1 ВВЕДЕНИЕ В АНАЛИЗ ДАННЫХ

1.1 Аналитический подход к моделированию. Движение от модели к результату. Информационный подход к моделированию. Построение модели от данных.

Аналитический подход к моделированию.

Модель – объект или описание объекта, системы для замещения (при определенных условиях, предположениях, гипотезах) одной системы (то есть оригинала) другой системой для лучшего изучения оригинала или воспроизведения каких-либо его свойств. Моделирование – универсальный метод получения, описания и использования знаний. Применяется в любой профессиональной деятельности. Таким образом, анализ данных тесно связан с моделированием.

Модель в традиционном понимании представляет собой результат отображения одной структуры (изученной) на другую (малоизученную). Так, отображая физическую систему (объект) на математическую (например, математический аппарат уравнений), получим физико-математическую модель системы, или математическую модель физической системы. Любая модель строится и исследуется при определенных допущениях, гипотезах. Делается это обычно с помощью математических методов.

Движение от модели к результату.

Аналитический подход к моделированию базируется на том, что исследователь при изучении системы отталкивается от модели (рис. 1.1).

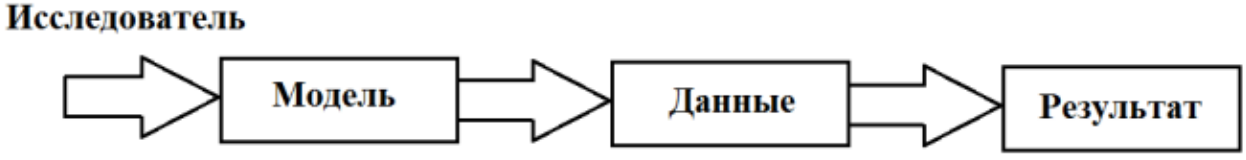


Рис. 1.1. Движение от модели к результату

В этом случае он по тем или иным соображениям выбирает подходящую модель. Как правило, это теоретическая модель, закон, известная зависимость,

представленная чаще всего в функциональном виде (например, уравнение, связывающее выходной параметр у с входными воздействиями $x1$, x2). Варьирование входных параметров на выходе даст результат, который моделирует поведение системы в различных условиях.

Результат моделирования может соответствовать действительности, а может и не соответствовать. В последнем случае исследователю ничего не остается, кроме как выбрать другую модель или другой метод ее исследования. Новая модель, возможно, будет более адекватно описывать рассматриваемую систему.

При аналитическом подходе не модель «подстраивается» под действительность, а мы пытаемся подобрать существующую аналитическую модель таким образом, чтобы она адекватно отражала реальность.

Модель всегда исследуется каким-либо методом (численным, качественным и т. п.). Поэтому выбор метода моделирования часто означает выбор модели.

Информационный подход к моделированию.

При использовании традиционного аналитического подхода неизбежно возникнут проблемы из-за несоответствия между методами анализа и реальностью, которую они призваны отражать. Существуют трудности, связанные с формализацией бизнес-процессов. Здесь факторы, определяющие явления, столь многообразны и многочисленно, их взаимосвязи так «переплетены, что почти никогда не удается создать модель, удовлетворяющую таким же условиям, Простое наложение известных аналитических методов, законов, зависимостей на изучаемую картину реальности не принесет успеха.

В сложности и слабой формализации бизнес-процессов главным образом «виноват» человеческий фактор, поэтому бывает трудно судить о характере закономерностей априори (а иногда и апостериори (знание, полученное из опыта в противоположность априори), после реализации какого-либо математического метода). С одинаковым успехом описывать эти закономерности могут различные модели. Использование разных методов для решения одной и той же задачи

нередко приводит исследователя к противоположным выводам. Какой метод выбрать? Получить ответ на подобный вопрос можно, лишь глубоко проанализировав как смысл решаемой задачи, так и свойство используемого математического аппарата.

Поэтому в последние годы получил распространение информационный подход к моделированию, ориентированный на использование данных, Его цель освобождение аналитика от рутинных операций и возможных сложностей в понимании и применении современных математических методов.

При информационном подходе реальный объект рассматривается как «черный ящик», имеющий ряд входов и выходов, между которыми моделируются некоторые связи. Иными словами, известна только структура модели (например, нейронная сеть, линейная регрессия), а сами параметры модели «подстраиваются» под данные, которые описывают поведение объекта. Для корректировки параметров модели используется обратная связь – отклонение результата моделирования от действительности, а процесс настройки модели часто носит итеративный (то есть цикличный) характер (рис, 1.2).

Построение модели от данных.

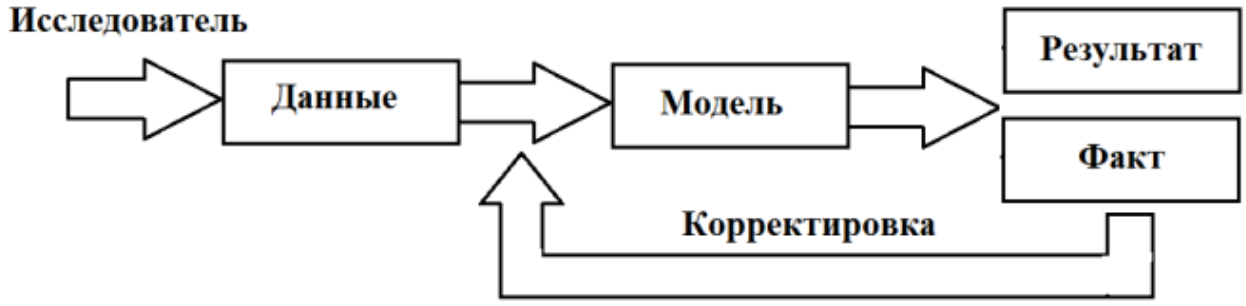


Рис. 1.2. Построение модели от данных

Таким образом, при информационном подходе отправной точкой являются данные, характеризующие исследуемый объект, и модель «подстраивается» под действительность. Модели, полученные с помощью информационного подхода, учитывают специфику моделируемою объекта, явления, в отличие от аналитического подхода.

1.2 Информационный подход к моделированию. Построение модели от данных. Этапы моделирования. Тиражирование знаний. Процесс построения модели.

Информационный подход к моделированию.

Смотреть вопрос №1.1.

Построение модели от данных.

Смотреть вопрос №1.1.

Этапы моделирования.

Построение моделей универсальный способ изучения окружающего мира, позволяющий обнаруживать зависимости, прогнозировать, разбивать на группы и решать множество других важных задач. Но самое главное: полученные таким образом знания можно тиражировать.

Тиражирование знаний.

Тиражирование знаний – совокупность методологических и инструментальных средств создания моделей, которые обеспечивают конечным пользователям возможность использовать результаты моделирования для принятия решений без необходимости понимания методик, при помощи которых эти результаты получены.

Процесс построения модели.

Процесс построения моделей состоит из нескольких шагов (рис. 1.3).

Процесс построения модели состоит из нескольких шагов (рис. 1.3):

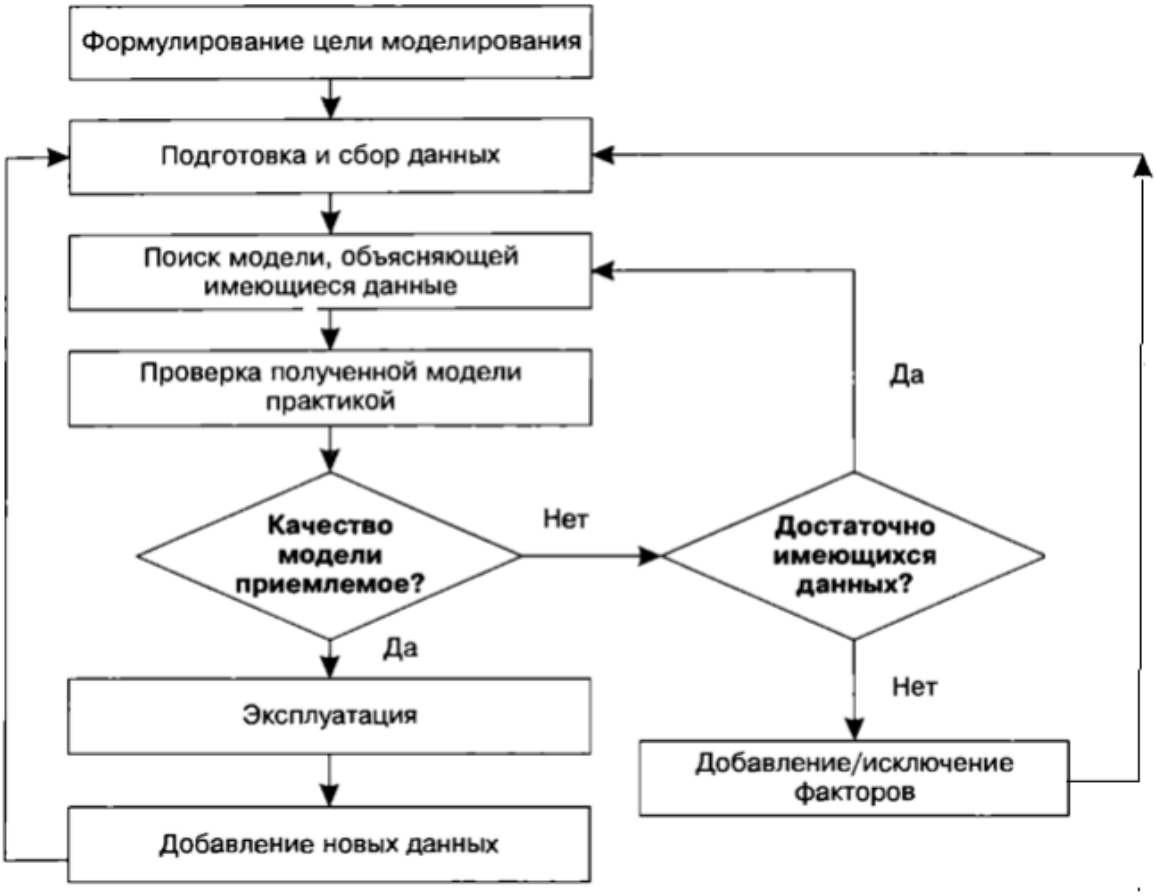


Рис. 1.3. Процесс построения модели

1.3 Процесс построения модели. Формулирование цели моделирования. Подготовка и сбор данных. Поиск модели.

Процесс построения модели.

Смотреть вопрос №1.2.

Формулирование цели моделирования.

При построении модели следует отталкиваться от задачи, которую можно рассматривать как получение ответа на интересующий заказчика вопрос. Например, в розничной торговле к таким вопросам относятся следующие:

- Какова структура продаж за определенный период?
- Какие клиенты приносят наибольшую прибыль?
- Какие товары продаются или заказываются вместе?
- Как оптимизировать товарные остатки на складах?

В этом случае можно говорить о создании модели прогнозирования продаж, модели выявления ассоциаций и т. д, Данный этап также называют анализом проблемной ситуации.

Подготовка и сбор данных.

Информационный подход к моделированию основан на использовании данных, подготовить и систематизировать которые – отдельная задача, Принципам подготовки данных, а также их очистке и обогащение Вы должны были проходить по предмету «Базы данных».

Поиск модели.

После сбора и систематизации данных переходят к поиску модели, которая объясняла бы имеющиеся данные, позволила бы добиться эмпирически обоснованных ответов на интересующие вопросы. В промышленном анализе данных предпочтение отдается самообучающимся алгоритмам, машинному обучению, методам Data Mining.

Если построенная модель показывает приемлемые результаты на практике (например, в тестовой эксплуатации), ее запускают в промышленную эксплуатацию. Так, при тестовой эксплуатации скоринговой модели (система оценки клиентов, в основе которой заложены статистические методы), рассчитывающей кредитный рейтинг клиента и принимающей решение о выдаче кредита, каждое решение может подтверждаться человеком кредитным экспертом. При запуске скоринга в промышленную эксплуатацию человеческий фактор удаляется теперь решение принимает только компьютер.

Если качество модели неудовлетворительное, то процесс построения модели повторяется, как это показано на рис. 1.3.

Моделирование позволяет получать новые знания, которые невозможно извлечь каким-либо другим способом. Кроме того, полученные результаты представляют собой формализованное описание некого процесса, вследствие чего поддаются автоматической обработке. Однако результаты, полученные при использовании моделей, очень чувствительны к качеству данных, к знаниям

аналитика и экспертов и к формализации самого изучаемого процесса. К тому же почти всегда имеются случаи, не укладывающиеся ни в какие модели.

На практике подходы комбинируются. Например, визуализация данных наводит аналитика на некоторые идеи, которые он пробует проверить при помощи различных моделей, а к полученным результатам применяются методы визуализации.

Полнофункциональная система анализа не должна замыкаться на применении только одного подхода или одной методики. Механизмы визуализации и построения моделей должны дополнять друг друга. Максимальную отдачу можно получить, комбинируя методы и подходы к анализу данных.

1.4 Структурированные данные. Степени структурированности. Формализация данных. Принципы сбора данных.

Структурированные данные.

Данные, описывающие реальные объекты, процессы и явления, могут быть представлены в различных формах и иметь разные тип и вид.

Степени структурированности.

По степени структурированности выделяют следующие формы представления данных:

- неструктурированные;
- структурированные;
- слабоструктурированные.

К неструктурированным относятся данные, произвольные по форме, включающие тексты и графику, мультимедиа (видео, речь, аудио). Эта форма представления данных широко используется, например, в Интернете, а сами данные представляются пользователю в виде отклика поисковыми системами.

Структурированные данные отражают отдельные факты предметной области. Структурированными называются данные, определенным образом упорядоченные и организованные с целью обеспечения возможности применения к ним некоторых действий (например, визуального или машинного анализа). Это основная форма представления сведений в базах данных.

