

Классификация на основе композиции случайных лесов и параллельных нейронных сетей

Л. В. Уткин¹, М. А. Рябинин²

Санкт-Петербургский политехнический университет
Петра Великого

¹lev.utkin@gmail.com, ²mihail-ryabinin@yandex.ru

К. Д. Жук

Санкт-Петербургский государственный университет
аэрокосмического приборостроения
zhuk_kd@mail.ru

Ю. А. Жук

Санкт-Петербургский государственный лесотехнический университет
zhuk_yua@mail.ru

Аннотация. В работе представлена новая композиция случайных лесов и параллельных нейронных сетей для решения задач классификации. Нейронные сети используются в качестве нелинейной функции, комбинирующей распределения вероятностей классов на выходе каждого дерева решений в случайном лесу. Особенность структуры нейронных сетей заключается в том, что для обработки вероятностей одного класса используется отдельная сеть. Нейронные сети для всех классов характеризуются тем, что имеет одинаковые параметры (веса соединений), что является обобщением сямских сетей. Результаты тестирования композиции показали, что использование нейронных сетей существенно повышает точность классификации.

Ключевые слова: случайные леса; нейронные сети; классификация; сямская нейронная сеть

1. ВВЕДЕНИЕ

Несмотря на успех большого числа современных моделей классификации, включая модели глубокого обучения, до сих пор композиционные модели остаются одними из наиболее эффективных для решения многих прикладных задач. Композиционные модели обучения основаны на построении множества классификаторов для обучения и на комбинировании их прогнозов в соответствии с определенным правилом. Это правило играет решающую роль в достижении требуемых характеристик итоговой модели.

Существует целый ряд обзорных статей, посвященных различным подходам на основе комбинирования классификаторов. Детальный анализ большинства методов можно найти, например, в работах [7, 16, 17].

Можно выделить три основных подхода к комбинированию классификаторов [17]: бэггинг, стекинг и бустинг. Каждый подход имеет свои плюсы и минусы. Его эффективность зависит от анализируемой обучающей выборки и прикладной задачи.

Одним из широко используемых и демонстрирующих чрезвычайно высокую эффективность методов бэггинга является случайный лес [2, 3]. Это классификатор, состоящий из набора деревьев решений. Согласно основному алгоритму построения случайного леса, определенное количество обучающих элементов и признаков случайным образом выбираются из всего обучающего множества с возвращением, чтобы построить отдельное дерево решений в лесу. Модели случайного леса успешно использовались в различных практических задачах. Подробное описание многих таких приложений и свойств случайных лесов были рассмотрены рядом авторов [1, 5, 8, 12, 13, 14].

Во многих моделях композиции используется среднее или средневзвешенное значение данных на выходах слабых классификаторов. При этом вес каждого слабого классификатора можно рассматривать как его вклад в итоговое решение. Однако существуют подходы к построению правила комбинирования, отличающиеся от подхода с использованием средневзвешенных значений, поскольку средневзвешенные значения имеют некоторые недостатки. Прежде всего, взвешенное среднее является линейным. С одной стороны, линейное правило простое и явно показывает вклад каждого слабого классификатора. С другой стороны, во многих случаях мы не можем представить сложное взаимодействие слабых классификаторов, особенно, когда функция потерь сильного классификатора (итоговая модель композиции) не является тривиальной.

В базовой модели случайных лесов также используется стандартное усреднение данных на выходах деревьев решений. Его можно рассматривать как взвешенное усреднение с одинаковыми весами деревьев.

Следует отметить, что идея взвешивания в случайных лесах не является новой. Многие реализации случайных лесов используют веса классов для работы с несбалансированными наборами данных [6]. В то же время существует множество публикаций, посвященных более сложным правилам назначения весов деревьев. В частности, в работах [9, 10] предлагается назначать веса

Исследование выполнено за счет гранта Российского научного фонда (проект № 18-11-00078)

деревьям решений в соответствии с их классификационной способностью. Главная особенность этих методов заключается в том, что они используют некоторые показатели качества классификации для назначения весов. Более того, эти показатели получены на основе данных тестирования. В результате вес в этом случае является настраиваемым параметром. В отличие от этого подхода в работе [15] предложено использовать веса деревьев в качестве обучающих параметров. Предлагаемый подход позволяет более гибко выбирать схему распределения весов, используя различные функции потерь для вычисления оптимальных весов.

