рамках этой модели Брейманом было показано, что случайный лес регрессий состоятелен и количество признаков, случайно выбираемых для расщепления, не зависит от количества наблюдений. Также для случая двух классов была показана состоятельность соответствующего классификатора. Анализ Бреймана существенно опирался на работу [70], в которой была установлена связь между случайными лесами и адаптивным методом ближайших соседей.

В работе [19] исследовался вопрос состоятельности модели случайного леса, близкой к модели, рассматривавшейся Брейманом [27]. Основной результат работы состоит в том, что модель состоятельна и скорость сходимости зависит только от количества "сильных" признаков и не зависит от количества иррелевантных ("шумовых") признаков.

В работах [17, 18] получен ряд результатов относительно состоятельности нескольких моделей случайных лесов — чисто случайного леса (purely random forest) и инвариантного к масштабу случайного леса (scale-invariant random forest). В основу работы положен результат, полученный авторами, о состоятельности ансамблей классификаторов, основанных на процедуре усреднения рандомизированных базовых классификаторов. Суть результата состоит в том, что для состоятельности ансамбля классификаторов необходимо и достаточно, чтобы базовый классификатор был состоятелен.

Случайные леса как инструмент статистического анализа

Основной недостаток случайных лесов по сравнению с деревьями решений состоит в отсутствии визуального представления процесса принятия решений и сложности интерпретации их решений. Однако предложенные Брейманом меры информативности признаков[4] и возможность использовать случайные леса для построения матрицы близости (proximity) наблюдений более чем компенсируют этот недостаток. Меры информативности дают возможность выделения наиболее информативных признаков, что является одной из важнейших задач статистического анализа. Матрица близости дает возможность применения методов, непосредственно не связанных с классификацией и регрессией, таких как многомерное шкалирование, кластеризация, определение прототипов (т. е. наиболее типичных представителей) классов и выявление аномальных наблюдений. Отмеченные возможности во многом способствовали усилению популярности случайных лесов.

Наиболее используемой сферой применимости случайных лесов (помимо классификации и регрессии) является задача выделения наиболее информативных признаков. Брейманом было предложено 4 меры информативности признаков [25].

Меры информативности признаков, основанные на случайных лесах

Пусть x_i некоторый признак. Три первые меры информативности основаны на оценке влияния случайной перестановки значений этого признака в OOB выборках на результаты классификации.

Первая мера вычисляется следующим образом:

1. Построить случайный лес и получить оценку вероятности ошибочной классификации e методом OOB;

2. В OOB выборках для каждого дерева из построенного случайного леса произвести

[4] Брейман использовал термин "меры важности" (importance measures); автор же придерживается терминологии, принятой в прикладной статистике.

случайную перестановку значений признака x_i;

3. Получить оценку вероятности ошибочной классификации $\hat{e_i}$ по модифицированным OOB выборкам;

4. Определить информативность признака x_i как $I_1(x_i) = \max_i(0, \hat{e_i} - e_i)$.

Вторая и третья меры используют понятие отступа (margin). Пусть $(\mathbf{x}, y)$ — элемент обучающей выборки. Отступ $marg(\mathbf{x}, y)$ определяется как разность между долей деревьев в лесу, правильно классифицирующих, $(\mathbf{x}, y)$ и максимумом из долей деревьев, относящих $(\mathbf{x}, y)$ в другие классы. В результате перестановки значений признака x_i отступ уменьшается. В качестве меры информативности берется среднее значение (по всем наблюдениям) уменьшения отступа, т. е.

$$I_2(x_i) = \frac{1}{l}\sum_{j=1}^{l}(marg(\mathbf{x}_j, y_j) - marg_i(\mathbf{x}_j, y_j),$$

где $marg_i(\mathbf{x}, y)$ означает отступ, вычисленный по OOB выборке со случайной перестановкой значений признака x_i.

Мера 3 равна разности между количеством отступов, которые уменьшились, и количеством отступов, которые увеличились в результате перестановки значений признака x_i:

$$I_3(x_i) = \begin{array}{l}\#[marg(\mathbf{x}, y) > marg_i(\mathbf{x}, y)] - \\ \#[marg(\mathbf{x}, y) < marg_i(\mathbf{x}, y)].\end{array} \quad (13)$$

Мера 4 определяется как среднее уменьшение загрязненности вершин, обусловленное данным признаком, именно

$$I_4(x_i) = \frac{1}{k}\sum_{t}\Delta i(t) I(i, t) \ ,$$

где суммирование осуществляется по всем вершинам деревьев случайного леса, $\Delta i(t)$ — уменьшение загрязненности в вершине t, и $I(i, t)$ — индикаторная функция, равная 1, если признак x_i был выбран для расщепления в вершине t.

Несмотря на многочисленные исследования как теоретического, так и экспериментального характера [51, 101–103], пока неизвестно, какая из предложенных мер предпочтительнее в той или иной ситуации. В работе Бреймана [26] обсуждаются некоторые вопросы, касающиеся различия между мерами информативности и приведен ряд примеров их применения. Подробное изложение вопросов, связанных с мерами информативности, содержится в работе [51].

В работе [101] показано, что описанные выше меры важности признаков не очень надежны в ситуациях, когда признаки измерены в различных шкалах или сильно вырьируют по количеству уровней (для признаков, измеренных в номинальной или порядковой шкале). В этой работе предложено использовать другой алгоритм построения случайного леса, для которого соответствующие меры важности признаков "хорошо работают" в описанных выше ситуациях, а именно, использовать для формирования обучающих выборок используемых для построения деревьев решений не выборку с возвращением (баггинг), а выборку без возвращения объемом равным 0,62 объема исходной выборки. Такой метод получил название подбаггинг (subbagging).