Организация того или иного вида хранения данных (структурированных или неструктурированных) связана с обеспечением доступа к ним. Под доступом понимается возможность выделения элемента данных (или множества элементов) среди других элементов по каким-либо признакам с целью выполнения некоторых действий над элементом.

Одной из самых распространенных моделей хранения, структурированных данных, является таблица.

Неструктурированные данные непригодны для обработки напрямую методами анализа данных, поэтому такие данные подвергаются специальным приемам структуризации, причем сам характер данных в процессе структуризации может существенно измениться.

Например, в анализе текстов при структурировании из исходного текста может быть сформирована таблица с частотами встречаемости слов, и уже такой набор данных будет обрабатываться методами, приводными для структурированных данных.

Слабоструктурированные данные – это данные, для которых определены некоторые правила и форматы, но в самом общем виде. Например, строка с адресом, строка в прайс-листе, ФИО и т. п. В отличие от неструктурированных, такие данные с меньшими усилиями преобразуются к структурированной форме, однако без процедуры преобразования они тоже непригодны для анализа.

Подавляющее большинство методов анализа данных работает только с хорошо структурированными данными, представленными в табличном виде.

Формализация данных. Принципы сбора данных.

При сборе данных нужно придерживаться следующих принципов:

1. Абстрагироваться от существующих информационных систем и

имеющихся в наличии данных. Большие объемы накопленных данных совершенно не говорят о том, что их достаточно для анализа. Необходимо отталкиваться от задачи и подбирать данные для ее решения, а не брать имеющуюся информацию. К примеру, при построении моделей прогноза продаж опрос экспертов показал, что на спрос очень влияет цветовая характеристика товара. Анализ имеющихся данных продемонстрировал, что информация о цвете товарной позиции отсутствует в учетной системе. Значит, нужно каким-то образом добавить эти данные, иначе не стоит рассчитывать на хороший результат использования моделей.

2. Описать все факторы, потенциально влияющие на анализируемый процесс/объект. Основным инструментом здесь становится опрос экспертов и людей, непосредственно владеющих проблемной ситуацией. Необходимо максимально использовать знания экспертов о предметной области и, полагаясь на здравый смысл, постараться собрать и систематизировать максимум возможных предположений и гипотез.
3. Экспертно оценить значимость каждою фактора. Эта оценка не является окончательной, она будет отправной точкой. В процессе анализа вполне может выясниться, что фактор, который эксперты посчитали очень важным, таковым не является, и наоборот, незначимый, с их точки зрения, фактор может оказывать значительное влияние на результат.
4. Определить способ представления информации число, дата, да/нет, категория (то есть тип данных). Определить способ представления, то есть формализовать некоторые данные, просто. Например, объем продаж в рублях – это определенное число. Но довольно часто бывает непонятно, как представить фактор. Чаще всего такие проблемы возникают с качественными характеристиками. Например, на объемы продаж влияет качество товара. Качество сложное понятие, но если

этот показатель действительно важен, то нужно придумать способ его формализации. Скажем, качество можно определять по количеству брака на тысячу единиц продукции либо оценивать экспертно, разбив на несколько категорий отлично/хорошо/удовлетворительно/плохо.

5. Собрать все легкодоступные факторы. Они содержатся в первую очередь в источниках структурированной информации учетных системах, базах данных и т. п.
6. Обязательно собрать наиболее значимые, с точки зрения экспертов, факторы. Вполне возможно, что без них не удастся построить качественную модель.
7. Оценить сложность и стоимость сбора средних и наименее важных по значимости факторов. Некоторые данные легкодоступны, их можно извлечь из существующих информационных систем. Но есть информация, которую непросто собрать, например, сведения о конкурентах, поэтому необходимо оценить, во что обойдется сбор данных. Сбор данных не является самоцелью. Если информацию получить легко, то, естественно, нужно ее собрать. Если сложно, то необходимо соизмерить затраты на ее сбор и систематизацию с ожидаемыми результатами.

1.5 Технология извлечения знаний из баз данных (Knowledge Discovery in Databases – KDD). Последовательность шагов, выполняемых в процессе KDD.

Технология извлечения знаний из баз данных (Knowledge Discovery in Databases – KDD).

Информационный подход к анализу получил распространение в таких методиках извлечения знаний, как KDD (Knowledge Discovery in Databases – извлечение знаний из баз данных) и Data Mining. Сегодня на базе этих методик создается большинство прикладных аналитических решений в бизнесе и многих

других областях. Несмотря на разнообразие бизнес задач, почти все они могут решаться по единой методике. Эта методика, зародившаяся в 1989 т., получила название Knowledge Discovery in Databases – извлечение знаний из баз данных. Она описывает не конкретный алгоритм или математический аппарат, а последовательность действий, которую необходимо выполнить для обнаружения полезного знания. Методика не зависит от предметной области; это набор атомарных операций, комбинируя которые можно получить нужное решение. KDD включает в себя этапы подготовки данных, выбора информативных признаков, очистки, построения моделей, постобработки и интерпретации полученных результатов. Ядром этого процесса являются методы Data Mining, позволяющие обнаруживать закономерности и знания.

Этими знаниями могут быть правила, описывающие связи между свойствами данных (деревья решений), часто встречающиеся шаблоны (ассоциативные правила), а также результаты классификации (нейронные сети) и кластеризации данных (карты Кохонена) и т.д.

Последовательность шагов, выполняемых в процессе KDD.

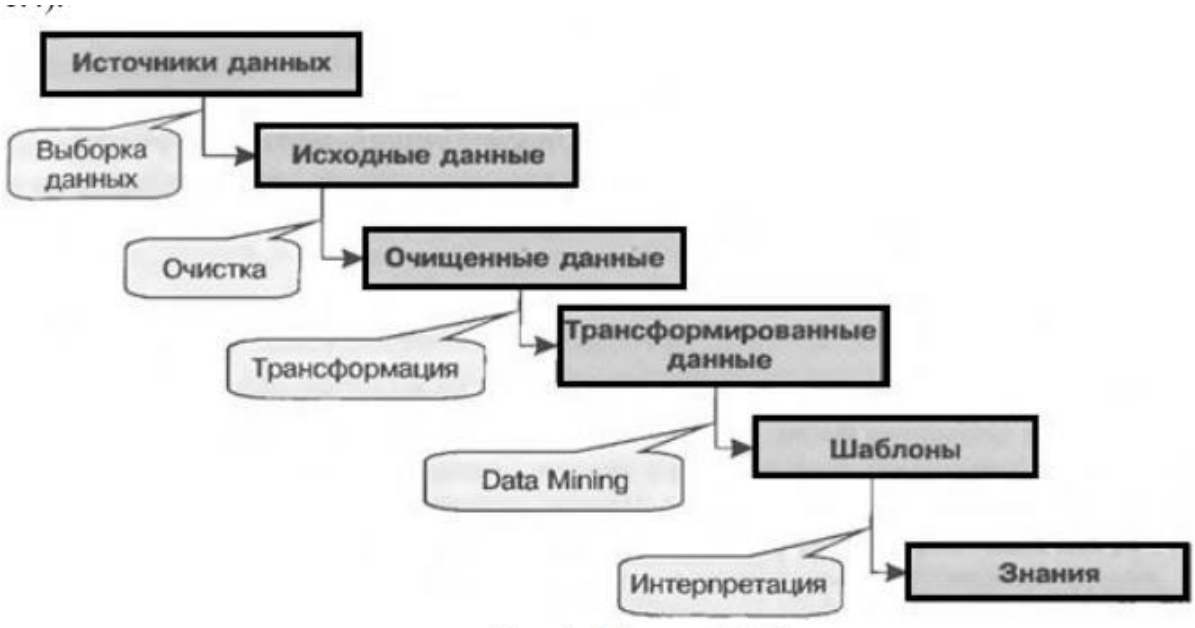


Рис. 1.4. Этапы KDD

Выборка данных. Первым шагом в анализе является получение исходной выборки. На основе отобранных данных строятся модели. Здесь требуется активное участие экспертов для выдвижения гипотез и отбора факторов,

влияющих на анализируемый процесс. Желательно, чтобы данные были уже собраны и консолидированы. Крайне необходимы удобные механизмы подготовки выборки: запросы, фильтрация данных. Чаще всего в качестве источника рекомендуется использовать специализированное хранилище данных, консолидирующее всю необходимую для анализа информацию.

Очистка данных. Реальные данные для анализа редко бывают хорошего качества. Необходимость в предварительной обработке при анализе данных возникает независимо от того, какие технологии и алгоритмы используются. Более того, эта задача может представлять самостоятельную ценность в областях, не имеющих непосредственного отношения к анализу данных. К задачам очистки данных относятся: заполнение пропусков, подавление аномальных значений, сглаживание. исключение дубликатов и противоречий и пр.

Трансформация данных. Этот шаг необходим для тех методов, при использовании которых исходные данные должны быть представлены в каком-то определенном виде. Дело в том, что различные алгоритмы анализа требуют специальным образом подготовленных данных. Например, для прогнозирования необходимо преобразовать временной ряд при помощи метода скользящего окна или вычислить агрегированные показатели. К задачам трансформации данных относятся: скользящее окно, приведение типов, выделение временных интервалов, квантование, сортировка, группировка и пр.

Data Mining. На этом этапе строятся модели.

Интерпретация. В случае, когда извлеченные зависимости и шаблоны непрозрачны для пользователя, должны существовать методы постобработки, позволяющие привести их к интерпретируемому виду. Для оценки качества полученной модели нужно использовать как формальные методы, так и знания аналитика. Именно аналитик может сказать, насколько применима полученная модель к реальным данным. Построенные модели являются, по сути, формализованными знаниями эксперта, а следовательно, их можно

тиражировать. Найденные знания должны быть применимы и к новым данным с некоторой степенью достоверности.

1.6 Технология добыча данных (Data Mining). Классификация задач Data Mining. Машинное обучение.

Технология добыча данных (Data Mining).

Термин Data Mining дословно переводится как добыча данных» или «раскопка данных» и имеет в англоязычной среде несколько определений, которые я уже приводил.

Data Mining обнаружение в сырых данных ранее неизвестных, нетривиальных, практически полезных и доступных интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности.

Зависимости и шаблоны, найденные в процессе применения методов Data Mining, должны быть нетривиальными и ранее неизвестными, например, сведения о средних продажах таковыми не являются, Знания должны описывать новые связи между свойствами, предсказывать значения одних признаков на основе других.

Нередко KDD (извлечение знаний из баз данных) отождествляют с Data Mining. Однако правильнее считать Data Mining шагом процесса KDD.

Классификация задач Data Mining.

1. Классификация – это установление зависимости дискретной выходной переменной от входных переменных.
2. Регрессия – это установление зависимости непрерывной выходной переменной от входных переменных.
3. Кластеризация – это группировка объектов (наблюдений, событий) на основе данных, описывающих свойства объектов. Объекты внутри кластера должны быть похожими друг на друга и отличаться от других, которые вошли в другие кластеры.

4. Ассоциация – выявление закономерностей между связанными событиями, Примером такой закономерности служит правило, указывающее, что из события X следует событие У. Такие правила называются ассоциативными. Впервые эта задача была предложена для нахождения типичных шаблонов покупок, совершаемых в супермаркетах, поэтому иногда ее называют анализом рыночной корзины. Если же нас интересует последовательность происходящих событий, то можно говорить о последовательных шаблонах установлении закономерностей между связанными во времени событиями. Примером такой закономерности служит правило, указывающее, что из события X спустя время t последует событие У.

Кроме перечисленных задач, часто выделяют анализ отклонений, анализ связей, отбор значимых признаков, хотя эти задачи граничат с очисткой и визуализацией данных.

Перечислим наиболее известные способы применения этих задач.

Классификация используется, если заранее известны классы, например, при отнесении нового товара к той или иной товарной группе, отнесении клиента к какой-либо категории (при кредитовании к одной из групп риска).

Регрессия используется для установления зависимостей между факторами. Например, в задаче прогнозирования зависимая величина объемы продаж, а факторами, влияющими на нее, могут быть предыдущие объемы продаж, изменение курсов валют, активность конкурентов и т. д. Или, например, при кредитовании физических лиц вероятность возврата кредита зависит от личных характеристик человека, сферы его деятельности, наличия имущества.

Кластеризация может использоваться для сегментации и построения профилей клиентов. При достаточно большом количестве клиентов становится трудно под ходить к каждому индивидуально, поэтому их удобно объединять в группы сегменты с однородными признаками. Выделять сегменты можно по нескольким группам признаков, например, по сфере деятельности или географическому расположению. После кластеризации можно узнать, какие

сегменты наиболее активны, какие приносят наибольшую прибыль, выделить характерные для них признаки. Эффективность работы с клиентами повышается благодаря учету их персональных предпочтений.

Ассоциативные правила помогают выявлять совместно приобретаемые товары. Это может быть полезно для более удобного размещения товара на при лавках, стимулирования продаж. Тогда человек, купивший пачку спагетти, не забудет купить к ней бутылочку соуса. Последовательные шаблоны могут использоваться при планировании продаж или предоставления услуг. Они похожи на ассоциативные правила, но в анализе добавляется временной показатель, то есть важна последовательность совершения операций. Например, если заемщик взял потребительский кредит, то с вероятностью 60 % через полгода он оформит кредитную карту.

В общем случае непринципиально, каким именно алгоритмом будет решаться задача, главное иметь метод решения для каждого класса задач. На сегодняшний день наибольшее распространение в Data Mining получили методы машинного обучения: деревья решений, нейронные сети, ассоциативные правила и т. д.

Машинное обучение.

Машинное обучение (machine learning) – обширный подраздел искусственного интеллекта, изучающий методы построения алгоритмов, способных обучаться на данных. Общая постановка задачи обучения следующая. Имеется множество объектов (ситуаций) и множество возможных ответов (откликов, реакций). Между ответами и объектами существует некоторая зависимость, но она неизвестна. Известна только конечная совокупность прецедентов пар вида «объект – ответ», – называемой обучающей выборкой. На основе этих данных требуется обнаружить зависимость, то есть построить модель, способную для любого объекта выдать достаточно точный ответ. Чтобы измерить точность ответов, вводится критерий качества.