Несмотря на преимущества весового комбинирования деревьев в случайном лесу, их использование оставляет нас в классе линейных функций комбинирования. Поэтому, чтобы преодолеть эту трудность, мы предлагаем использовать нейронную сеть специальной формы для вычисления векторов вероятностей классов. Нейронная сеть играет роль нелинейного аналога линейной функции весов. Важно отметить, что в этом случае нет весов деревьев решений. Однако веса соединений нейронной сети можно рассматривать как замену весов деревьев. Предлагаемая нейронная сеть не является стандартной. Она состоит из нескольких параллельных сетей с общими одинаковыми параметрами и может рассматриваться как обобщение известной сиамской нейронной сети [4]. При этом число параллельных сетей совпадает с числом классов в рассматриваемой задаче классификации. В частности, если мы имеем дело с бинарной классификацией, то получим стандартную сиамскую нейронную сеть.

II. КРАТКОЕ ОПИСАНИЕ СЛУЧАЙНЫХ ЛЕСОВ

Прежде всего, мы рассмотрим формальную постановку задачи классификации. Пусть имеется обучающая выборка $S = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$, содержащая n обучающих примеров, в которых вектор $\mathbf{x}_i \in \mathbf{R}^m$ представляет собой вектор признаков, состоящий m элементов, и $y_i \in \{1, \dots, C\}$ представляет собой метки классов соответствующих примеров. Задача классификации заключается в построении классификатора $c: \mathbf{R}^m \rightarrow \{1, \dots, C\}$, который максимизирует вероятность того, что $c(\mathbf{x}_i) = y_i$ для всех $i = 1, \dots, n$.

Один из известных и популярных классификаторов – дерево решений, которое состоит из множества вершин решений и множества листьев (конечных вершин). При этом, на каждой вершине решений осуществляется расщепление данных и переход на левую или правую вершины решения, являющиеся наследниками исходной расщепляющей вершины.

Для листьев дерева можно определить распределение вероятностей классов $p = (p_1, \dots, p_C)$. Оценка распределения p осуществляется вычислением доли обучающих примеров из разных классов среди всех примеров, которые попали в заданную вершину (лист).

Предположим, что случайный лес состоит из T деревьев решений. Тогда распределение классов для случайного леса, образованное i -ым обучающим примером \mathbf{x}_i и обозначенное как $\mathbf{v}_i = (v_{i,c}, c = 1, \dots, C)$, вычисляется усреднением всех распределений вероятностей классов p деревьев, т.е., выполняется равенство

$$\mathbf{v}_{i,c} = T^{-1} \cdot \mathbf{p}_{i,c} \cdot \mathbf{1}^T.$$

Здесь $p_{i,c}^{(t)}$ – вероятность класса c для примера \mathbf{x}_i , образованная t -ым деревом, и $\mathbf{p}_{i,c} = (p_{i,c}^{(t)}, t = 1, \dots, T)$; $\mathbf{1}$ – единичный вектор размерности T .

Если предположить, что все деревья в случайном лесу имеют веса, обозначенные как $\mathbf{w} = (w_t, t = 1, \dots, T)$, то распределение вероятностей классов случайного леса вычисляется как весовое среднее распределений вероятностей классов деревьев:

$$\mathbf{v}_{i,c} = \mathbf{p}_{i,c} \cdot \mathbf{w}.$$

Иллюстрация весового среднего показана на рис. 1, где три дерева имеют распределения вероятностей классов $p^{(1)} = (0.6, 0.4)$, $p^{(2)} = (0.2, 0.8)$, $p^{(3)} = (1.0, 0.0)$. Предполагается, что пример \mathbf{x}_i попадает в заштрихованные вершины. На рис. 1 показано, что количество примеров первого класса (малые круги) и второго класса (треугольники) попадают в заштрихованную вершину первого дерева равно 3 и 2 соответственно. Отсюда распределение вероятностей классов для этой вершины имеет вид $p^{(1)} = (0.6, 0.4)$. Аналогичным образом можно получить распределения вероятностей классов для других вершин и деревьев. Можно заметить из рис. 1, что вероятности классов $v_{i,c}$, $c = 1, \dots, C$, всего случайного леса определяются как весовые суммы, т.е., справедливы равенства