График частной зависимости

Для случайных лесов регрессий возможно построение графиков частных зависимостей отклика от некоторых независимых переменных [55]. Пусть X_S — подвектор входных признаков $x_1, x_2, \ldots, x_n$ с индексами из множества $S \subset \{1, 2, \ldots, n\}$ и C — дополнительное ему множество $S \cup C = \{1, 2, \ldots, n\}$. Оценка функции частной (маргинальной) зависимости отклика от значений подвектора входных признаков X_S определяется как

$$f_S(x_S) = \frac{1}{l}\sum_{i=1}^{l} f_S(x_S, x_{iC}) \ .$$

Эта функция (и соответствующий график) позволяют выделить эффект воздействия подвектора входных признаков X_S на отклик после удаления эффекта воздействия X_C.

Построение матрицы близости

Индукция случайного леса может служить промежуточным этапом для построения матрицы близости (proximity) наблюдений [26]. Элемент (i, j) матрицы близости равен доле деревьев в лесе, таких, что элементы $\mathbf{x}_i$ и $\mathbf{x}_j$ классифицируются в одну терминальную вершину (лист). Ясно, что элементы, часто попадающие в одну терминальную вершину, "подобны" (близки) в некотором смысле. Матрица близости затем может быть использована для:

1. Кластеризации наблюдений;

2. Многомерного шкалирования;

3. Нахождения прототипов классов;

4. Выявления аномальных наблюдений.

Кластеризация наблюдений осуществляется с помощью генерации искусственных данных. Именно исходная выборка, подлежащая кластеризации, образует первый класс, а наблюдения второго класса генерируются искусственным образом. Генерация может осуществляться двумя способами:

- генерируются независимые бутстреп-выборки для каждого признака отдельно;
- значения каждого признака генерируются в соответствии с равномерным распределением на области его значений в исходной выборке.

После формирования такой двухклассовой выборки осуществляется построение случайного леса. Суть подхода состоит в том, что близкие точки исходной выборки будут часто попадать в одинаковые терминальные вершины и поэтому матрица близости может быть использована для кластеризации. Некоторые теоретические результаты такого подхода можно найти в [96], а практические примеры его использования в [26, 69].

Определение прототипов классов производится следующим образом [26]. Для каждого класса находится наблюдение, имеющее наибольшее количество наблюдений этого класса среди своих k ближайших соседей (определяемых по матрице близости). Затем для каждого признака находится медиана этих k ближайших соседей. Эти медианы и считаются прототипами рассматриваемого класса.

Мера аномальности наблюдения определяется как величина, обратная сумме квадратов близостей между этим наблюдением и другими наблюдениями того же класса.

Вопросы методического и практического применения рассмотренных выше методов можно найти в [26, 28].

Разновидности случайных лесов

Случайные леса выживаемости

Обобщением случайных лесов на случай данных, цензурированных справа, важным примером которых являются данные о выживаемости, являются случайные леса выживаемости (Random Survival Forests). С их помощью возможно построение непараметрических регрессионных кривых выживаемости, не предполагающих пропорциональности риска. В R случайные леса выживаемости реализованы в пакете **randomSurvivalForest** [62, 63].

Квантильные леса регрессий

В случае непрерывного отклика случайные леса являются хорошим методом построения непараметрической регрессии, т. е. дают хорошее приближение к условному среднему отклика. В работе [73] предложено важное обобщение – квантильные леса регрессий (Quantile Regression Forests), которые дают существенно более полную картину условного распределения отклика, а именно, возможность непараметрической оценки квантилей условного распределения. Это позволяет, в частности, использовать их для построения доверительных интервалов и обнаружения аномальных значений отклика. В R квантильные леса регрессий реализованы в пакете **quantregForest** [74].

Логические леса

В работе [117] предложен метод построения ансамблей классификаторов, основанных на логических деревьях решений. Теоретической основой является логическая регрессия [88, 89], предполагающая, что все признаки, включая классовый, являются бинарными. Предсказание отклика в логической регрессии осуществляется с использованием логических комбинаций бинарных признаков, позволяя строить хорошо интерпретируемые модели сложных взаимодействий признаков. Однако при наличии шума возможности логической регрессии падают. Кроме классификации логические леса могут использоваться для выявления наиболее информативных признаков и их взаимодействий. В R логические леса реализованы в пакете **logicForest** [118].

Вероятностные случайные леса

Оценка вероятности ошибочной классификации случайных лесов, даваемая методом OOB, является усредненной оценкой [23, 25]. В работе [29] предложен метод и алгоритм построения так называемых вероятностных случайных лесов (Probabilistic Random Forests), с помощью которых возможно получить оценку вероятности ошибочной классификации конкретного наблюдения в случае бинарной классификации. Matlab и C коды, реализующие алгоритм, могут быть загружены с сайта http://ucsu.colorado.edu/ breitenm/.

Потоковые случайные леса

Многие приложения, такие как анализ телекоммуникационных данных, финансовые приложения, анализ интернент протоколов, электронной почты и т. д. основаны на потоках

данных, представляющих собой записи некоторых событий (причем о некоторых известна их принадлежность определенному классу, а о некоторых нет), изменяющихся во времени (и часто с высокой скоростью). В такой постановке классические алгоритмы распознавания образов не могут быть применены непосредственно и требуют определенной (не всегда очевидной) модификации. Основными требованиями к такой модификации являются высокая скорость обработки потока данных, инкрементальность (пошаговость) обучения (т. е. каждая запись должна обрабатываться не более одного раза) и адаптация к текущему распределению событий в потоке. Для случайных лесов одна из таких модификаций, ориентированная на задачи визуального слежения (visual tracking), была предложена в работе [93]. Программная реализация этого алгоритма в пакете "Online Multiclass LPBoost" может быть загружена с сайта http://www.ymer.org/amir/software/online-random-forests/.

