Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра: теории функций и
стохастического анализа

РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА ДЛЯ ПОСТРОЕНИЯ ДЕРЕВА РЕШЕНИЙ

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студента 4 курса 451 группы направления
38.03.05 — Бизнес-информатика механико-
математического факультета Еремеевой
Елены Валерьевны

Научный руководитель
д. ф.-м. н., доцент _____ С. П. Сидоров

Заведующий кафедрой
д. ф.-м. н., доцент _____ С. П. Сидоров

Саратов 2017

ВВЕДЕНИЕ

Актуальность темы исследования.

Стремительное развитие информационных технологий, в частности, прогресс в методах сбора, хранения и обработки данных позволил многим организациям собирать огромные массивы данных, которые необходимо анализировать. Объемы этих данных настолько велики, что возможностей экспертов уже не хватает, что породило спрос на методы автоматического исследования (анализа) данных, который с каждым годом постоянно увеличивается.

В современных условиях быстроизменяющейся внешней среды существует большое количество методов принятия решений. Одним из наиболее эффективных и прогрессивных из них является «дерево решений».

Традиционно, «дерево решений» – это способ представления классификационных правил в иерархической, последовательной структуре, позволяющей наглядно отобразить последовательность принятия решений и их результаты. В настоящее время наблюдается тенденция мировой интеграции и увеличения размеров фирм, а, следовательно, и роста объемов информации, необходимой для принятия решений, появляется потребность в ускорении процесса принятия решений по методу «дерево решений». В связи с вышесказанным требуется автоматизация алгоритмов принятия решений. Для этого разработан и активно применяется целый ряд алгоритмов таких как, C 4.5, CART, ID3 и др.

Метод деревьев решений является одним из наиболее популярных методов на сегодняшний день, используемых на этапе выбора альтернатив.

Актуальность определила выбор **темы** данной работы: «Разработка программного продукта для построения дерева решений».

Цель работы: получение теоретических и практических навыков построения деревьев решений с использованием алгоритма C 4.5.

Объект исследования: деревья решений.

Предмет исследования: алгоритм для решения задач классификации, основанный на построении дерева решений.

Для достижения поставленных целей в работе необходимо решить следующие **задачи**:

1) рассмотреть задачи классификации;
2) рассмотреть понятие и структуру дерева решений;
3) описать метод построения деревьев решений;
4) разработать программный продукт для построения деревьев решений.

Для решения поставленных задач были использованы следующие теоретические методы исследования: изучение источников, теоретический анализ, обобщение литературных данных.

Практическая значимость проводимой работы заключается в разработке программного продукта для построения дерева решений.

Основное содержание работы.

Выпускная квалификационная работа состоит из введения, двух теоретических и одной практической главы, заключения, списка использованных источников.

Введение содержит основные положения: статистически подкреплённую актуальность темы исследования; цель, объект, предмет, задачи исследования; практическую значимость исследования.

Первый раздел «Некоторые методы машинного обучения» описывает теоретические основы классификации.

Задача классификации — формализованная задача, в которой имеется множество объектов, разделённых некоторым образом на классы. Задано конечное множество объектов, для которых известно, к каким классам они относятся. Это множество называется выборкой. Классовая принадлежность остальных объектов неизвестна.

Классификация — это общенаучный метод систематизации знания, направленный на организацию некоторой совокупности изучаемых объектов различных областей действительности, знания и деятельности, в систему соподчинённых групп (классов), по которым эти объекты распределены на основании их сходства в определённых сущностных свойствах.

Классифицировать объект — значит, указать номер класса, к которому относится данный объект.

Классификация объекта — номер или наименование класса, выдаваемый алгоритмом классификации в результате его применения к данному конкретному объекту.

В математической статистике задачи классификации называются также задачами дискриминантного анализа. В машинном обучении задача классификации решается, в частности, с помощью методов искусственных нейронных сетей при постановке эксперимента в виде обучения с учителем.

Существуют также другие способы постановки эксперимента — обучение без учителя, но они используются для решения другой задачи — кластеризации или таксономии. В этих задачах разделение объектов обучающей выборки на классы не задаётся, и требуется классифицировать объекты только на основе их сходства друг с другом.