$$v_{i,1} = 0.6w_1 + 0.2w_2 + 1.0w_3,$$

$$v_{i,2} = 0.4w_1 + 0.8w_2 + 0.0w_3.$$

Веса ограничены следующим очевидным условием:

$$\mathbf{w} \cdot \mathbf{1}^T = 1, w_t \geq 0, t = 1, \dots, T.$$

Важно отметить, что веса не зависят от каждого \mathbf{x}_i . Кроме того, они не зависят от классов. Они определяются всеми примерами в среднем и уникальны для каждого дерева.

Для вычисления оптимальных весов деревьев решения необходимо определить целевую функцию задачи оптимизации поставленной в соответствии с общей целью классификации. Целевая функция для вычисления оптимальных весов определяется как евклидово расстояние между вектором классов и таким вектором, сто его элемент с индексом y_i равен 1, а другие элементы равны 0. Другими словами, необходимо найти такие веса, что среднее расстояние между метками классов из

обучающей выборки и вычисленными вероятностями классов стремилось бы к минимуму. Теперь можно записать целевую функцию для вычисления оптимальных весов следующим образом:

$$J(\mathbf{w}) = \min_{\mathbf{w}} \sum_{i=1}^n \|\mathbf{v}_i - \mathbf{o}_i\|_2^2 + \lambda R(\mathbf{w}).$$

Здесь $R(\mathbf{w})$ – регуляризационное слагаемое, используемое для ограничения возможного множества решений; λ – параметр, который управляет степенью влияния регуляризации на решение, $\mathbf{o}_i = (0, \dots, 0, 1_c, 0, \dots, 0)$.

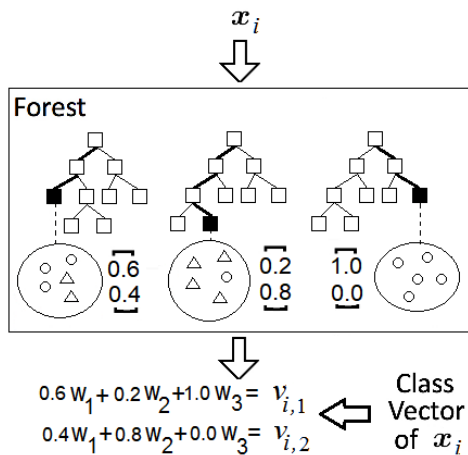


Рис. 1. Пример определения весовых средних

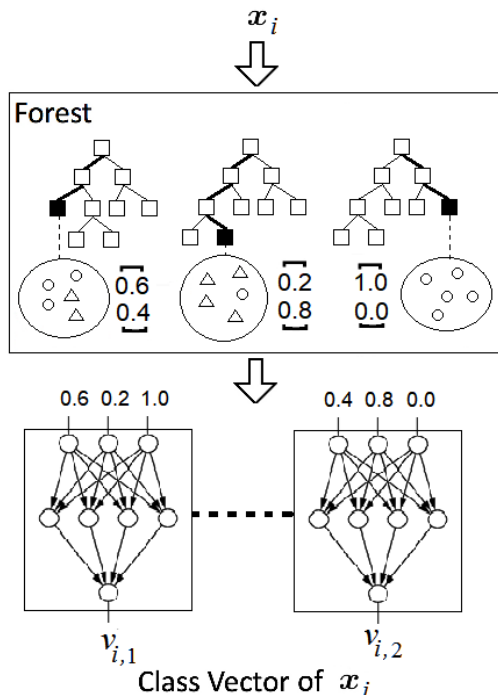


Рис. 2. Сиамская нейронная сеть для вычисления вероятностей классов

Учитывая тот факт, что ограничения для весов линейные, можно увидеть, что полученная задача

оптимизации квадратичная. Поэтому ее решение не представляет больших сложностей.

Однако следует отметить, что полученные векторы классов случайного леса являются линейными функциями распределений вероятностей классов отдельных деревьев решений, что не позволяет моделировать более сложные зависимости.