Другая модификация, ориентированная на задачи Data Mining и получившая название потоковые случайные леса (Streaming Random Forests), была предложена и разрабатывалась в работах [10, 11].

Случайный наивный Байесовский классификатор и случайная мультиномиальная логит модель

Экспериментальные исследования и практика применения случайных лесов показали, что метод обладает рядом хороших качеств – высокой точностью, робастностью по отношению к защумленности обучающей выборки и отсутствием переподгонки. В связи с этим естественно предположить, что основные элементы алгоритма Бреймана – формирование подвыборок методом баггинга и случайный отбор признаков на стадии расщепления вершин деревьев решений могут быть успешно использованы и для некоторых других базовых классификаторов. Эта идея была реализована в работе [84]. В этой работе в качестве базовых классификаторов рассматривались наивный Байесовский классификатор и мультиномиальная логит модель. Авторами проведен ряд экспериментов по сравнению точности классификации соответствующих ансамблей с точностью случайных лесов и машин опорных векторов, которые показали, что случайный наивный Байесовский классификатор и случайная мультиномиальная логит модель в определенных ситуациях превосходят эти методы.

Программное обеспечение индукции деревьев решений и случайных лесов

Деревья решений и случайные леса реализованы во многих программных средствах как коммерческого назначения, так и свободно доступных. К свободно доступным относится программная система WEKA [53, 116], разрабатываемая с 1993 г. группой машинного обучения университета Вайкато (Новая Зеландия). Сайт проекта находится по адресу http://www.cs.waikato.ac.nz/ml/weka (дата обращения 15.04.2011), с которого можно загрузить систему, руководства по работе с ней и соответствующие публикации. Отметим, что WEKA содержит также большой выбор нерекурсивных процедур распознавания образов и имеет интерфейс с пакетом R, называемый RWEKA [59]. Другой свободно доступной системой является система RapidMiner; сайт соответствующего проекта находится по адресу http://www.rapidminer.com (дата обращения 15.04.2011). В среде MATLAB имеется интерфейс к алгоритму, реализующему метод случайных лесов [113]. В работе [64] предложен свободно доступный C++ код, ориентированный на использование случайных лесов для задач кластеризации. Наиболее широкий спектр процедур, содержащий реализации не только классических алгоритмов CART для деревьев решений и алгоритма Бреймана для случайных лесов содержится в пакетах, разработанных для среды R [85] — свободно распространяемом программном обеспечении для статистических вычислений и графики, доступном на платформах Windows, Linux, Mackintosh. Пакет R может быть загружен с сайта http://cran.r-project.org/, где также имеется руководство по инсталляции и руководства по работе с пакетом. Полезные материалы по пакету R на русском языке могут быть найдены по адресу http://herba.msu.ru/shipunov/software/r/r-ru.htm. Ниже приводятся описания практически всех пакетов, реализующих описанные выше методы и имеющиеся на момент написания настоящего обзора. Эти пакеты также могут быть загружены с сайта http://cran.r-project.org/. Нет сомнений, что их количество будет увеличиваться. Все пакеты, разработанные для среды R, оформлены по единому стандарту — содержат описания всех содержащихся в них процедур и хорошо продуманные примеры их применения, которые могут служить в качестве шаблонов. Ряд пакетов, кроме того, описан в журнале R

Journal (до 2005 г. R News). В частности, один из номеров этого журнала содержит краткий обзор применения метода случайных лесов в среде R [99]. Ниже кратко описаны эти пакеты.

Пакеты **tree** [86] и **rpart** [106] содержат процедуры рекурсивного разбиения для решения задач классификации, регрессии и анализа выживаемости, основанных на идеях классической работы [21]. Адаптация процедур пакета **rpart** на случай многомерного отклика содержится в пакете **mvpart** [36, 37]. Имеется также вспомогательный пакет **maptree** [114], содержащий полезные процедуры для улучшенного графического представления и усечения дендрограмм, деревьев классификации и регрессий, построенных с использованием пакетов **cluster** [72] и **rpart**.

Пакет **rpartOrdinal** [13] содержит процедуры построения деревьев решений, ориентированные на случай, когда классовая переменная измерена в порядковой шкале.

Одним из наиболее мощных пакетов, реализующих алгоритмы рекурсивного разбиения для построения деревьев решений и регрессий, является пакет **party** [60]. Он включает возможность построения деревьев условного вывода (conditional inference trees), лесов условного вывода (conditional inference forests), основанных на теории условного вывода [60], а также процедуры рекурсивного разбиения для параметрических моделей (линейной и обобщенной линейной модели). Допускается использование признаков, измеренных в различных шкалах: числовой, номинальной и порядковой, а также цензурированные и многомерные отклики. Пакет содержит процедуры улучшенного представления бинарных деревьев и распределения отклика в вершинах деревьев. Ряд вопросов, касающихся применения пакета, рассмотрен в работе [104].

Процедуры построения косоугольных деревьев решений содержатся в пакете **oblique.tree** [109].

Пакет **Margin.tree** [108] реализует иерархическую версию метода опорных векторов для построения деревьев решений. Особенно полезен для обработки обучающих выборок с количеством классов большим 2 и когда количество признаков превышает количество наблюдений. Его применение описано в [107].

Пакет **ipred** [82] содержит процедуры построения деревьев решений для задач непрямой (т. е. использующей некоторые типы априорной информации) классификации, регрессии и анализа выживаемости. В пакете реализованы процедуры получения улучшенных оценок вероятности ошибочной классификации и предсказания. Примеры применения этого пакета для решения задач классификации в медицине содержатся в [83].