2 ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ: ЗАДАЧА АССОЦИАЦИИ

2.1 Аффинитивный анализ. Примеры ассоциативных правил. Транзакция. Предметный набор. Ассоциативное правило.

Аффинитивный анализ.

Аффинитивный анализ (affinity analysis) – один из распространенных методов Data Mining. Его название происходит от английского слова affinity, которое в переводе означает «близость», «сходство». Цель дангоro метода исследование взаимной связи между событиями, которые происходят совместно. Разновидностью аффинитивноro анализа является анализ рыночной корзины, цель которого обнаружить ассоциации между различными событиями, то есть найти правила для количественного описания взаимной связи между двумя или более событиями. Такие правила называются ассоциативными правилами.

Примеры ассоциативных правил.

Примерами приложения ассоциативных правил могут быть следующие задачи:

- выявление наборов товаров, которые в супермаркетах часто покупаются вместе или никогда не покупаются вместе;
- определение доли клиентов, положительно относящихся к нововведениям в их обслуживании;
- определение профиля посетителей веб-ресурса;
- определение доли случаев, в которых новое лекарство оказывает опасный побочный эффект.

Транзакция.

Базовым понятием в теории ассоциативных правил является транзакция – некоторое множество событий, про исходящих совместно. Типичная транзакция

приобретение клиентом товара в супермаркете. В подавляющем большинстве случаев клиент покупает не один товар, а набор товаров, который называется рыночной корзиной. При этом возникает вопрос: является ли покупка одного товара в корзине следствием или причиной покупки другого товара, то есть связаны ли данные события? Эту связь и устанавливают ассоциативные правила, например, может быть обнаружено ассоциативное правило, утверждающее, что клиент, купивший молоко, с вероятностью 75 % купит и хлеб.

Предметный набор.

Следующее важное понятие – предметный набор. Это непустое множество предметов, появившихся в одной транзакции.

Анализ рыночной корзины – это анализ наборов данных для определения комбинаций товаров, связанных между собой. Иными словами, производится поиск товаров, присутствие которых в транзакции влияет на вероятность наличия других товаров или комбинаций товаров.

Ассоциативное правило.

Ассоциативное правило состоит из двух наборов предметов, называемых условием и следствием, записываемых в виде $X \rightarrow Y$, что читается так: «Из X следует Y». Таким образом, ассоциативное правило формулируется в виде: «Если условие, то следствие».

Условие может ограничиваться только одним предметом. Правила обычно отображаются с помощью стрелок, направленных от условия к следствию, например, помидоры → салат. Условие и следствие часто называются соответственно: левосторонним и правосторонним компонентами ассоциативного правила.

2.2 Ассоциативное правило. Поддержка ассоциативного правила. Достоверность ассоциативного правила. Примеры.

Ассоциативное правило.

Смотреть вопрос №2.1.

Поддержка ассоциативного правила.

Ассоциативные правила описывают связь между наборами предметов, соответствующие условию и следствию. Эта связь характеризуется двумя показателями – поддержкой (support) и достоверностью (confidence).

Обозначим базу данных транзакций как D, а число транзакций в этой базе – как T. Каждая транзакция d_i представляет собой некоторый набор предметов. Для вывода правил используются следующие ключевые показатели:

Поддержка - показатель частотности данного предметного набора во всех анализируемых транзакциях или число транзакций, которые содержат как условие, так и следствие (Формула 2.2.1).

$$supp(\mathrm{X} \to \mathrm{Y}) = P(\mathrm{X} \cap \mathrm{Y}) = \frac{\text{количество транзакций, содержащих } X \text{ и } Y}{\text{общее количество транзакций}} = \frac{|\mathrm{X} \cap \mathrm{Y}|}{|T|} \quad (2.2.1)$$

Достоверность ассоциативного правила.

Достоверность представляет собой меру точности правила и определяется как отношение количества транзакций, содержащих и условие, и следствие, к количеству транзакций, содержащих только условие. Показатель является условной вероятностью, отражающей наличие множества Y в наборе при наличии множества X. Достоверность вычисляется по Формуле 2.2.2.

$$conf(\mathrm{X} \to \mathrm{Y}) = \frac{P(\mathrm{X} \cap \mathrm{Y})}{P(\mathrm{X})} = \frac{\text{количество транзакций,содержащих } X \text{ и } Y}{\text{количество транзакций,содержащих только } X} = \frac{|\mathrm{X} \cap \mathrm{Y}|}{|\mathrm{X}|} \quad (2.2.2)$$

Примеры.

Из головы.

2.3 Ассоциативное правило. Значимость ассоциативных правил. Лифт. Левередж. Примеры.

Ассоциативное правило.

Смотреть вопрос №2.1.

Значимость ассоциативных правил.

Методики поиска ассоциативных правил обнаруживают все ассоциации, которые удовлетворяют ограничениям на поддержку и достоверность, наложенным пользователем. Это приводит к необходимости рассматривать десятки и сотни тысяч ассоциаций, что делает невозможным обработку такого количества данных вручную. Число правил желательно уменьшить таким образом, чтобы проанализировать только наиболее значимые из них. Значимость часто вычисляется как разность между поддержкой правила в целом и произведением поддержки только условия и поддержки только следствия.

Если условие и следствие независимы, то поддержка правила примерно соответствует произведению поддержек условия и следствия, то есть $SAB \approx SASB$. Это значит, что, хотя условие и следствие часто встречаются вместе, не менее часто они встречаются и по отдельности. Например, если товар A встречался в 70 транзакциях из 100, а товар B в 80 и в 50 транзакциях из 100 они встречаются вместе, то, несмотря на высокую поддержку ($SAB = 0{,}5$), это не обязательно правило. Просто эти товары покупаются независимо друг от друга, но в силу их популярности часто встречаются в одной транзакции. Поскольку произведение поддержек условия и следствия$SASB = 0{,}7 \times 0{,}8 = 0{,}56$, то есть отличается от $SAB = 0{,}5$ всего на 0,06, предположение о независимости товаров A и B достаточно обоснованно.

По этой причине при поиске ассоциативных правил используются дополнительные показатели, позволяющие оценить значимость правила. Можно выделить объективные и субъективные меры значимости правил, Объективными являются такие меры, как поддержка и достоверность, которые могут применяться независимо от конкретного приложения. Субъективные меры

связаны со специальной информацией, определяемой пользователем в контексте решаемой задачи, Такими субъективными мерами являются лифт (lift) и левередж (от англ., leverage «плечо», «рычаг»).

Лифт.

Отношение частоты появления условия в транзакциях, которые также содержат и следствие к частоте появления следствия в целом. Значения лифта большие 1 показывают, что условие чаще появляется в транзакциях, содержащих следствие, чем в остальных. Можно утверждать, что лифт является обобщенной мерой связи двух предметных наборов: при значениях лифта больше 1 связь положительная, при 1 она отсутствует, а при значениях меньше 1 – отрицательная. Лифт помогает понять, есть ли между X и Y значимая зависимость, или их совместное появление случайно. Метрика вычисляется по Формуле 2.3.1.

$$lift(\mathrm{X} \to \mathrm{Y}) = \frac{P(\mathrm{X} \cap \mathrm{Y})}{P(\mathrm{X}) * P(\mathrm{Y})} = \frac{supp(\mathrm{X} \to \mathrm{Y})}{supp(\mathrm{X}) * supp(\mathrm{Y})} \quad (2.3.1)$$

Левередж.

Отражает разность между наблюдаемой частотой, с которой условие и следствие появляются совместно (то есть поддержкой ассоциации), и произведением частот появления (поддержек) условия и следствия по отдельности. Вычисляется по Формуле 2.3.2.

$$leverage(\mathrm{X} \to \mathrm{Y}) = supp(\mathrm{X} \cap \mathrm{Y}) - supp(\mathrm{X}) * supp(\mathrm{Y}) \quad (2.3.2)$$

Примеры.

Из головы.

2.4 Поиск ассоциативных правил. Алгоритм Apriori. Частый предметный набор. Пример.

Поиск ассоциативных правил.

В процессе поиска ассоциативных правил может производиться обнаружение всех ассоциаций, поддержка и достоверность для которых превышают заданный минимум. Простейший алгоритм поиска ассоциативных правил рассматривает все возможные комбинации условий и следствий, оценивает для них поддержку и достоверность, а затем исключает все ассоциации, которые не удовлетворяют заданным ограничениям, Число возможных ассоциаций с увеличением числа предметов растет экспоненциально.

Поскольку реальные базы данных транзакций, рассматриваемые при анализе рыночной корзины, обычно содержат тысячи предметов, вычислительные затраты при поиске ассоциативных правил огромны.

Поиск ассоциативных правил путем вычисления поддержки и достоверности для всех возможных ассоциаций и сравнения их с заданным пороговым значением малоэффективен из-за больших вычислительных затрат.

Поэтому в процессе генерации ассоциативных правил широко используются методики, позволяющие уменьшить количество ассоциаций, которое требуется проанализировать. Одной из наиболее распространенных является методика, основанная на обнаружении так называемых частых наборов, когда анализируются только те ассоциации, которые встречаются достаточно часто. На этой концепции основан известный алгоритм поиска ассоциативных правил Apriori.

Алгоритм Apriori.

При практической реализации систем поиска ассоциативных правил используют различные методы, которые позволяют снизить пространство поиска до размеров, обеспечивающих приемлемые вычислительные и временные затраты, например, алгоритм Apriori. В основе алгоритма Apriori лежит понятие частого набора, который также можно назвать частым

предметным набором, часто встречающимся множеством (соответственно, он связан с понятием частоты). Под частотой понимается простое количество транзакций, в которых содержится данный предметный набор. Тогда частыми наборами будут те из них, которые встречаются чаще, чем в заданном числе транзакций.

Алгоритм состоит из следующих шагов:

1. Вычислить поддержку для каждого одиночного элемента.
2. Отобрать все элементарные наборы с поддержкой больше min_support.
3. Объединить попарно все наборы из L_{k-1}, чтобы получить кандидатов размера k.
4. Отсеять кандидатов, у которых хотя бы одно *(k-1)* подмножество не содержится в L_{k-1}.
5. Вычислить support и оставить тех кандидатов, поддержка которых больше min_support.
6. Повторить шаги 3–5 для k = 2, 3, … n.
7. Продолжать генерировать наборы L_1, L_2 пока L_k не пуст.
8. Объединить все наборы в результирующий набор частых предметов (items).
9. Для каждого частого набора с размером больше двух сгенерировать все возможные непустые подмножества X, Y.
10. Для каждого правила $X \rightarrow Y$ рассчитать метрики.

Частый предметный набор.

Частый предметный набор – предметный набор с поддержкой больше заданного порога либо равной ему. Этот порог называется минимальной поддержкой.

Методика поиска ассоциативных правил с использованием частых наборов состоит из двух шагов:

1. Следует найти частые наборы.

2. На их основе необходимо сгенерировать ассоциативные правила, удовлетворяющие условиям минимальной поддержки и достоверности. Чтобы сократить пространство поиска ассоциативных правил, алгоритм Apriori использует свойство антимонотонности. Свойство утверждает, что если предметный набор Z не является частым, то добавление некоторого нового предмета A к набору Z не делает его более частым. Другими словами, если Z не является частым набором, то и набор $Z \cup A$ также не будет являться таковым, Данное полезное свойство позволяет значительно уменьшить пространство поиска ассоциативных правил.

Пример.

Из головы.

3 ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ: ЗАДАЧА КЛАСТЕРИЗАЦИИ

3.1 Кластеризация. Постановка задачи кластеризации. Задачи, решаемые кластеризацией.

Кластеризация.

Кластеризация – одна из задач Data Mining, а кластер – группа похожих объектов. Существует много определений кластеризации, поэтому приведем несколько из них.

Кластеризация – группировка объектов на основе близости их свойств; каждый кластер состоит из схожих объектов. а объекты разных кластеров существенно отличаются;

Кластеризация – процедура. которая любому объекту x ∈ X ставит в соответствие метку кластера y ∈ У. Кластеризацию используют, когда отсутствуют априорные сведения относительно классов, к которым можно отнести объекты исследуемого набора данных, либо кода число объектов велико, что затрудняет их ручной анализ.

Постановка задачи кластеризации.

Постановка задачи кластеризации сложна и неоднозначна, так как:

- оптимальное количество кластеров в общем случае неизвестно;
- выбор меры «похожести» или близости свойств объектов между собой, как и критерия качества кластеризации, часто носит субъективный характер.

Объекты внутри кластера должны быть похожими друг на друга и отличаться от других, которые вошли в другие кластеры. В задачах кластеризации не требуется указание выходной переменной, т.е. имени кластера, а число кластеров, в которые необходимо сгруппировать все множество данных, может быть неизвестным. Выходом кластеризации является не готовый ответ

(например, «плохо» / «удовлетворительно» / «хорошо»), а группы похожих объектов — кластеры. Кластеризация указывает только на схожесть объектов, и не более того. Для объяснения образовавшихся кластеров необходима их дополнительная интерпретация. Кластеризация может использоваться, например, для сегментации и построения профилей клиентов банка, телекоммуникационной или страховой кампаний. Так, в задаче определения групп клиентов при достаточно большом их числе становится трудно подходить к каждому индивидуально, поэтому их удобно объединять в группы — сегменты с однородными признаками. Выделять сегменты можно по нескольким группам признаков, например, по сфере деятельности, географическому расположению, статусу и т.п. После кластеризации можно узнать, какие сегменты наиболее активны, какие приносят наибольшую прибыль, выделить характерные для них признаки. Эффективность работы с клиентами повышается благодаря учету их персональных предпочтений. Задача кластеризации известна давно, и специалисты в различных областях знаний оперируют рядом других терминов — таксономия, сегментация, группировка, автоматическая классификация и др.

В Data Mining используется термин «кластеризация». Например, в аналитике кластеризация применяется для решения следующих задач.