III. НЕЙРОННЫЕ СЕТИ КАК ФУНКЦИЯ ВЕРОЯТНОСТЕЙ

Взглянем на усреднение весов с другой стороны. Значение $v_{i,c}$ может быть представлено как функция f вероятностей $\mathbf{p}_{i,c}$, т.е., $v_{i,c} = f(\mathbf{p}_{i,c})$. Важно отметить, что функция f не зависит от класса c . Она одинаковая для всех классов. Предположим теперь, что функция f является нелинейной и реализуется, используя нейронную сеть. Это означает, что для каждого класса необходимо одинаковым образом преобразовать вектор $\mathbf{p}_{i,c}$ для получения вектора \mathbf{v}_i для случайного леса. Фактически вектор классов \mathbf{v}_i можно рассматривать как представление признаков вектора $\mathbf{p}_{i,c}$. Это можно сделать, используя C одинаковых нейронных сетей с одинаковыми параметрами. Входным вектором c -ой сети является вектор $\mathbf{p}_{i,c}$ длины T . Выходом c -ой сети является значение близкое к 1, если метка класса i -го примера совпадает с номером сети c , т.е., если условие $y_i = c$ выполняется, иначе выходное значение ожидается близким к 0. Нейронные сети обучаются на основе обучающих примеров $\mathbf{p}_{i,c}$, полученных для каждого обучающего примера (\mathbf{x}_i, y_i) , $i = 1, \dots, n$. Главное условие для обучения заключается в том, что параметры (веса всех соответствующих соединений) всех нейронных сетей должны быть одинаковыми. Это означает, что все сети обучаются одновременно.

Рис. 2 иллюстрирует использование одинаковых нейронных сетей с одинаковыми параметрами для вычисления векторов классов случайного леса. Штриховая линия между нейронными сетями указывает, что параметры сетей должны быть одинаковыми. В частном случае двух классов имеет известную сиамскую нейронную сеть [4]. Настраиваемые параметры нейронных сетей (число нейронов, число скрытых слоев, вид функции активации и т.д.) определяются в процессе тестирования. Они зависят от рассматриваемых наборов данных и числа деревьев.

IV. ЧИСЛОВЫЕ ЭКСПЕРИМЕНТЫ

Для иллюстрации предлагаемой комбинации случайных леса и нейронной сети, выполнено его сравнение с обычным случайным лесом, используя некоторые общедоступные наборы данных из репозитория UCI Machine Learning Repository [11]. Нами разработано программное обеспечение с использованием Python, реализующее процедуру для вычисления векторов классов.

Показатель точности классификации, F-мера или F_1 -мера, используется в числовых экспериментах для учета ситуации несбалансированности классов обучающей выборки. Для оценки точности реализуется процедура кросс-валидации со 100 повторениями, где в процессе каждого запуска модели случайным образом выбираются $n_{\text{train}}=3n/4$ обучающих элементов и $n_{\text{test}}=n/4$ тестовых данных. Обозначения F_{RF} и $F_{\text{RF-NN}}$ используются для F-меры, соответствующей случайному лесу и предлагаемой комбинации соответственно.

Использовалась нейронная сеть, имеющая два скрытых слоя. Количество нейронов в каждом слое является настраиваемым параметром, который зависит от анализируемого набора данных. Нейронные сети с различными значениями числа нейронов тестировались для выбора тех значений, которые приводят к лучшим результатам. То же самое можно сказать и о числе деревьев решений в случайном лесу.

Первый набор данных называется Ionosphere. Он содержит 351 пример из 2 классов с 34 признаками. Для него получены показатели $F_{\text{RF}}=0.939$ и $F_{\text{RF-NN}}=0.941$.

Второй набор называется Mammographic masses. Он содержит 961 пример из 2 классов с 5 признаками. Для этого набора получены показатели $F_{\text{RF}}=0.836$ и $F_{\text{RF-NN}}=0.84$.

Третий набор называется Ecoli. Он состоит из 336 примеров с 8 признаками из 8 классов. Для этого набора получены показатели $F_{\text{RF}}=0.846$ и $F_{\text{RF-NN}}=0.85$.