Основным пакетом, реализующим классический алгоритм случайных лесов Бреймана (и процедуры которого используются в некоторых других пакетах), является пакет **randomForest** [69]. Возможности пакета и вопросы его методического и практического применения рассмотрены в работах [69, 99].

Пакеты **varSelRF** [38, 39] и **Boruta**: [67] содержат процедуры отбора множества наиболее информативных признаков с использованием случайных лесов. В частности, пакет **varSelRF** включает процедуры, основанные на методе обратного исключения переменных и на спектре важности переменных, ориентирован на данные высокой размерности и предусматривает возможность организации параллельных вычислений. Пакет **Boruta** ориентирован на нахождение наиболее информативных признаков в информационных системах. Процедура поиска информативных признаков основана на оригинальном алгоритме, предусматривающем итеративное построение случайных лесов.

Пакет **quantregForest** [73] содержит процедуры построения квантильных лесов регрессий.

Случайные леса выживаемости реализованы в пакете **randomSurvivalForest** [62]. Вопросы применения пакета рассмотрены в [63].

Пакет **LogicForest** [118] содержит процедуры построения логического леса для выявления логических соотношений между откликом и независимыми переменными. Предполагается, что все признаки обучающей выборки являются бинарными.

Пакет **Gbev** [94] ориентирован на построение ансамблей деревьев регрессий методом бустинга для данных с ошибками измерений в независимых переменных.

Пакет **ModelMap** [46] содержит процедуры детального картирования для данных, представленных в виде изображений, основанные на методе случайных лесов и модели стохастического градиентного бустинга.

Заключение

Говоря о периоде, прошедшем со времени возникновения метода случайных лесов, необходимо отметить, что он не только состоялся как метод классификации и регрессии и широко применяется в различных предметных областях, но и послужил основой для разработки

различных его модификаций, важных для приложений. Кроме того, он дал импульс для разработки методов, в основе которых лежит близкая схема, не использующая деревья решений в качестве базового классификатора. Нет сомнений, что в будущем это направление будет развиваться. Представляется, что потенциал метода не исчерпан и с точки зрения решения других статистических задач — нахождения наиболее информативных переменных, кластеризации и т. д. В связи с этим автор считает, что использование этого метода в научных учреждениях различного профиля нашей страны (в частности, в институтах Карельского научного центра РАН) для анализа статистических данных было бы весьма полезно.

Автор выражает признательность Ю. Л. Павлову за предложение написать данный обзор и полезные замечания.

Литература

1. *Айвазян С. А., Бухштабер В. Н., Енюков И. С., Мешалкин Л. Д.* Прикладная статистика. Классификация и снижение размерностей. Справочное издание / Ред. Айвазян С. А. // Финансы и статистика. М., 1989. С. 607.

2. *Воронцов К. В.* О проблемно-ориентированной оптимизации базисов задач распознавания // ЖВМ и МФ. 1998. Т. 38, № 5. С. 870–880. http://www.ccas.ru/frc/papers/voron98jvm.pdf. (дата обращения 15.04.2011).

3. *Воронцов К. В.* Оптимизационные методы линейной и монотонной коррекции в алгебраическом подходе к проблеме распознавания // ЖВМ и МФ. 2000. Т. 40, № 1. С. 166–176. URL: http://www.ccas.ru/frc/papers/voron00jvm.pdf (дата обращения 15.04.2011).

4. *Воронцов К. В.* Обзор современных исследований по проблеме качества обучения алгоритмов // Таврический вестник информатики и математики. 2004. Р. 1–20. URL: www.ccas.ru/frc/papers/voron04twim.pdf (дата обращения 15.04.2011).

5. *Колчин В. Ф.* Случайные отображения. М.: Наука, 1984. 208 с.

6. *Павлов Ю. Л.* Предельные теоремы для числа деревьев заданнного объема в случайном лесе // Математический сборник. 1977. Т. 103, № 3. С. 392–403.

7. *Павлов Ю. Л.* Асимптотическое распределение максимального объема дерева в в случайном лесе // Теория вероятностей и ее применения. 1977. Т. 22, № 3. С. 523–533.

8. *Павлов Ю. Л.* Случайный лес // Вероятность и математическая статистика. М.: Изд. "Большая Российская Энциклопедия", 1999. С. 604–605.

9. *Рудаков К. В., Воронцов К. В.* О методах оптимизации и монотонной коррекции в алгебраическом подходе к проблеме распознавания // Доклады РАН. 1999. Т. 367, № 3. С. 314–317. URL: http://www.ccas.ru/frc/papers /rudvoron99dan.pdf (дата обращения 15.04.2011).

10. *Abdulsalam H., Skillicorn D. B., Martin P.* Streaming Random Forests // Proceedings of the 11th International Database Engineering and Applications Symposium (IDEAS), (September 2007). 2007. P. 225–232.

11. *Abdulsalam H., Skillicorn D. B., Martin P.* Classification Using Streaming Random Forests // IEEE Transactions on Knowledge and Data Engineering. 2011. Vol. 23, N. 1. P. 22–36.

12. *Amit H., Geman D.* Shape quantization and recognition with randomized trees // Neural Computation. 1997. Vol. 9. P. 1545–1588.

13. *Archer K. J.* rpartOrdinal: An R Package for Deriving a Classification Tree for Predicting an Ordinal Response // Journal of Statistical Software. 2010. Vol. 34, N 7. P. 1–17. URL: http://www.jstatsoft.org/v34/i07/ (дата обращения 15.04.2011).