Задачи, решаемые кластеризацией.

Изучение данных. Разбиение множества объектов на схожие группы помогает выявить структуру данных, увеличить наглядность их представления, выдвинуть новые гипотезы, понять, насколько информативны свойства объектов.

Облегчение анализа. При помощи кластеризации можно упростить дальнейшую обработку данных и построение моделей. Каждый кластер обрабатывается индивидуально, и модель создается для каждого кластера индивидуально. В этом смысле кластеризация является подготовительным этапом перед решением других задач Data Mining.

Сжатие данных. В случае, когда данные имеют большой объем (сотни тысяч и миллионы строк), кластеризация позволяет сократить объем хранимых

данных, оставив по одному наиболее типичному представителю от каждого кластера.

Прогнозирование. Кластеры используются не только для краткого описания имеющихся объектов, но и для распознавания новых. Каждый новый объект относится к тому кластеру, присоединение к которому наилучшим образом удовлетворяет критерию качества кластеризации. Далее можно прогнозировать поведение объекта, предположив, что оно будет схожим с поведением других объектов кластера.

Обнаружение аномалий. Кластеризация применяется для выделения нетипичных объектов, которые не присоединяются ни к одному из кластеров.

3.2 Кластеризация. Алгоритм k-means. Шаги алгоритма k-means. Евклидово расстояние. Расстояние Манхеттена.

Кластеризация.

Смотреть вопрос №3.1.

Алгоритм k-means.

Сегодня предложено несколько десятков алгоритмов кластеризации и еще больше их разновидностей. Несмотря на это, в Data Mining применяются в первую очередь понятные и простые в использовании алгоритмы. К таким относится алгоритм k-means — в русскоязычном варианте k-средних (от англ. mean — «среднее значение»). Его основная идея состоит в том, что для выборки данных, содержащей n записей (объектов), задается число кластеров — k, которое должно быть сформировано. Затем алгоритм разбивает все объекты выборки на k разделов ($k < n$), которые и представляют собой кластеры.

Шаги алгоритма k-means.

Алгоритм выполняется в четыре шага:

1) задается число кластеров — k, которое должно быть сформировано из объектов исходной выборки;

2) случайным образом выбирается k записей исходной выборки, которые будут служить начальными центрами кластеров. Начальные точки, из которых потом вырастает кластер, часто называют «семенами» (от англ. seeds — «семена», «посевы»). Каждая такая запись представляет собой своего рода «эмбрион» кластера, состоящий только из одного элемента;

3) для каждой записи исходной выборки определяется ближайший к ней центр кластера. Чтобы определить, в сферу влияния какого центра кластера входит та или иная запись, вычисляется расстояние от каждой записи до каждого центра в многомерном пространстве признаков и выбирается то «семя», для которого данное расстояние минимальное;

4) в анализе данных распространенной оценкой близости между объектами является метрика, или способ задания расстояния. Выбор конкретной метрики зависит от аналитика и конкретной задачи. Наиболее популярные метрики — евклидово расстояние и расстояние Манхэттена. Используя метрики L1 или L2, для каждой записи исходной выборки определяется ближайший к ней центр (центроид) кластера.

Например, если в кластер вошли три записи с наборами признаков (x1, y1), (x2, y2), (x3, y3), то координаты его центроида по метрике L1 будут рассчитываться следующим образом:

$$(x, y) = \left(\frac{(x_1 + x_2 + x_3)}{3}, \frac{(y_1 + y_2 + y_3)}{3} \right);$$

5) старый центр кластера смещается в его центроид.

Таким образом, центроиды становятся новыми центрами кластеров для следующего итерации алгоритма. Шаги 3 и 4 повторяются до тех пор, пока выполнение алгоритма не будет прервано или пока не будет выполнено условие в соответствии с некоторым критерием сходимости.

Остановка алгоритма производится, когда границы кластеров и расположение центроидов перестают изменяться от итерации к итерации, т.е. на

каждой итерации в каждом кластере остается один и тот же набор записей. Алгоритм k-means обычно позволяет находить набор стабильных кластеров за несколько десятков итераций.

Что касается критерия сходимости, то чаще всего используется сумма квадратов ошибок между центроидом кластера и всеми вошедшими в него записями.

Евклидово расстояние.

Евклидово расстояние, или метрика L2, применяется для вычисления расстояний следующее правило по формуле:

$$d_E(X,Y) = \sqrt{\sum_i (x_i - y_i)^2},$$

где $X = (x_1, x_2, \ldots, x_m)$, $Y = (y_1, y_2, \ldots, y_m)$ — векторы значений признаков двух записей.

Поскольку множество точек, равноудаленных от некоторого центра, при использовании евклидовой метрики будут образовывать сферу (или круг в двумерном случае), то кластеры, полученные с использованием евклидова расстояния, также будут иметь форму, близкую к сферической.

Расстояние Манхеттена.

Расстояние Манхеттена, или метрика L1, вычисляется по формуле:

$$d_M(X,Y) = \sum_i |x_i - y_i|.$$

Фактически расстояние Манхеттена — кратчайшее расстояние между двумя точками, пройденное по линиям, параллельным осям координатой системы. Преимущество метрики L1 заключается в том, что она позволяет снизить влияние аномальных значений на работу алгоритмов. Кластеры, построенные на основе расстояния Манхеттена, стремятся к кубической форме.

3.3 Кластеризация. Алгоритм k-means. Евклидово расстояние. Расстояние Манхеттена. Критерий сходимости. Недостатки алгоритма k-means.

Кластеризация.

Смотреть вопрос №3.1.

Алгоритм k-means.

Смотреть вопрос №3.2.

Евклидово расстояние.

Смотреть вопрос №3.2.

Расстояние Манхеттена.

Смотреть вопрос №3.2.

Критерий сходимости.

Остановка алгоритма производится, когда границы кластеров и расположение центроидов перестают изменяться от итерации к итерации, т.е. на каждой итерации в каждом кластере остается один и тот же набор записей. Алгоритм k-means обычно позволяет находить набор стабильных кластеров за несколько десятков итераций.

Что касается критерия сходимости, то чаще всего используется сумма квадратов ошибок между центроидом кластера и всеми вошедшими в него записями.

Недостатки алгоритма k-means.

Один из основных недостатков, присущих алгоритму k-means, — отсутствие четких критериев выбора числа кластеров, целевой функции их инициализации и модификации. Кроме того, он очень чувствителен к шумам в данных и аномальным значениям, поскольку они способны существенно повлиять на среднее значение, используемое при вычислении положений центроидов. Чтобы снизить влияние таких факторов, как шумы и аномальные значения, иногда на каждой итерации используют не среднее значение

признаков, а их медиану. Данная модификация алгоритма называется k-mediods (k-медиан).

3.4 Алгоритм G-means. Недостатки алгоритма k-means.

Алгоритм G-means.

Одним из недостатков алгоритма k-means является отсутствие, как мы уже сказали, ясного Критерия для выбора оптимального числа кластеров. Действительно, пусть множество данных содержит 5 групп, внутри которых объекты похожи, а в различных группах существенно отличаются. Тогда логично задать k = 5, чтобы каждая группа оказалась ассоциирована с отдельным кластером. Но, как правило, такая априорная информация отсутствует и аналитику приходится действовать методом проб и ошибок. Если будет выбрано k = 3, то какие-то две из 5 групп окажутся «распылены» по «чужим» кластерам и не будут обнаружены. Это может привести к потере потенциально ценных знаний. Кроме того, в общем то, мало похожие объекты могут оказаться в одном кластере, что затруднит интерпретацию результатов анализа. Если выбрать большее число кластеров, например, k = 7, то будут сформированы «лишние» кластеры. При этом получится, что достаточно похожие объекты окажутся в различных кластерах. Это проиллюстрировано на рис. 5.4, где знаком «+» отмечены центры кластеров, сформированные обычным алгоритмом k-means.

На рис. 5.4.а, где для 5 групп сформировано всего 3 кластера, можно увидеть, что группы 1 и 5, а также 2 и 3 оказались ассоциированы с одним центром, то есть попали в общий кластер. В результате аналитик может сделать ошибочный вывод о сходстве объектов из групп, объединенных в один кластер. На рис. 5.4.б, где число кластеров было задано слишком большим, наблюдается обратная ситуация, группы 3 и 5 оказались ассоциированы с двумя центрами, и, следовательно, каждая из них разбросана по двум кластерам.

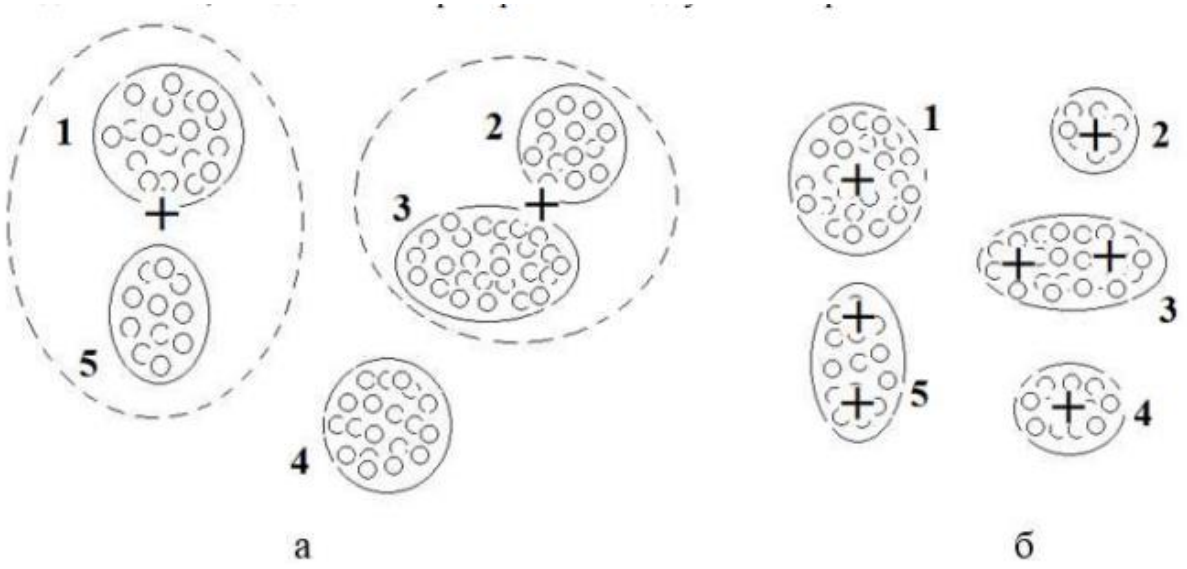


Рис. 5.4. Недостатки алгоритма k-means

Чтобы решить данную проблему, было разработано большое количество алгоритмов, позволяющих производить автоматический выбор числа кластеров, оптимального с точки зрения того или иного критерия. Обычно в них строится несколько моделей для различных значений k, а затем выбирается наиболее подходящая. Примерами могут служить алгоритм X-means, в основе которого лежат байесовские оценки логарифмического правдоподобия; алгоритм, основанный на принципе минимальной длины описания (MDL) и др.

Одним из самых популярных алгоритмов кластеризации с автоматическим выбором числа кластеров является G- means. В его основе лежит предположение о том, что кластеризуемые данные подчиняются некоторому унимодальному закону распределения, например, гауссовскому (откуда и название алгоритма), тогда центр кластера, определяемый как среднее значений признаков, попавших в него объектов, может рассматриваться как мода соответствующего распределения. Если исходные данные описываются унимодальным гауссовским распределением с заданным средним, то можно предположить, что все они относятся к одному кластеру. Если распределение данных не гауссовское, то можно попробовать выполнить разделение на два кластера. Если в этих кластерах распределения окажутся близки к гауссовскому, то можно в первом приближении считать, что k = 2 будет оптимальным. В противном случае будут построены новые модели с большим числом кластеров, и так до тех пор, пока

распределение в каждом из них не окажется достаточно близким к гауссовскому. Такая модель и, соответственно, число кластеров в ней будут считаться оптимальными.

Алгоритм G-means является итеративным, где на каждом шаге с помощью обычного k-means строится модель с определенным числом кластеров. Обычно G-means начинает работу с небольшого значения k, и на каждой итерации оно увеличивается. Как правило, начальное значение k выбирается равным 1. На каждой итерации увеличение k производится за счет разбиения кластеров, в которых данные не соответствуют гауссовскому распределению.

Алгоритм принимает решение о дальнейшем разбиении на основе статистического теста данных, связанных с каждым центроидом. Если при этом будет обнаружено, что они распределены по гауссовскому закону, то дальнейшего смысла в их разбиении нет. Фактически в процессе работы G-means алгоритм k-means будет повторен k раз.

Недостатки алгоритма k-means.

Смотреть вопрос №3.3.

3.5 Кластеризация. Проблемы алгоритмов кластеризации. Выбор числа кластеров.

Кластеризация.

Смотреть вопрос №3.1.

Проблемы алгоритмов кластеризации.

Ранее уже отмечалось, что одно и то же множество объектов можно разбить на несколько кластеров по-разному. Это привело к изобилию алгоритмов кластеризации. Пожалуй, ни одна другая задача Data Mining не имеет в своем арсенале столько алгоритмов и методов решения. Причинами сложившейся ситуации является несколько факторов, имеющих общее объяснение: не существует одного универсального алгоритма кластеризации. Перечислим эти факторы.

1. Неопределенность в выборе критерия качества кластеризации.

В Data Mining при решении задач кластеризации популярны алгоритмы, которые ищут оптимальное разбиение множества данных на группы. Критерий оптимальности определяется видом целевой функции, от которой зависит результат кластеризации.

Например, семейство алгоритмов k-means показывает хорошие результаты, когда данные в пространстве образуют компактные сгустки, четко отличимые друг от друга. Поэтому и критерий качества основан на вычислении расстояний точек до центров кластера.

Главная трудность в выборе критерия качества кластеризации заключается в том, что на практике в условиях, когда объекты описываются десятками и сотнями свойств, становится сложно оценить взаимное расположение объектов и подобрать адекватный алгоритм.