Второй раздел «Деревья решений» описывает теоретические основы построения дерева решений.

Деревья решения являются одним из наиболее популярных подходов к решению задач data mining. Они создают иерархическую структуру классифицирующих правил типа «если... то...», имеющую вид дерева. Для того чтобы решить, к какому классу отнести некоторый объект или ситуацию, требуется ответить на вопросы, стоящие в узлах этого дерева, начиная с его корня. Для бинарных деревьев, вопросы имеют вид «значение параметра A больше x». Если ответ положительный, осуществляется переход к правому узлу

следующего уровня, если отрицательный – то к левому узлу; затем снова следует вопрос, связанный с соответствующим узлом.

Популярность подхода связана с наглядностью и понятностью. Но очень остро для деревьев решений стоит проблема значимости. Дело в том, что отдельным узлам на каждом новом построенном уровне дерева соответствует все меньшее и меньшее число записей данных – дерево дробит данные на большое количество частных случаев. Чем больше этих частных случаев, чем меньше обучающих примеров попадает в каждый такой частный случай, тем менее уверенной становится их классификация. Если построенное дерево слишком «кустистое» – состоит из неоправданно большого числа мелких веточек – оно не будет давать статистически обоснованных ответов. Как показывает практика, в большинстве систем, использующих деревья решений, эта проблема не находит удовлетворительного решения. Кроме того, деревья решений дают полезные результаты только в случае независимых признаков. В противном случае они лишь создают иллюзию логического вывода.

Область применения деревья решений в настоящее время широка, но все задачи, решаемые этим аппаратом, могут быть объединены в следующие три класса:

- Описание данных**:** деревья решений позволяют хранить информацию о данных в компактной форме, вместо них мы можем хранить дерево решений, которое содержит точное описание объектов.
- Классификация: деревья решений отлично справляются с задачами классификации, т.е. отнесения объектов к одному из заранее известных классов. Целевая переменная должна иметь дискретные значения.
- Регрессия: если целевая переменная имеет непрерывные значения, деревья решений позволяют установить зависимость целевой переменной от независимых (входных) переменных. Например, к этому классу относятся задачи численного прогнозирования.

Преимущества использования деревьев решений.

1) Простота восприятия. Результат построения дерева решений легко интерпретируется пользователем. Дерево решений наглядно поясняет, почему конкретный объект отнесен к тому или иному классу.

2) Алгоритм построения дерева решений не требует выбора входных атрибутов. Для построения используются все атрибуты, и алгоритм сам выбирает наиболее значимые и строит на их основе дерево решений.

3) Быстрота обучения.

4) Для построения дерева требуется малый объем информации, поэтому они занимают мало места в памяти.

5) Относительная гибкость. Деревья решений позволяют работать с непрерывными и символьными целевыми признаками. Во многих алгоритмах построения деревьев решений имеется возможность обработки пропущенных значений. Это позволяет применять деревья решения в самых разных задачах.

Дерево решений можно определить как структуру, которая состоит из

- узлов-листьев, каждый из которых представляет определенный класс;
- узлов принятия решений, специфицирующих определенные тестовые процедуры, которые должны быть выполнены по отношению к одному из значений атрибутов; из узла принятия решений выходят ветви, количество которых соответствует количеству возможных исходов тестирующей процедуры.

Более формально *дерево* можно определить как конечное множество *T*, состоящее из одного или множества *узлов*, таких, что

а) Имеется один специально обозначенный узел, называемый *корнем* данного дерева.

б) Остальные узлы (исключая корень) содержатся в $m \geq 0$ попарно непересекающихся множествах $T_1, \ldots, T_m$, каждое из которых в свою очередь является деревом. Деревья $T_1, \ldots, T_m$ называются *поддеревьями* данного корня.

Из данного определения следует, что каждый узел дерева является корнем некоторого поддерева, которое содержится в этом дереве. Число поддеревьев данного узла называется *степенью* этого узла. Узел с нулевой степенью называется *листом*. *Уровень* узла по отношению к дереву T определяется следующим образом: говорят, что корень имеет уровень 1, а другие узлы имеют уровень на единицу выше их уровня относительно содержащего их поддерева T_j этого корня.