14. *Banfield R. E., Hall L. O., Bowyer K. W., Kegelmeyer W. P.* A Comparison of Decision Tree Ensemble Creation Techniques // IEEE Trans. Pattern analysis and Machine Intelligence. 2007. Vol. 29, N 1. P. 173–180.

15. *Behlmann P., Yu B.* Analyzing bagging // Annals of Statistics. 2002. Vol. 30. P. 927–961.

16. *Bennet K. P.* Decision tree construction via linear programming // Proceedings of the 4th Midvest Artificial Intelligence and Cognitive Science Society Conference. (Utica, Illinois 1992). 1992. P. 97–101.

17. *Biau G., Devroye L., Lugosi G.* Consistency of random forests and other averaging classifiers // Journal of Machine Learning Research. 2008. Vol. 9. P. 2015–2033.

18. *Biau G. Devroye L.* On the layered nearest neighbour estimate, the bagged nearest neighbour estimate and the random forest method in regression and classification // Journal of Multivariate Analysis. 2010. Vol. 101. P. 2499–2518. URL: http://www.lsta.upmc.fr/BIAU/bd4.pdf (дата обращения 15.04.2011).

19. *Biau G.* Analysis of a Random Forests Model // Technical report, University Paris 6. 2010. P. 1–31. URL: http://hal.archives-ouvertes.fr/docs/00/47/65/45/PDF/article2.pdf (дата обращения 15.04.2011).

20. *Borra S., Ciaccio A.* Improving nonparametric regression methods by bagging and boosting // Computational Statistics and Data Analysis. 2002. Vol. 38, N 4. P. 407–420.

21. *Breiman L., Friedman R., Olshen R., Stone C.* Classification and Regression Trees. Belmont, California: Wadsworth International, 1984. 342 p.

22. *Breiman L.* Bagging predictors // Machine Learning. 1996. Vol. 24, N 2. P. 123–140.

23. *Breiman L.* Out-of-bag estimation // Technical report, Statistics Department University of California, Berkeley. 1996. P. 1–13. URL: ftp://ftp.stat.berkeley.edu/pub/users /breiman/OOBestimation.ps.Z (дата обращения 15.04.2011).

24. *Breiman L.* Some infinite theory for predictor ensembles // Technical Report 577, Statistics Department University of California, Berkeley, 2000. P. 1–30. URL: http://www.stat.berkeley.edu/breiman (дата обращения 15.04.2011).

25. *Breiman L.* Random forests // Machine Learning. 2001. Vol. 45, N 1. P. 5–32.

26. *Breiman L.* Manual on setting up, using, and understanding random forests v3.1. 2002. URL: http://oz.berkeley.edu/users/breiman/ Using_random_forests_V3.1.pdf.

27. *Breiman L.* Consistency for a simple model of random forests // Technical Report 670, Statistics Department, UC Berkeley. 2004. P. 1–10. URL: http://www.stat.berkeley.edu/breiman (дата обращения 15.04.2011).

28. *Breiman L., Cutler A.* Random Forests. Berkeley. 2005. 56 p. URL: http://www.stat. berkeley.edu/users /breiman/RandomForests

29. *Breitenbach M., Grudic G., Nielsen R.* Probabilistic Random Forests: Predicting Data Point Specific Misclassification Probabilities // Machine Learning. 2009. Vol. 23, N 2. P. 48–73.

30. *Brodley C. E., Utgoff P. E.* Multivariate decision trees // Machine Learning. 1995. Vol. 19, N 1. P. 45–77.

31. *Bylander T.* Estimating generalization error on twoclass datasets using outofbag estimates // Machine Learning. 2002. Vol. 48. P. 287–297.

32. *Cherkauer K. G.* Human expert–level performance on a scientific image analysis task by a system using combined artificial neural networks / Eds. P. Chan // Working Notes on the AAAI Workshop on Integrating Multiple Learned Models. 2006. P. 15–21. URL: http://www.cs.fit.edu/ imlm (дата обращения 15.04.2011).

33. *Cios K. G., Sztandera L. M.* Continuous ID3 algorithm with fuzzy entropy measures // Proceedings of IEEE International Conference on Fussy Systems. 1992. P. 469–476.

34. *Cutler D. R., Edwards T. C. et al.* Random forests for classification in ecology // Ecology. 2007. Vol. 88, N. 11. P. 2783–2792.

35. *Cutler A., Breiman L.* RAFT: RAndom Forest Tool. URL: http://www.stat.berkeley.edu/ users/breiman/RandomForests/ (дата обращения 15.04.2011).

36. *De'ath G.* Multivariate Regression Trees: A New Technique for Constrained Classification Analysis // Ecology. 2002. Vol. 83, N 4. P. 1103–1117.

37. *De'ath G.* mvpart: Multivariate partitioning // R package version 1.31. 2010. URL: http://CRAN.Rproject.org/package=mvpart (дата обращения 15.04.2011).

38. *Diaz-Uriarte R.* varSelRF: Variable selection using random forests // R package version 0.71. 2009. URL: http://ligarto.org/rdiaz/Software /Software.html (дата обращения 15.04.2011).

39. *Diaz-Uriarte R., Andrя́es S.* Gene Selection and classification of microarray data using random forest // BMC Bioinformatics. Vol. 7, N 3. P. 1–13.

40. *Dietterich T. G., Kong E. B.* Machine learning bias, statistical bias, and statistical variance of decision tree algorithms // Technical report, Department of computer science Oregon State University. 1995. P. 1–22. URL: ftp://ftp.cs.orst.edu/pub/tgd/papers/tr-bias.ps.gz (дата обращения 15.04.2011).

41. *Dietterich T. G.* An Experimental Comparison of Three Methods for Constructing Ansembles of Decision Trees: Bagging, Boosting, and Randomization // Machine Learning. 1999. P. 1–20.