2. Трудность выбора меры близости, обусловленная различной природой данных.

Особенность данных такова, что в таблицах, описывающих свойства объектов, могут присутствовать различные типы данных. Для задачи кластеризации это чаще всего числовые и строковые данные. Строковый тип, в свою очередь, делится на упорядоченный и категориальный. Присутствие тех или иных типов данных в наборе определяет его природу. Назовём набор данных числовым, если он состоит только из целых и вещественных признаков. Для вычисления расстояний между объектами таких наборов чаще всего применяется популярная метрика евклидово расстояние. Назовем набор данных строковым, если он состоит из упорядоченных и категориальных признаков (сюда же относятся логические признаки). Для упорядоченных можно также использовать евклидово расстояние, закодировав значения признака целыми числами. А вот к категориальным типам эта мера не подходит. Здесь нужно применять специальную меру расстояния, например функцию отличия (difference function), которая задается следующим образом:

$$d(x,y) = \begin{cases} 0, & \text{если } x = y \\ 1, & \text{в остальных случаях} \end{cases}$$

где x и y – категориальные значения.

Наборы данных, содержащие признаки, к которым нельзя применять одну и ту же меру расстояния, называются смешанными.

Главная трудность в выборе меры близости состоит в том, что необходимость использования комбинации метрик ухудшает работу алгоритма, а эффективных алгоритмов кластеризации для смешанных наборов данных мало.

3. Различные требуемые машинные ресурсы (память и время).

Алгоритмы кластеризации, как и любые другие, имеют различную вычислительную сложность. Вопрос масштабируемости в кластеризации стоит особенно остро, так как эта задача Data Mining часто выступает первым шагом в анализе: после выделения схожих групп применяются другие методы, для каждой группы строится отдельная модель. В частности, именно из-за больших вычислительных затрат в Data Mining не получили распространение иерархические алгоритмы, которые строят полное дерево вложенных кластеров.

Для кластеризации больших массивов данных, содержащих миллионы строк, разработаны специальные алгоритмы, позволяющие добиваться приемлемого качества за несколько проходов по набору данных. Такие задачи, к примеру, актуальны при сегментации покупателей супермаркета по их чекам.

Получение масштабируемых алгоритмов основано на идее отказа от локальной функции оптимизации. Парное сравнение объектов между собой в алгоритме k-means есть не что иное, как локальная оптимизация: на каждом шаге необходимо рассчитывать расстояние от центра кластера до каждого объекта. Это ведет к большим вычислительным затратам. При задании глобальной функции оптимизации добавление новой точки в кластер не требует больших вычислений: расстояние рассчитывается на основе старого значения, нового объекта и параметров кластера. К сожалению, ни k-means, ни сеть Кохонена не используют глобальную функцию оптимизации.

Выбор числа кластеров.

Хоть и редко, но встречаются случаи, когда точно известно, сколько кластеров нужно выделить. Но чаще всего перед процедурой кластеризации этот вопрос остается открытым. Если алгоритм не поддерживает автоматическое определение оптимального количества кластеров (как, например, G-means), есть несколько эмпирических правил, которые можно применять при условии, что каждый кластер будет в дальнейшем подвергаться содержательной интерпретации аналитиком.

- Двух или трех кластеров, как правило, недостаточно: кластеризация будет слишком грубой, приводящей к потере информации об индивидуальных свойствах объектов.
- Больше десяти кластеров не укладываются в «число Миллера 7 ∓ 2»: аналитику трудно держать в кратковременной памяти столько кластеров.

Поэтому в подавляющем большинстве случаев число кластеров варьируется от 4 до 9.

При взгляде на изобилие алгоритмов кластеризации возникает вопрос, существует ли объективная, естественная кластеризация, или она всегда носит субъективный характер? Не существует. Любая кластеризация субъективна, потому что выполняется на основе конечного подмножества свойств объектов. А выбор этого подмножества всегда субъективен, как и выбор критерия качества и меры близости.

Популярные алгоритмы k-means и сеть Кохонена изначально разрабатывались для числовых данных, и, хотя впоследствии появились их модификации, применяемые к смешанным наборам данных, они все равно лучше решают задачи кластеризации на числовых признаках. Чтобы применять кластеризацию корректно и снизить риск получения результатов моделирования, не имеющих никакого отношения к действительности, необходимо придерживаться следующих правил.

Правило 1. Перед кластеризацией четко обозначьте цели ее проведения: облегчение дальнейшего анализа, сжатие данных и т. п. Кластеризация сама по себе не представляет особой ценности.

Правило 2. Выбирая алгоритм, убедитесь, что он корректно работает с теми данными, которыми вы располагаете для кластеризации. В частности, если присутствуют категориальные признаки, удостоверьтесь, что та реализация алгоритма, которую вы используете, умеет правильно обрабатывать их. Это особенно актуально для алгоритмов k-means и сетей Кохонена (впрочем, и для других, основанных на метриках), в большинстве случаев при меняющих евклидову меру расстояния. Если алгоритм не умеет работать со смешанными наборами данных, постарайтесь сделать набор данных однородным, то есть отказаться от категориальных или числовых признаков.

Правило 3. Обязательно проведите содержательную интерпретацию каждого полученного кластера: постарайтесь понять, почему объекты были сгруппированы в определенный кластер, что их объединяет. Для этого можно использовать визуальный анализ, графики, кластерограммы, статистические характеристики кластеров, карты. Полезно каждому кластеру дать емкое название, состоящее из нескольких слов. Встречаются ситуации, кода алгоритм кластеризации не выделил никаких особых групп. Возможно, набор данных и до кластеризации был однороден, не расслаивался на изолированные подмножества, а кластеризация подтвердила эту гипотезу.

Таким образом, не существует единого универсального алгоритма кластеризации. При использовании любого алгоритма важно пони мать его достоинства, недостатки и ограничения. Только тогда кластеризация будет эффективным инструментом в руках аналитика.

4 ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ: БАЙЕСОВСКИЙ КЛАССИФИКАТОР.

4.1 Простой байесовский классификатор. Теоретические сведения.

Простой байесовский классификатор.

Байесовский подход представляет собой группу алгоритмов классификации, основанных на принципе максимума апостериорной вероятности. Сначала определяется апостериорная вероятность (условная вероятность случайного события при условии того, что известны апостериорные данные, т. е. полученные после опыта.) отношения объекта к каждому из классов, а затем выбирается тот класс, для которого она максимальна. Алгоритм предсказывает вероятность того, что объект или наблюдение относится к определенному классу.

Этот подход к классификации является одним из старейших и до сих пор сохраняет прочные позиции в технологиях анализа данных. Кроме того, он лежит в основе многих удачных алгоритмов классификации. Рассмотрим один из них так называемый простой, или наивный, байесовский классификатор. Это специальный случай байесовского классификатора, в котором используется предположение о статистической независимости признаков, описывающих классифицируемые объекты. Такое предположение существенно упрощает задачу, поскольку вместо одной многомерной плотности вероятности по всем признакам достаточно оценить несколько одномерных плотностей. К сожалению, на практике предположение о независимости признаков редко выполняется, является «наивным», что и дало название методу.

Основные преимущества наивного байесовского классификатора: легкость программной реализации и низкие вычислительные затраты при обучении и классификации. В тех редких случаях, когда признаки действительно

независимы (или близки к этому), он почти оптимален. Главный его недостаток относительно низкое качество классификации в большинстве реальных задач. Поэтому чаще всего его используют либо как примитивный эталон для сравнения различных моделей, либо как блок для построения более сложных алгоритмов.

Рассмотрим базовые принципы работы простого байесовского классификатора.

Теоретические сведения.

Пусть имеется объект или наблюдение X, класс которого неизвестен. Пусть также имеется гипотеза H, согласно которой X относится к некоторому классу C. Для задачи классификации можно определить вероятность $P(H|X)$, то есть вероятность того, что гипотеза H для X справедлива. $P(H|X)$ называется условной вероятностью того, что гипотеза H верна при условии, что классифицируется объект X, или апостериорной вероятностью.

Предположим, что объектами классификации являются фрукты, которые описываются их цветом и размером. Определим объект X как красный и круглый и выдвинем гипотезу H, что это яблоко. Тогда условная вероятность $P(H|X)$ отражает меру уверенности в том, что объект X является яблоком при условии, что он красный и круглый. Кроме условной (апостериорной) вероятности, рассмотрим так называемую априорную вероятность $P(H)$. В нашем примере это вероятность того, что любой наблюдаемый объект является яблоком, безотносительно к тому, как он выглядит. Таким образом, апостериорная вероятность основана на большей информации, чем априорная, не предполагающая зависимость от свойств объекта X.

Аналогично $P(H|X)$ апостериорная вероятность X при условии H, или вероятность того, что X является красным и круглым, если известно, что это яблоко, $P(X)$априорная вероятность X. В нашем примере это просто вероятность того, что объект является красным и круглым. Вероятности $P(X)$, $P(H)$ и $P(X|H)$ могут быть оценены на основе наблюдаемых данных.

Для вычисления апостериорной вероятности на основе $P(X)$, $P(H)$ и $P(X|H)$ используется формула Байеса:

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)}$$

Алгоритм работы простого байесовского классификатора содержит следующие шаги.

1. Пусть исходное множество данных S содержит атрибуты A_1, A_2, ..., A_n Тогда каждый объект или наблюдение $X \in S$ будет представлено своим набором значений этих атрибутов $x_1, x_2, \ldots, x_n$ где x_i значение, которое принимает атрибут A_i в данном наблюдении.

2. Предположим, что задано m классов $C = \{C_1, C_2, \ldots C_m\}$ и наблюдение X, для которого класс неизвестен. Классификатор должен определить, что X относится к классу, который имеет наибольшую апостериорную вероятность $P(H|X)$. Простой байесовский классификатор относит наблюдение X к классу $C_k (k = 1, \ldots, m)$ тогда и только тогда, когда выполняется условие $P(C_k|X) > P(C_j|X)$ для любых $1 \le j \le m$:.т.к, $k \ne j$.

По формуле Байеса:

$$P(C_k|X) = \frac{P(X|C_k)P(C_k)}{P(X)} \quad (2)$$

3. Поскольку вероятность $P(X)$ для всех классов одинакова, максимизировать требуется только числитель формулы (2), Если априорная вероятность класса $P(C_k)$ неизвестна, то можно предположить, что классы равновероятны, $P(C_1) = P(C_2) = \ldots = P(C_m)$, и, следовательно, мы должны выбрать максимальную вероятность $P(C_k|X)$.

Заметим, что априорные вероятности классов могут быть оценены как $P(C_k) = s_k/s$, где s_k – число наблюдений обучающей выборки, которые относятся к классу C_k, а s – общее число обучающих примеров.

4. Если исходное множество данных содержит большое количество атрибутов, то определение $P(X|C_k)$ может потребовать значительных вычислительных затрат. Чтобы их уменьшить, используется «наивное» предположение о независимости признаков. То есть для набора атрибутов $X = (x_1, x_2, \ldots, x_n)$ можно записать:

$$P(X|C_k) = P(x_1|C_k) \times P(x_2|C_k) \times \ldots \times P(x_n|C_k) \quad (3)$$

Вероятности, стоящие в правой части формулы (3), могут быть определены из обучающего набора данных для следующих случаев.

Атрибут A является категориальным, тогда $P(X|C_k) = {}^{s_{ik}}/_{s_k}$, где s_{ik} – общее число наблюдений класса C_i, в которых A_i принимает значение x_i, а s_k – общее число наблюдении, относящихся к классу C_k.

Атрибут A является непрерывным, тогда предполагается, что его значения подчиняются закону распределения Гаусса:

$$P(x_i|C_k) = \frac{1}{\sqrt{2\pi\sigma}} exp\left(\frac{(x-m)^2}{2\sigma^2}\right)$$

где m и σ^2 – математическое ожидание и дисперсия значений атрибута A_i для наблюдений, относящихся к классу C_k.

При классификации неизвестного наблюдения объект X будет относиться к классу, для которого $P(X|C_i) \times P(C_i)$ принимает наибольшее значение.

4.2 Байесовские сети. Байесовские сети с двумя и тремя переменными. Примеры.

Байесовские сети.

Проблема заключается в том, что распределения, которые нас интересуют, обычно слишком сложные, чтобы их можно было максимизировать напрямую, аналитически. В них слишком много переменных, между переменными слишком сложные связи. Но, с другой стороны, часто в них есть дополнительная структура, которую можно использовать, структура в виде независимостей ($p(x, y) = p(x)p(y)$) и условных независимостей ($p(x, y|\ z) = p(x|z)p(y|z)$) некоторых переменных.

Итак, давайте рассмотрим один из самых удобных способов представлять большие и сложные распределения вероятностей – байесовские сети доверия, которые в последнее время чаще называются просто направленными графическими моделями.

Байесовская сеть – это направленный граф без направленных циклов (это очень важное условие!), в котором вершины соответствуют переменным в распределении, а рёбра соединяют «связанные» переменные. В каждом узле задано условное распределение узла при условии своих родителей $p(x|parents(x))$ (родители), и граф байесовской сети означает, что большое совместное распределение раскладывается в произведение этих условных распределений.

Вот, например, граф, соответствующий наивному Байесу (рис.1):

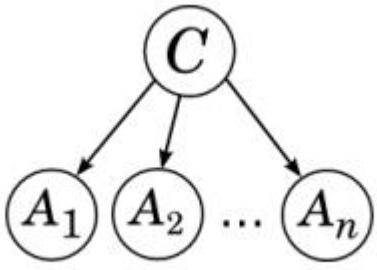


Рис.1. Граф, соответствующий наивному Байесу

Он соответствует разложению $p(A1, \dots, An, C) = p(C)p(A1\ |C)p(A2\ |C)\ \dots\ p(An\ |C)$): у C нет предков, так что мы берём его безусловное распределение, а

каждый из Ai «растёт» непосредственно из C и больше ни с кем не связан. Мы уже знаем, что в этом случае все атрибуты Ai условно независимы при условии категории C: $p(Ai\ ,\ Aj\ |C) = p(Ai\ |C)p(Aj\ |C)$ Давайте теперь рассмотрим все простейшие варианты байесовской сети и посмотрим, каким условиям независимости между переменными они соответствуют.