Можно заметить из представленных числовых результатов, что предложенная комбинация обеспечивает лучшие результаты по сравнению со случайным лесом.

V. ЗАКЛЮЧЕНИЕ

Комбинация случайного леса и нейронной сети была представлена в работе. Идея, лежащая в основе этой комбинации исходит из того факта, что процедура весового усреднения вероятностей классов для вычисления вероятностей класса всего случайного леса может быть улучшена за счет введения нейронной сети как нелинейной функции распределений вероятностей классов. Оказывается можно управлять выходом случайного леса в соответствии с целью решаемой задачи. В частности, если решается задача классификации, то функция потерь сети – евклидово расстояние между вектором классов и вектором $\mathbf{o}_i = (0, \dots, 0, 1, 0, \dots, 0)$, соответствующим ожидаемому вектору классов.

Следует отметить, что предлагаемая реализация может рассматриваться как первый шаг, который должен быть сделан перед построением более сложной и интересной модели глубокого леса. [18]. Это новый метод композиции, который можно представить как комбинация комбинаций

моделей, включая случайные леса и стекинг. Структура композиции состоит из слоев аналогично многослойной нейронной сети, но каждый слой в глубоком лесу состоит из нескольких случайных лесов вместо нейронов. Идея обобщения заключается в том, что все случайные леса в глубоком лесу дополняются нейронными сетями. Это является одним из направлений для дальнейших исследований.

СПИСОК ЛИТЕРАТУРЫ

- [1] Biau G. and Scornet E. A random forest guided tour // arXiv:1511.05741v1, Nov 2015.
- [2] Breiman L. Bagging predictors // Machine Learning, 24(2):123-140, 1996.
- [3] Breiman L. Random forests. Machine learning, 45(1):5-32, 2001.
- [4] Bromley J. et al. Signature verification using a Siamese time delay neural network // International Journal of Pattern Recognition and Artificial Intelligence, 7(4):737-744, 1993.
- [5] Criminisi A., Shotton J., Konukoglu E. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning // Foundations and Trends in Computer Graphics and Vision, 7(2-3):81-227, 2011.
- [6] Daho M.E.H., Settout N., Lazouni M.E.A., and Chikh M.E.A.. Weighted vote for trees aggregation in random forest // International Conference on Multimedia Computing and Systems (ICMCS), pages 438-443. IEEE, April 2014.
- [7] Ferreira A.J. and Figueiredo M.A.T.. Boosting algorithms: A review of methods, theory, and applications // Ensemble Machine Learning: Methods and Applications, pages 35-85. Springer, New York, 2012.
- [8] Genuer R., Poggi J.-M., Tuleau-Malot C., and Villa-Vialaneix N. Random forests for big data // Big Data Research, 9:28-46, 2017.
- [9] Kim H., Kim H., Moon H., and Ahn H.. A weight-adjusted voting algorithm for ensemble of classifiers // Journal of the Korean Statistical Society, 40(4):437-449, 2011.
- [10] Li H.B., Wang W., Ding H.W., and Dong J.. Trees weighting random forest method for classifying high-dimensional noisy data // IEEE 7th International Conference on E-Business Engineering, pages 160-163. IEEE, Nov 2010.
- [11] Lichman M. UCI machine learning repository, 2013.
- [12] Louppe G.. Understanding random forests: From theory to practice // arXiv:1407.7502v3, June 2015.
- [13] Rokach L.. Ensemble-based classifiers // Artificial Intelligence Review, 33(1-2):1-39, 2010.
- [14] Rokach L.. Decision forest: Twenty years of research // Information Fusion, 27:111-125, 2016.
- [15] Utkin L.V. and Ryabinin M.A. A Siamese deep forest // Knowledge-Based Systems, 139:13-22, 2018.
- [16] Wozniak M., Grana M., and Corchado E.. A survey of multiple classifier systems as hybrid systems // Information Fusion, pages 3-17, 2014.
- [17] Zhou Z.-H.. Ensemble Methods: Foundations and Algorithms. CRC Press, Boca Raton, 2012.
- [18] Zhou Z.-H. and Feng J.. Deep forest: Towards an alternative to deep neural networks // Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI'17), pages 3553-3559, Melbourne, Australia, 2017.