42. *Dietterich T. G.* Ensemble methods in machine learning // First International Workshop on Multiple Classifier Systems. Lecture Notes in Computer Science. New York: Springer, 2000. P. 1–15.

43. *Drucker H., Cortes C.* Boosting decision trees / Eds. D. Touretsky, M. Mozer et al. // Advances in Neural Information Processing Systems. Cambridge: MA. MIT Press, 1996. Vol. 8. P. 479–485.

44. *Duda R. O., Hart P. E., Stork D. G.* Pattern Classification. NY.: John Wiley Sons, 2001. 639 p.

45. *Esposito F., Malerba D., Semeraro G.* A comparative analysis of methods for pruning decision trees // IEEE Transactions on Pattern Analysis and Machine Intelligence. 1997. Vol. 19, N 5. P. 476–491.

46. *Freeman E., Frescino T.* ModelMap: Modeling and Map production using Random

Forest and Stochastic Gradient Boosting // USDA Forest Service, Rocky Mountain Research Station, 507 25th street, Ogden, UT, USA. 2009. P. 134–156.

47. *Freund Y., Shapire R. E.* Experiments with a new boosting algorithm // Proceedings of the 13rd International Conference on Machine Learning. P. Morgan Cauffman. 1996. P. 146–148.

48. *Freund Y., Shapire R. E.* Discussion of the paper "Arcing classifiers" by Leo Breiman // The Annals of Statistics. 1998. Vol. 26, N 3. P. 824–832.

49. *Furnkranz J.* Pruning algorithms for rule learning // Machine Learning. 1997. Vol. 27. P. 139–172.

50. *Gehrke J., Ramakrishnan R., Ganti V.* Rainforest a framework for fast decision tree construction of large datasets // Data Mining and Knowledge Discovery. 2000. Vol. 4, N 2/3. P. 127–162.

51. *Genuer R., Poggi J.-M., Tuleau C.* Random Forests: Some methodological insights // Research Report 6729, INRIA Saclay-Ile-de-France. 2008. P. 1–35. URL: http://hal.inria.fr/inria-00340725/fr (дата обращения 15.04.2011).

52. *Genuer R.* Risk bounds for purely uniformly random forests // Technical report 7318, INRIA. 2010. P. 1–22. URL: http://www.math.u-psud.fr/ genuer/genuer.purf.pdf (дата обращения 15.04.2011).

53. *Hall M., Frank E., Holmes G. et al.* The WEKA Data Mining Software: An Update // SIGKDD Explorations. 2009. Vol. 11, N 1. P. 1–9.

54. *Hansen L., Salamon P.* Neural Network Ensembles // IEEE Trans. Pattern analysis and Machine Intelligence. 1990. Vol. 12. P. 993–1001.

55. *Hastie T., Tibshirani R., Friedman J.* The Elements of Statistical Learning. Springer, 2001. 533 p.

56. *Ho T. K.* Random Decision Forests // Proceedings of the 3rd International Conference on Document Analisis and Recognition, (Montreal,Canada 1995), 1995. P. 278–282.

57. *Ho T. K.* C4.5 Decision Forests // Proceedings of the 14rd International Conference on Pattern Recognition. (Brisbane, Australia, 1998). 1998. P. 17–20.

58. *Ho T. K.* The Random Subspace Method for Constructing Decision Forests // IEEE Transactions on Pattern Analysis and Machine Intelligence. 1998. Vol. 20, N 8. P. 832–844.

59. *Hornik K., Buchta C., Zeileis A.* Open-Source Machine Learning: R Meets Weka // Computational Statistics. 2009. Vol. 24, N. 2. P. 225–232.

60. *Hothorn T., Hornik K., Zeileis A.* Unbiased Recursive Partitioning: A Conditional Inference Framework // Journal of Computational and Graphical Statistics. 2006. Vol. 15, N 3. P. 651–674. URL: http://statmath.wuwien.ac.at/ zeileis/papers/ Hothorn+Hornik+Zeileis2006.pdf. (дата обращения 15.04.2011).

61. *Hothorn T., Buehlmann P., Dudoit S. et al.* Survival Ensembles // Biostatistics. 2006. Vol. 7, N 3. P. 355–373.

62. *Ishwaran H., Kogalur U.* Random survival forests for R // R News. 2007. Vol. 7, N 2. P. 25−31.

63. *Ishwaran H., Kogalur U., Blackstone E., Lauer M.* Random survival forests // Ann. Appl. Statist. 2008. Vol. 2, N 3. P. 841–860.

64. *Karpievitch, Y. V., Leclerc, A. P., Hill, E. G., Almeida, J. S.* RF++: Improved Random Forest for Clustered Data Classification. URL: http://www.ohloh.net/p/rfpp. (дата обращения 15.04.2011).

65. *Kooperberg C., Ruczinski I., LeBlanc M., Hsu L.* Sequence Analysis using Logic Regression // Genetic Epidemiology. 2001. Vol. 21. P. 626–631. URL: http://kooperberg.fhcrc.org/logic/documents/ingophdlogic.pdf (дата обращения 15.04.2011).

66. *Kuncheva L. I.* Combining Pattern Classifiers: Methods and Algorithms / Hoboken, New Jersey: John Wiley Sons, 2004. 349 p.

67. *Kursa M. B., Rudnicki W. R.* Feature Selection with the Boruta Package // Journal of Statistical Software. 2010. Vol. 36, N 11. P. 1–13. URL: http://www.jstatsoft.org/v36/i11/ (дата обращения 15.04.2011).