Байесовские сети с двумя и тремя переменными.

Начнём с сети из двух переменных, x и y. Здесь всего два варианта: либо между x и y нет ребра, либо есть. Если ребра нет, это просто значит, что x и y независимы (рис.2), ведь такой граф соответствует разложению $p(x,\ y) = p(x)p(y)$.

Рис.2. Переменные x и y независимы

А если ребро есть (пусть оно идёт из x в y, это не важно), мы получаем разложение $p(x,\ y) = p(x)p(y|x)$, которое буквально по определению условной вероятности тупо верно всегда, для любого распределения $p(x,\ y)$. Таким образом, граф из двух вершин с ребром не даёт нам новой информации (рис.3).

Теперь переходим к сетям из трёх переменных x, y и z. Самый простой случай – когда рёбер совсем нет.

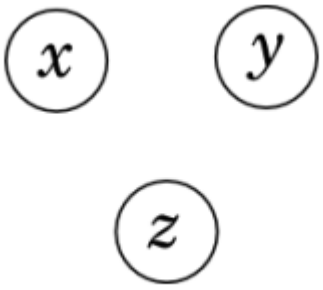


Рис.3. Переменные x, y и z независимы

Как и с двумя переменными, это значит, что x, y и z просто независимы: $p(x,\ y,\ z) = p(x)p(y)p(z)$.Другой простой случай – когда между переменными проведены все рёбра (рис.4).

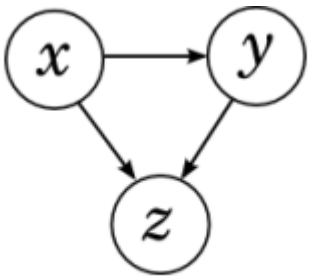


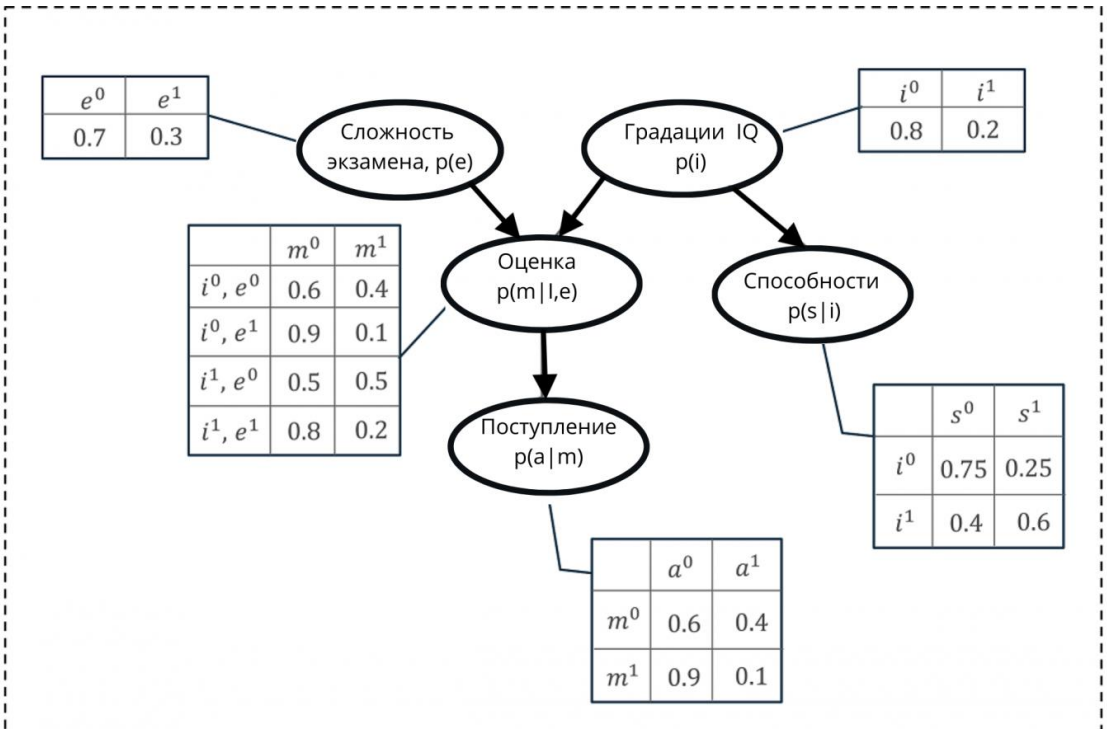
Рис.4. Переменные x, y и z независимы

Этот случай тоже аналогичен рассмотренному выше; пусть, например, рёбра идут из x в y и z, а также из y в z. Получаем разложение $p(x, y, z) = p(x)p(y|x)p(z|\text{x, y})$ которое, опять же, верно всегда, для любого совместного распределения $p(x, y, z)$. В этой формуле можно было бы выбирать переменные в любом порядке, ничего не изменилось бы. Обратите внимание, что направленные циклы в байесовских сетях запрещены, и в результате вариантов, как можно провести все рёбра, всего шесть, а не восемь.

Рассмотрим три более интересных случая – это и будут те «кирпичики», из которых можно составить любую байесовскую сеть. К счастью, для этого достаточно рассмотреть графы на трёх переменных – всё остальное будет обобщаться из них.

В примерах ниже будут интуитивно интерпретировать ребро, стрелочку между двумя переменными, как «x влияет на y», т.е. по сути, как причинноследственную связь. На самом деле это, конечно, не совсем так.

Примеры.



4.3 Байесовские сети. Последовательная связь. Примеры.

Байесовские сети.

Смотреть вопрос №4.2.

Последовательная связь.

Начнём с последовательной связи между переменными: x «влияет на» y, а y, в свою очередь, «влияет на» z (рис.5).

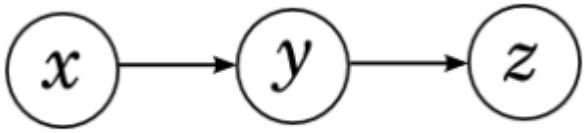


Рис.5. Последовательная связь переменных x, y и z

Такой граф изображает разложение $p(x, y, z) = p(x)p(y|x)p(z|y)$

Интуитивно это соответствует последовательной причинно-следственной связи: если вы будете бегать зимой без шапки, вы простудитесь, а если простудитесь, у вас поднимется температура. Очевидно, что x и y, а также y и z друг с другом связаны, между ними даже непосредственно стрелочки проведены. Связаны ли между собой в такой сети x и z, зависимы ли эти переменные? Конечно! Если вы бегаете зимой без шапки, вероятность получить высокую

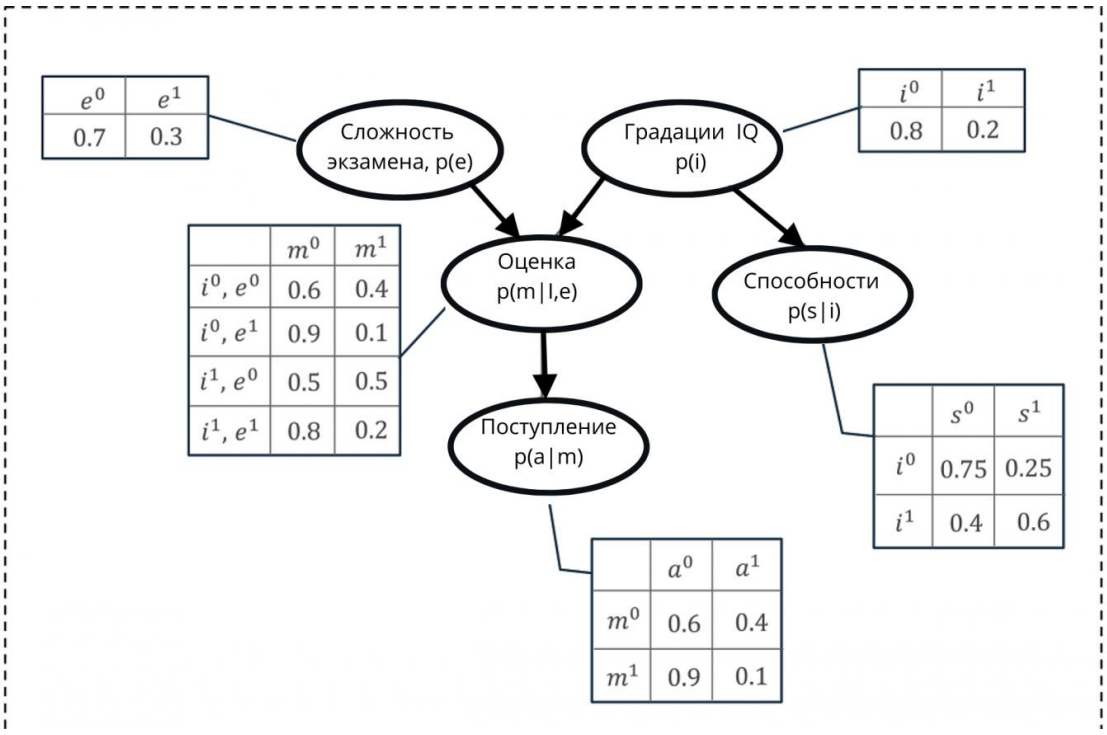
температуру повышается. Однако в такой сети x и z связаны только через y, и если мы уже знаем значение y, x и z становятся независимыми: если вы уже знаете, что простудились, совершенно не важно, чем это было вызвано, температура теперь повысится (или не повысится) именно от простуды.

Формально это соответствует условной независимости x и z при условии y; давайте это проверим:

$$p(x,z|y) = \frac{p(x,y,z)}{p(y)} = \frac{p(x)p(y|x)p(z|y)}{p(y)} = p(x|y)p(z|y)$$

где первое равенство – это определение условной вероятности, второе – наше разложение, а третье – применение теоремы Байеса. Итак, последовательная связь между тремя переменными говорит нам о том, что крайние переменные условно независимы при условии средней. Всё очень логично и достаточно прямолинейно.

Примеры.



4.4 Байесовские сети. Расходящаяся связь. Примеры.

Байесовские сети.

Смотреть вопрос №4.2.

Расходящаяся связь.

Следующий возможный вариант – расходящаяся связь: x «влияет» и на y, и на z. (рис.6).

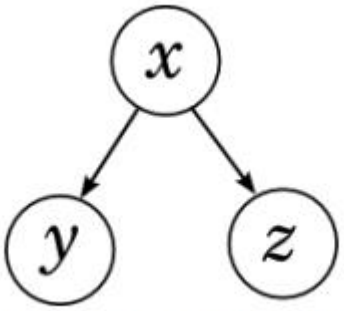


Рис.6. Расходящаяся связь переменных x, y и z

Такой граф изображает разложение $p(x, y, z) = p(x)p(y|x)p(z|x)$.

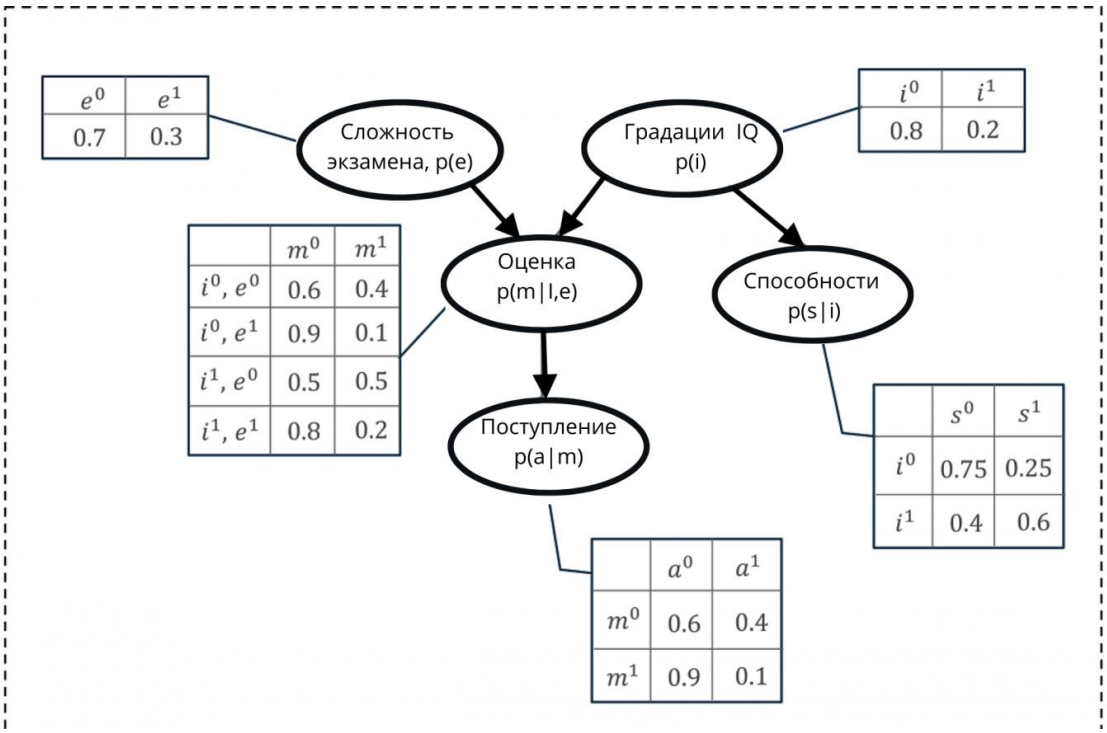
Интуитивно это соответствует двум следствиям из одной и той же причины: если вы простудитесь, у вас может подняться температура, а также может начаться насморк. Как и в предыдущем случае, очевидно, что x и y, а также x и z зависимы, и вопрос заключается в зависимости между y и z. Опять же, очевидно, что эти переменные зависимы: если у вас насморк, это повышает вероятность того, что вы простудились, а значит, вероятность высокой температуры тоже повышается. Однако в такой сети, подобно предыдущему случаю, y и z связаны только через x, и если мы уже знаем значение общей причины x, y и z становятся независимыми: если вы уже знаете, что простудились, насморк и температура становятся независимы.