Если в дереве существует относительный порядок поддеревьев $T_1, \ldots, T_m$, то говорят, что дерево является *упорядоченным*; в случае, когда в упорядоченном дереве $m \geq 2$, имеет смысл называть T_2 «вторым поддеревом» данного корня и т.д.; если два дерева, отличающиеся друг от друга только относительным порядком узлов поддеревьев, не считать различными, то в этом случае говорят, что дерево является *ориентированным*, поскольку здесь имеет значение только относительная ориентация узла, а не их порядок.

Стандартная терминология для структур типа дерева: каждый корень является *отцом* корней своих поддеревьев, последние являются *братьями* между собой и *сыновьями* своего *отца*. Корень же всего дерева не имеет отца.

Дерево решения представляет один их способов разбиения множества данных на классы или категории. Корень дерева неявно содержит все классифицируемые данные, а листья – определенные классы после выполнения классификации. Промежуточные узлы дерева представляют пункты принятия решения о выборе или выполнения тестирующих процедур с атрибутами элементов данных, которые служат для дальнейшего разделения данных в этом узле.

Можно рассматривать дерево решений и с другой точки зрения: промежуточные узлы дерева соответствуют атрибутам классифицируемых объектов, а дуги – возможным альтернативным значениям этих атрибутов.

В третьем разделе работы «Разработка программного продукта для построения деревьев решений» описывается практическая реализация программного продукта для построения дерева решений.

Алгоритм C4.5 строит дерево решений с неограниченным количеством ветвей у узла. Данный алгоритм может работать только с дискретным зависимым атрибутом и поэтому может решать только задачи классификации. C4.5 считается одним из самых известных и широко используемых алгоритмов построения деревьев классификации.

Для работы алгоритма C4.5 необходимо соблюдение следующих требований:

Каждая запись набора данных должна быть ассоциирована с одним из предопределенных классов, т.е. один из атрибутов набора данных должен являться меткой класса.

Классы должны быть дискретными. Каждый пример должен однозначно относиться к одному из классов.

Количество классов должно быть значительно меньше количества записей в исследуемом наборе данных.

Высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

Распространенный способ реализации деревьев решений – это построение дерева на языке программирования Python. Чтобы оценить, насколько хорош выбранный атрибут, алгоритм сначала вычисляет энтропию всей группы. Затем он пытается разбить группу по возможным значениям каждого атрибута и вычисляет энтропию двух новых групп.

Для определения того, какой атрибут дает наилучшее разбиение, вычисляется информационный выигрыш, то есть разность между текущей

энтропией и средневзвешенной энтропией двух новых групп. Он вычисляется для каждого атрибута, после чего выбирается тот, для которого информационный выигрыш максимален. Вычисляя для каждого узла наилучший атрибут и расщепляя ветви, алгоритм создает дерево решений с помощью языка программирования Python 3.6.

В приложении представлен код на Python для реализации построения дерева решений.

ЗАКЛЮЧЕНИЕ

В заключении хотелось бы отметить, что своевременная разработка и принятие правильного решения — главные задачи любой организации. Непродуманное решение может дорого стоить компании. Когда нужно принять несколько решений в условиях неопределенности, когда каждое решение зависит от исхода предыдущего решения или исходов испытаний, то применяют схему, называемую деревом решений.

Список используемых источников

1. BreimanL.,Friedman J.H., Olshen R.A., & Stone C.J. Classification and regression trees. – Monterey, CA: Wadsworth & Brooks / Cole Advanced Books & Software, 1984.

2. Митчелл, Том М. Машинное обучение. – McGraw-Hill, 1997.

3. Quinlan JR C4.5:Программы для машинного обучения. – Morgan Kaufmann Publishers, 1993.

4. Пальмов С.В., Мифтахова А.А. Реализация деревьев решений в различных аналитических системах // Перспективы науки. №1(64), 2015. – С. 81-87.

5. Мифтахова А.А. Реализация алгоритма С 4.5 интеллектуального анализа данных, основанного на деревьях решений // Труды 12 МНПК «Проблемы теории и практики современной науки». Нефтекамск, 2015. – С. 113-120.