68. *Kwok S. W., Carter C.* Multiple decision trees // Uncertainty in Artifcial Intelligence. 1990. Vol. 4. P. 327–335.

69. *Liaw A., Wiener M.* Classification and Regression by randomForest // R News. 2002. Vol. 2, N 3. P. 18–22. URL: http://CRAN.R-project.org/doc/Rnews/

70. *Lin Y., Jeon Y.* Random forests and Adaptive Nearest Neighbors // Technical report 1055. Dept. Statistics, Univ. Wisconsin, 2002. P. 1–31.

71. *Lin Y., Jeon Y.* Random forests and adaptive nearest neighbors // Journal of the American Statistical Association. 2006. Vol. 101. P. 578–590.

72. *Maechler M.* cluster: Cluster Analysis Extended Rousseeuw et al. // R package version 1.13.3. 2011. URL: http://CRAN.Rproject.org/package=cluster (дата обращения 15.04.2011).

73. *Meinshausen N.* Quantile Regression Forests // Journal of Machine Learning Research. 2006. Vol. 7. P. 983–999.

74. *Meinshausen N.* quantregForest: Quantile Regression Forests // R package version 0.22. 2007. URL: http://www.stat.berkeley.edu/ nicolai (дата обращения 15.04.2011).

75. *Mingers J.* Expert systems experiments with rule induction // Journal of the Operational Research Society. 1987. Vol. 38. P. 39–47.

76. *Mingers J.* An empirical comparison of pruning methods for decision tree induction // Machine Learning. 1989. Vol. 4. P. 227–243.

77. *Murthy S. K.* Automatic Construction of Decision Trees from Data: A MultiDisciplinary Survey // Data Mining and Knowledge Discovery. 1998. Vol. 2. P. 345–389.

78. *Murthy S., Kasif F., Salzberg S., Beigel R.* OCI: Randomized induction of oblique decision trees // Proceedings of the Eleventh National Conference on Artificial Intelligence, Mit Press, Boston. 1993. P. 322–327.

79. *Niblett T., Bratko I.* Learning decision rules in noisy domains / Eds. M. A. Bramer // Research and development in Expert Systems. Cambridge University Press, 1986. P. 25–34.

80. *Parmanto B., Munro P. W., Doyle H. R.* Improving committee diagnosis with resumpling technique // Advances in Neural Information Processing Systems. 1996. Vol. 8. P. 882–888.

81. *Pavlov Yu. L.* Random Forests. Utrecht, Boston, Koln, Tokyo: VSP, 2000. 122 p.

82. *Peters A., Hothorn T.* ipred: Improved Predictors // R package version 0.8-11. 2011. URL: http://CRAN.Rproject.org/package=ipred (дата обращения 15.04.2011).

83. *Peters A., Hothorn T., Lausen B.* ipred: Improved Predictors // R News. 2002. Vol. 2, N 2. P. 33–36. URL: http://CRAN.R-project.org/doc/Rnews/ (дата обращения 15.04.2011).

84. *Prinzie A., Poel D.* Random Multiclass Classification: Generalizing Random Forests to Random MNL and Random NB // Working paper, Department of Marketing, Ghent University, 2007. P. 1–12.

85. *R Development Core Team* R: A language and environment for statistical computing // R Foundation for Statistical Computing: Vienna, Austria, ISBN 3-900051-07-0, URL: http://www.R-project.org (дата обращения 15.04.2011).

86. *Ripley B. D.* tree: Classification and regression trees // R package version 1.026. 2007. URL: http://CRAN.Rproject.org/package=tree (дата обращения 15.04.2011).

87. *RobnikSikonja M.* Improving Random Forests / Eds. J. F. Boulicaut et al. ECML 2004, LNAI 3210, Berlin: Springer, 2004. P. 359–370. URL: http://lkm.fri.unilj.si/rmarko/papers/ (дата обращения 15.04.2011).

88. *Ruczinski I., Kooperberg C., LeBlanc M.* Logic Regression methods and software / Eds: D. Denison et al. // Proceedings of the MSRI workshop on Nonlinear Estimation and Classification. New York: Springer, 2002. P. 333–344.

89. *Ruczinski I., Kooperberg C., LeBlanc M.* Logic Regression // Journal of Computational and Graphical Statistics. 2003. Vol. 12. P. 475–511.

90. *Quinlan J. R.* Learning efficient classification procedures and their application to chess end games / Eds. R. S. Michalski et al. // Machine Learning: An artificial intelligence approach. San Francisko: Morgan Kaufmann, 1983. P. 463–482.

91. *Quinlan J. R.* Simplifying decision trees // International Journal of ManMachine Studies. 1987. Vol. 27. P. 221–234.

92. *Quinlan J. R.* C4.5 Programs for Machine Learning. San Mateo, California: Morgan Kaufmann, 1993.

93. *Saffari A., Leistner C., Jakob Santner J. et al.* On-line Random Forests // 3rd IEEE ICCV Workshop on On-line Computer Vision, 2009. P. 112–127.

94. *Sexton J.* gbev: Gradient Boosted Regression Trees with Errors in Variables / R package version 0.1.1. 2009. URL: http://CRAN.Rproject.org/package=gbev (дата обращения 15.04.2011).

95. *Shapire R., Freund Y., Bartlett P., Lee W.* Boosting the margin: A new explanation for the effectiveness of voting methods // Annals of Statistics. 1998. Vol. 26, N 5. P. 1651–1686.

96. *Shi T., Horvath S.* Unsupervised Learning with Random Forest Predictors // Journal of Computational and Graphical Statistics. 2006. Vol. 15, N 1. P. 118–138.

97. *Shlien S.* Multiple Binary Decision Tree Classifiers // Pattern Recognition. 1990. Vol. 23, N 7. P. 757–763.