Формально это соответствует условной независимости y и z при условии x; проверить это ещё проще, чем для последовательной связи:

$$p(y, z|x) = \frac{p(x, y, z)}{p(y)} = \frac{p(x)p(y|x)p(z|x)}{p(y)} = p(y|x)p(z|x)$$

Итак, расходящаяся связь между тремя переменными говорит нам о том, что «следствия» условно независимы при условии своей «общей причины». Если причина известна, то следствия становятся независимы; пока причина неизвестна, следствия через неё связаны.

Примеры.



4.5 Байесовские сети. Сходящаяся связь. Примеры.

Байесовские сети.

Смотреть вопрос №4.2.

Сходящаяся связь.

У нас остался только один возможный вариант связи между тремя переменными: сходящаяся связь, когда x и y вместе «влияют на» z.

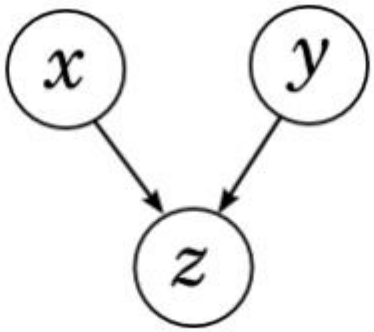


Рис.6. Сходящаяся связь переменных x,y и z

Разложение здесь получается такое: $p(x, y, z) = p(x)p(y)p(z|x, y)$

Это ситуация, в которой у одного и того же следствия могут быть две разные причины: например, температура может быть следствием простуды, а может – отравления. Зависимы простуда и отравление? Нет! В этой ситуации,

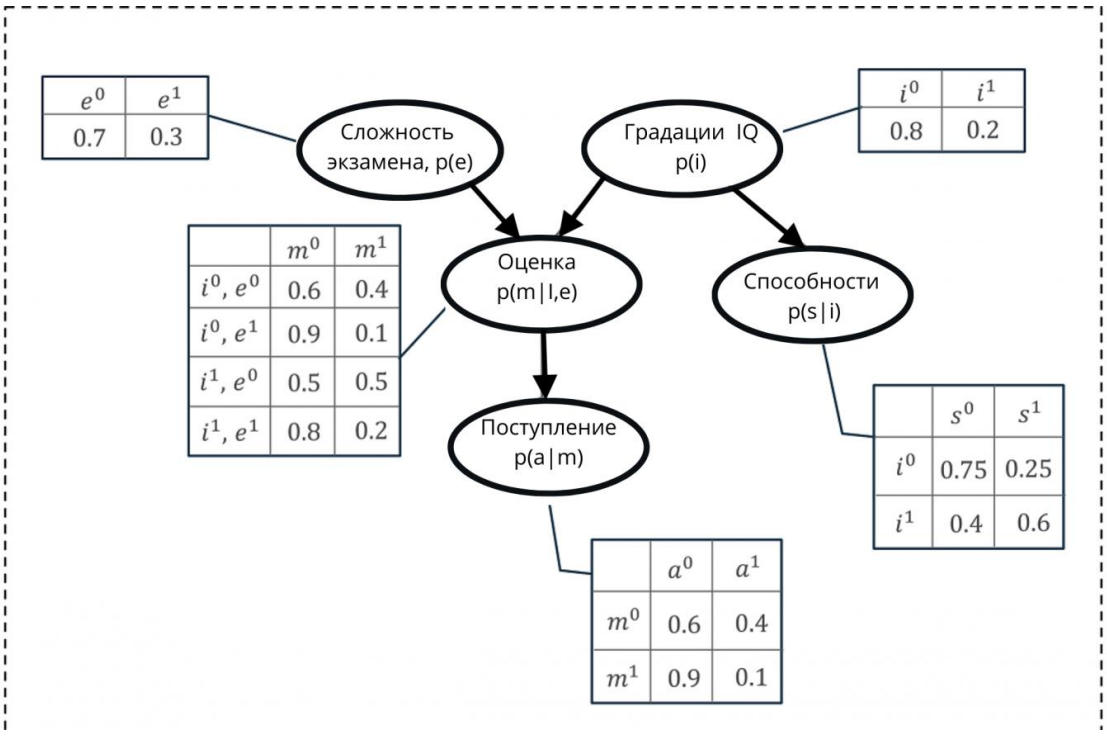
пока общее следствие неизвестно, две причины никак не связаны друг с другом, и это очень легко проверить формально:

$$p(x,y) = \sum_z p(x,y,z) = \sum_z p(x), p(y), p(z|x,y) = p(x)p(y)$$

Однако если «общее следствие» z становится известным, ситуация меняется. Теперь общие причины известного следствия начинают влиять друг на друга. Предположим, что вы знаете, что у вас температура. Это сильно повышает вероятность, как простуды, так и отравления. Однако если вы теперь узнаете, что отравились, вероятность простуды уменьшится – симптом «уже объяснён» одной из возможных причин, и вторая становится менее вероятной. Таким образом, при сходящейся связи две «причины» независимы, но только до тех пор, пока значение их «общего следствия» неизвестно; если же общее следствие получает означивание, причины становятся зависимыми. Но есть один нюанс: когда на пути встречается сходящаяся связь, недостаточно посмотреть только на её «общее следствие», чтобы определить независимость. На самом деле, если даже у z означивания нету, но оно есть у одного из её потомков (возможно, достаточно далёких), две причины всё равно станут зависимы. Интуитивно это тоже легко понять: например, пусть мы не наблюдаем собственно температуру, а наблюдаем её потомка – показания градусника. На свете, наверное, бывают неисправные градусники, так что это тоже некая вероятностная связь. Однако наблюдение показаний градусника точно так же делает простуду и отравление зависимыми. Последовательная, сходящаяся и расходящаяся связи – это те три кирпичика, из которых состоит любой ациклический направленный граф. И наших рассуждений вполне достаточно для того, чтобы обобщить результаты об условной зависимости и независимости на все такие графы. Конечно, здесь не время и не место для того, чтобы формально доказывать общие теоремы, но результат достаточно предсказуем – предположим, что вам нужно в большом графе проверить, независимы ли две вершины (или даже два множества вершин). Для этого вы смотрите на все пути в графе (без учёта стрелочек), соединяющие

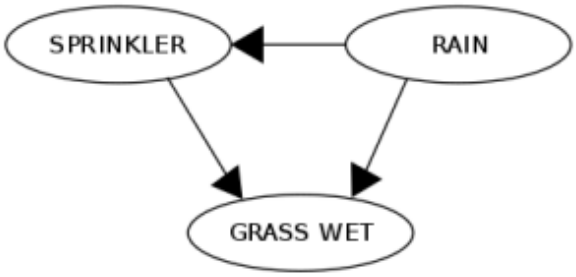
эти два множества вершин. Каждый из этих путей можно «разорвать» одной из вышеописанных конструкций: например, последовательная связь разорвётся, если в середине у неё есть означивание (значение переменной известно), а сходящаяся связь разорвётся, если, наоборот, означивания нет (причём нет ни в самой вершине из пути, ни в её потомках). Если в результате все пути окажутся разорваны, значит, множества переменных действительно независимы; если нет – нет.

Примеры.



Предположим, что может быть две причины, по которым трава может стать мокрой (GRASS WET): сработала дождевальная установка (SPRINKLER), либо прошёл дождь (RAIN). Также предположим, что дождь влияет на работу дождевальной машины (во время дождя установка не включается). Тогда ситуация может быть смоделирована проиллюстрированной Байесовской сетью. Каждая из трёх переменных может принимать лишь одно из двух возможных значений: T (правда — true) и F (ложь — false), с вероятностями, указанными в таблицах на иллюстрации.

	SPRINKLER	
RAIN	T	F
F	0.4	0.6
T	0.01	0.99



	RAIN	
T	F	
0.2	0.8	

		GRASS WET	
SPRINKLER	RAIN	T	F
F	F	0.0	1.0
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

Совместная вероятность функции:

$$\mathrm{P}(G, S, R) = \mathrm{P}(G|S, R)\mathrm{P}(S|R)\mathrm{P}(R)$$

где имена трёх переменных означают G = Трава мокрая (Grass wet), S = Дождевальная установка (Sprinkler), и R = Дождь (Rain). Модель может ответить на такие вопросы как «Какова вероятность того, что прошел дождь, если трава мокрая?» используя формулу условной вероятности и суммируя переменные:

$$\mathrm{P}(R = T \mid G = T) = \frac{\mathrm{P}(G = T, R = T)}{\mathrm{P}(G = T)} = \frac{\sum_{S \in \{T,F\}} \mathrm{P}(G = T, S, R = T)}{\sum_{S,R \in \{T,F\}} \mathrm{P}(G = T, S, R)}$$

$$= \frac{(0.99 \times 0.01 \times 0.2 = 0.00198_{TTT}) + (0.8 \times 0.99 \times 0.2 = 0.1584_{TFT})}{0.00198_{TTT} + 0.288_{TTF} + 0.1584_{TFT} + 0_{TFF}} \approx 35.77\%.$$

5 ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ: ДЕРЕВО РЕШЕНИЙ

5.1 Дерево решений. Понятия, встречающиеся в теории деревьев решений. Разделение на классы. Пример.

Дерево решений.

Деревья решений (decision trees) относятся к числу самых популярных и мощных инструментов Data Mining, позволяющих эффективно решать задачи классификации и регрессии. В отличие от методов, использующих статистический подход, таких как классификатор Байеса, линейная и логистическая регрессия, деревья решений основаны на машинном обучении и в большинстве случаев не требуют предположений о статистическом распределении значений признаков. В основе деревьев решений лежат решающие правила вида «если... то...», которые могут быть сформулированы на естественном языке. Поэтому деревья решений являются наиболее наглядными и легко интерпретируемыми моделями.

Понятия, встречающиеся в теории деревьев решений.

Таблица 1.1. Понятия, встречающиеся в теории деревьев решений

Название	Описание
Объект	Пример, шаблон, наблюдение, запись
Атрибут	Признак, независимая переменная, свойство, входное поле
Метка класса	Зависимая переменная, целевая переменная, выходное поле
Узел	Внутренний узел дерева
Лист	Конечный узел дерева, узел решения
Проверка	Условие в узле

Разделение на классы.

Атрибутами в теории деревьев решений называются признаки, описывающие классифицируемые объекты. В основе работы деревьев решений лежит процесс рекурсивного разбиения исходного множества наблюдений или объектов на подмножества, ассоциированные с классами. Разбиение

производится с помощью решающих правил, в которых осуществляется проверка значений атрибутов по заданному условию. Рекурсивными называются алгоритмы, которые работают в пошаговом режиме, при этом на каждом последующем шаге используются результаты, полученные на предыдущем шаге.

Пример.

Рассмотрим главную идею алгоритмов построения деревьев решений на примере. Пусть требуется предсказать возврат или невозврат кредита с помощью набора решающих правил на основе единственного атрибута Возраст клиента. Для этого будем использовать множество наблюдений, в каждом из которых указывается возраст, а также факт возврата/невозврата кредита, графически такое множество наблюдений представлено на рис. 1.1. Условно примем, что объект в форме круга указывает на дефолт, в форме прямоугольника на возврат по кредиту, а внутри каждого объекта указан возраст. Необходимо разбить множество объектов на подмножества таким образом, чтобы в каждое из них попали объекты только одного класса.

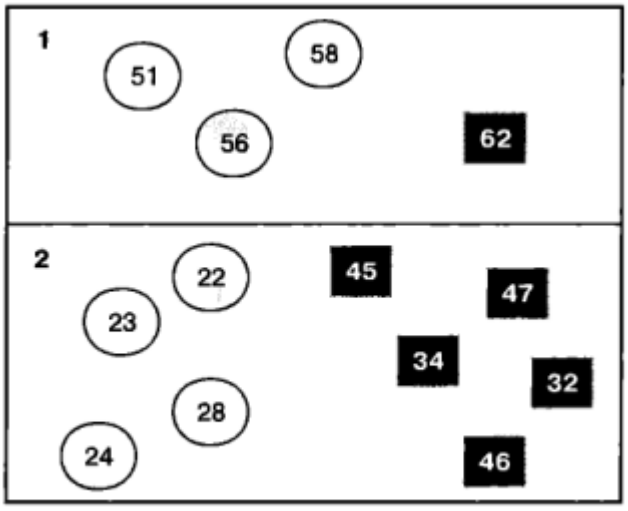


Рис. 1.1. Разделение на классы

Выберем некоторое значение возрастного порога, например равное 50, и разобьем исходное множество на два подмножества в соответствии с условием Возраст> 50. В результате разбиения в одном подмножестве окажутся все записи, для которых значение атрибута Возраст больше 50, а во втором меньше 50. На рис. 1.1 данные подмножества обозначены номерами 1 и 2 соответственно. Легко

увидеть, что выбор возрастного порога 50 не позволил получить подмножества, содержащие только объекты одного класса, поэтому для решения задачи применяется разбиение полученных подмножеств. Поскольку для этого имеется только один атрибут Возраст, будем использовать его и в дальнейшем, но в условиях выберем другой порог. Например, для подмножества 1 применим порог – 60, а для подмножества 2 – 30. Результаты повторного разбиения представлены на рис. 1.2.