98. *Shlien S.* Nonparametric classification using matched binary decision trees // Pattern Recognition Letters. 1992. Vol. 13. P. 83–87.

99. *Siroky D.* Navigating Random Forests and related advances in algorithmic modeling // Statistics Surveys. 2009. Vol. 3. P. 147–163.

100. *Skurichina M., Kuncheva L., Duin R.* Bagging and boosting for the nearest mean classifier: Effects of sample size on diversity and accuracy / Eds. J. K. F. Roli // Multiple Classifier Systems, Proc. Third International

Workshop MCS. (Cagliari, Italy 2002). Berlin: Springer, 2002. Vol. 2364. P. 62–71. http://citeseer.nj.nec.com/539135.html

101. *Strobl C., Boulesteix A. L., Augustin T., Zeileis A.* Bias in RandomForest Variable Importance Measures: Illustrations, Sources and a Solution // BMC Bioinformatics. 2007. Vol. 8, N 25. URL: http://www.biomedcentral.com/ 14712105/8/25/abstract. (дата обращения 15.04.2011).

102. *Strobl C., Boulesteix A. L., Augustin T., Zeileis A.* Conditional variable importance for Random Forests // BMC Bioinformatics. Vol. 9, N 307. P. 67–78. http://www.stat.uni-muenchen.de/ carolin/research.html (дата обращения 15.04.2011).

103. *Strobl C., Zeileis A.* Danger: High Power! Exploring the Statistical Properties of a Test for Random Forest Variable Importance // Technical Report 017. Department of Statistics, University of Munich, 2008. P. 1–9. URL: http://www.stat.uni-muenchen.de (дата обращения 15.04.2011).

104. *Strobl C., Hothorn T., Zeileis A.* Party On! A New, Conditional Variable-Importance Measure for Random Forests Available in the party Package // R Journal. 2009. Vol. 1, N 2. P. 14–17.

105. *Strobl C., Malley J., Tutz G.* An Introduction to Recursive Partitioning: Rationale, Application and Characteristics of Classification and Regression Trees, Bagging and Random Forests // Technical Report 55. Department of Statistics University of Munich, 2009. P. 1–50. URL: http://www.stat.uni-muenchen.de (дата обращения 15.04.2011).

106. *Therneau T., Atkinson B.* rpart: Recursive Partitioning // R package version 3.145. 2009. URL: http://CRAN.Rproject.org/package=rpart (дата обращения 15.04.2011).

107. *Tibshirani R., Hastie T.* Margin trees for highdimensional classification // Technical report. Stanford University, 2006. P. 1–21. URL: http://www.stanford.edu/ hastie/pub.htm (дата обращения 15.04.2011).

108. *Tibshirani R.* marginTree: margin trees for highdimensional classification // R package version 1.01. 2010. URL: http://CRAN.Rproject.org/package=marginTree (дата обращения 15.04.2011).

109. *Truong A.* oblique.tree: Oblique Trees for Classification Data // R package version 1.0. 2009. URL: http://CRAN.Rproject.org/package =oblique.tree (дата обращения 15.04.2011).

110. *Truong A.* Fast Growing and Interpretable Oblique Trees via Probabilistic Model // Univ. of Oxford, A thesis submitted for the degree of Doctor of Philosophy, Trinity term. 2009. P. 1–119.

111. *Wang X., Chen B., Qian G., Ye F.* On the optimization of fuzzy decision trees // Fussy Sets and Systems. 2000. Vol. 11, N 2. P. 117–125.

112. *Wang L. X., Mendel J. M.* Generating fuzzy rules by learning from examples // IEEE Transaction on Systems, Man and Cybernetics. 1992. Vol. 22. P. 1414–1427.

113. *Wang T.* MATLAB R13. // URL: http://lib.stat.cmu.edu/matlab (дата обращения 15.04.2011).

114. *White D.* maptree: Mapping, pruning, and graphing tree models // R package version 1.4-6. 2010. http://CRAN.R-project.org/package=maptree

115. *Williams G. J.* Combining decision trees: Initial results from the MIL algorithm / Eds. J. S. Gero, R. B. Stanton // Artificial Intelligence Developments and Applications. NorthHolland: Elsevier Science Publishers, 1988. P. 273–289.

116. *Witten I., Frank E.* Data Mining: Practical Machine Learning Tools and Techniques. 2nd Edition. San Francisco: Morgan Kaufmann, 2005. 567 p.

117. *Wolf B. J., Slate E. H., Hill E. G.* Logic Forest: An ensemble classifier for discovering logical combinations of binary markers // Bioinformatics. 2010. Vol. 26, N 17. P. 2183–2189.

118. *Wolf B. J.* LogicForest: Logic Forest // R package version 1.0. 2010. URL: http://CRAN.Rproject.org/package=LogicForest (дата обращения 15.04.2011).

119. *Zeileis A., Hothorn T.* ModelBased Recursive Partitioning // Journal of Computational and Graphical Statistics. 2008. Vol. 17, N 2. P. 492–514.

СВЕДЕНИЯ ОБ АВТОРЕ:

Чистяков Сергей Павлович
младший научный сотрудник
Институт прикладных математических исследований Карельского научного центра РАН
ул. Пушкинская, 11, Петрозаводск, Республика Карелия, Россия, 185910
эл. почта: chistiakov@krc.karelia.ru
тел.: (8142) 763370

Chistiakov, Sergey
Institute of Applied Mathematical Research, Karelian Research Centre, Russian Academy of Sciences
11 Pushkinskaya St., 185910 Petrozavodsk, Karelia, Russia
e-mail: chistiakov@krc.karelia.ru
tel.: (8142) 763370