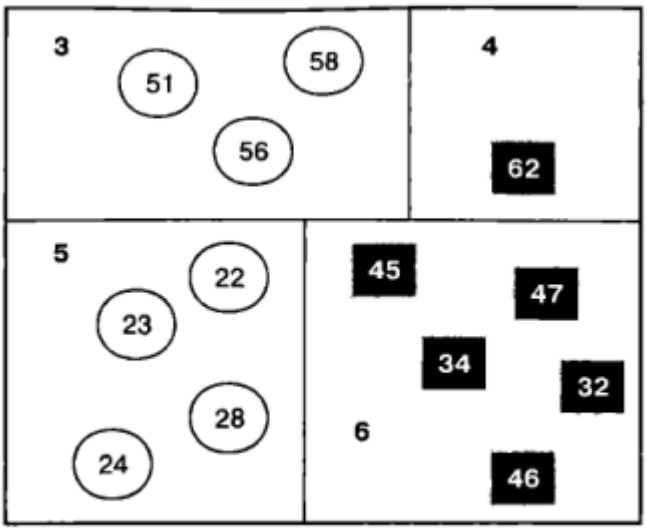


Рис. 1.2. Продолжение деления на классы

На рис. 1.2 можно увидеть, что задача решена: исходное множество удалось разбить на чистые подмножества, содержащие только наблюдения одного класса. Дерево, реализующее данную процедуру, представлено на рис. 1.3.

Рис. 1.3. Сформированное дерево решений

Сопоставление рис. 1.2 и 1.3 показывает, что подмножества 4 и 6 ассоциированы с классом добросовестных клиентов, а 3 и 5 с классом клиентов,

не погасивших кредит. Применяя построенную модель к новым клиентам, мы можем предсказать риск, связанный с выдачей им кредита, на основе того, в какое из подмножеств модель поместит соответствующую запись.

Конечно, приведенный пример тривиален. В реальности задача классификации по одному атрибуту встречается очень редко, поскольку для эффективного разделения на классы требуется несколько атрибутов. Но даже этот пример позволяет увидеть причину привлекательности деревьев решений. Они не только классифицируют объекты и наблюдения, но и объясняют, почему объект был отнесен к данному классу. Так, если с помощью дерева решений было предсказано, что вероятность возврата кредита данным клиентам слишком мала и на этом основании в выдаче кредита было отказано, то можно не только принять решение, но и объяснить его причину. В нашем случае это возраст клиента, обладая высокой объясняющей способностью, деревья решений могут использоваться и как эффективные классификаторы, и как инструмент исследования предметной области. Таким образом, мы получили систему правил вида «если…то…», которые позволяют принять решение относительно принадлежности объекта к определенному классу. Решающие правила образуют иерархическую древовидную структуру, дающую возможность выполнять классификацию объектов и наблюдений. Эта структура и называется деревом решений.

5.2 Дерево решений. Условия построения дерева решений. Факторы популярности дерева решений.

Дерево решений.

Смотреть вопрос №5.1.

Условия построения дерева решений.

Для эффективного построения дерева решений должны выполняться следующие условия:

- Описание атрибутов. Анализируемые данные должны быть

представлены в виде структурированного набора, в котором вся информация об объекте или наблюдении должна быть выражена совокупностью атрибутов.

- Предварительное определение классов. Категории, к которым относятся наблюдения (метки классов), должны быть заданы предварительно, то есть имеет место обучение с учителем.
- Различимость классов. Должна обеспечиваться принципиальная возможность установления факта принадлежности или непринадлежности примера к определенному классу. При этом количество примеров должно быть намного больше, чем количество классов.
- Полнота данных. Обучающее множество должно содержать достаточно большое количество различных примеров. Необходимая численность зависит от таких факторов, как количество признаков и классов, сложность классификационной модели и т.д.

Факторы популярности дерева решений.

Деревья решений стали одним из наиболее популярных методов Data Mining, используемых при решении задач классификации. Это обусловлено следующими факторами:

- Деревья решений – это модели, основанные на обучении. Процесс обучения сравнительно прост в настройке и управлении.
- Процесс обучения деревьев решений быстр и эффективен.
- Деревья решений универсальны способны решать задачи как классификации, так и регрессии.
- Деревья решений обладают высокой объясняющей способностью и интерпретируемостью.

5.3 Дерево решений. Структура дерева решений. Выбор атрибута разбиения в узле.

Дерево решений.

Смотреть вопрос №5.1.

Структура дерева решений.

Как можно увидеть на рис. 1.3. структура деревьев решений проста и в целом аналогично древовидным иерархическим структурам, используемым в других областях Data Mining, например деревьям ассоциативных правил. В состав деревьев решений входят два вида объектов узлы (node) и листья (leaf). В узлах содержатся правила, с помощью которых производится проверка атрибутов и множество объектов в данном узле разбивается на подмножества. Листья – это конечные узлы дерева, в которых содержатся подмножества, ассоциированные с классами. Основным отличием листа от узла является то, что в листе не производится проверка, разбивающая ассоциированное с ним подмножество и, соответственно, нет ветвления. В принципе, листом может быть объявлен любой узел, если принято решение, что множество в узле достаточно однородно в плане классовой принадлежности объектов и дальнейшее разбиение не имеет смысла, поскольку не приведет к значимому увеличению точности классификации, а только усложнит дерево.

В дереве, представленном на рис. 1.3. объекты с номерами 1 и 2 узлы, а с номерами 3, 4, 5 и 6 – листья. Обратим внимание на то, что в дереве имеется по два листа, ассоциированных с одним классом. В этом нет никакого противоречия, просто для классификации объектов в них использовались различные способы проверки. Для каждого листа в дереве имеется уникальный путь. Начальный узел дерева является входным: через него проходят все объекты, предъявляемые дереву. Обычно входной узел называют корневым узлом (root node). Следовательно, дерево растет сверху вниз. Узлы и листья, подчиненные узлу более высокого иерархического уровня, называются потомками, или дочерними узлами, а тот узел по отношению к ним предком, или

родительским узлом. Обобщенная структура дерева проиллюстрирована на рис. 1.4.

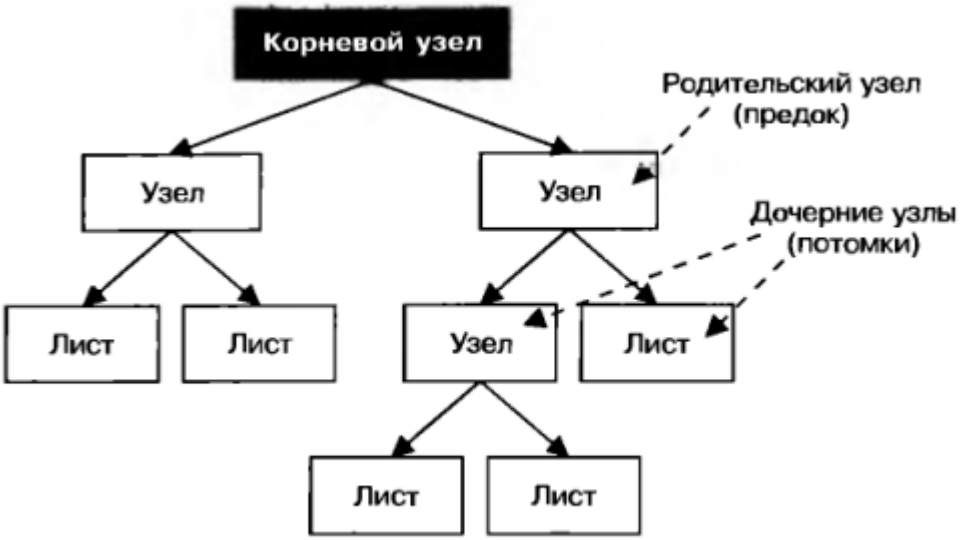


Рис. 1.4. Узлы и листья в дереве решений

Как и любая модель Data Mining, дерево решений строится на основе обучающего множества. Атрибуты могут быть как числовыми (непрерывными), так и категориальными (дискретными). Одно из полей обязательно должно содержать независимую переменную метку класса. Иными словами, для каждой записи в обучающем множестве должна быть задана метка класса, определяющая классовую принадлежность связанного с ней объекта.

В процессе построения дерева решений формируются решающие правила и для каждого из них создается узел. Для каждого узла нужно выбрать атрибут, по которому будет производиться проверка правила. Его принято называть атрибутом ветвления, или атрибутом разбиения (splitting attribute), и от того, насколько удачно он выбран, зависит классифицирующая сила правила. Метод, в соответствии с которым осуществляется выбор атрибута ветвления на каждом шаге, называется алгоритмом построения дерева решений. Разработано достаточно много таких алгоритмов. Сформулируем общую цель, которая должна преследоваться при выборе атрибута ветвления: очередной выбранный атрибут должен обеспечивать наилучшее разбиение в узле. Наилучшим разбиением считается то, которое позволяет классифицировать наибольшее число примеров и создавать максимально чистые подмножества, в которых

примесь объектов другого класса (то есть не ассоциированного с данным узлом или листом) минимальна.

Хотя между алгоритмами построения деревьев решений имеются существенные различия, все они основаны на одной и той же процедуре рекурсивном разбиении данных на все более малые труппы таким образом, чтобы каждое новое поколение узлов содержало больше примеров одного класса, чем родительский.

Выбор атрибута разбиения в узле.

Процесс создания дерева начинается с подготовки обучающего множества. В итоге будет построено дерево, которое назначает класс (или вероятность принадлежности к классу) для выходного поля новых записей на основе значений входных переменных. Мерой оценки возможного разбиения является так называемая чистота (purity), под которой понимается отсутствие примесей. Существует несколько способов определения чистоты, но все они имеют один и тот же смысл. Низкая чистота означает, что в подмножестве представлены объекты, относящиеся к различным классам. Высокая чистота свидетельствует о том, что члены отдельного класса доминируют. Наилучшим разбиением можно назвать то, которое дает наибольшее увеличение чистоты дочерних узлов относительно родительского. Кроме того, хорошее разбиение должно создавать узлы примерно одинакового размера или как минимум не создавать узлы, содержащие всего несколько записей. Рассмотрим рис. 1.5.

В исходном множестве представлена смесь объектов различной формы (треугольники и круги) в пропорции 1:1 (10 кругов и 10 треугольников). Разбиение слева признано плохим, потому что оно не увеличивает чистоту результирующих узлов: пропорция объектов обоих классов в них сохраняется и также составляет 1:1 (5 кругов и 5 треугольников). Во втором случае плохого разбиения (справа) с помощью условия В родительском узле удалось добиться доминирования класса в одном из дочерних узлов (круги), но к классу было отнесено только два объекта, Такое разбиение неудачно по двум причинам. Во-

первых, правило, с помощью которого было получено это разбиение, имеет очень низкую значимость, то есть относится к малому числу примеров.

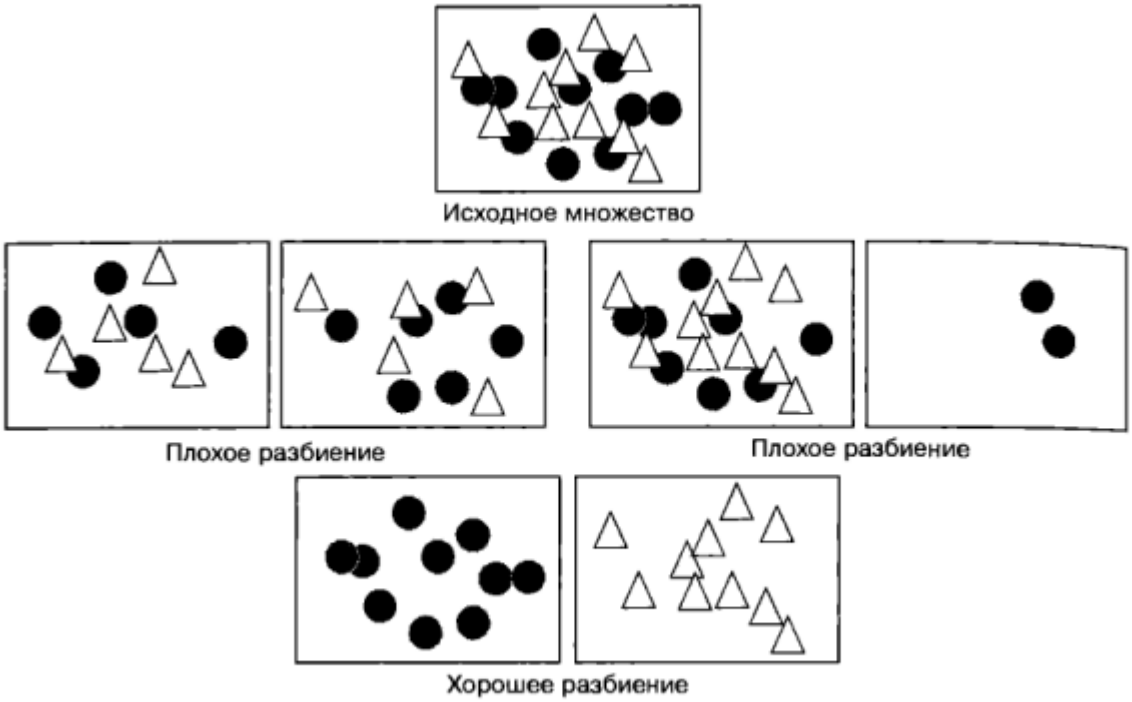


Рис. 1.5. Различные варианты разбиений

Во-вторых, чистота другого узла осталась низкой: соотношение объектов в нем 0,8:1 (8 кругов и 10 треугольников). И хотя в целом чистота относительно родительского узла улучшилась (масс треугольников стал слабо доминирующим), это не позволяет говорить о положительном результате, Наконец, случай хорошего разбиения обеспечивает абсолютную чистоту обоих дочерних узлов, поскольку в каждый из них распределены объекты только одного класса.

Алгоритмы построения деревьев решений являются «жадными». Они развиваются путем рассмотрения каждого входного атрибута по очереди и оценивают увеличение чистоты, которое обеспечило разбиение с помощью данного атрибута.

5.4 Дерево решений. Процесс рекурсивного разбиения (разделяй и властвуй). Варианты разбиения.

Дерево решений.

Смотреть вопрос №5.1.

Процесс рекурсивного разбиения (разделяй и властвуй).

Процесс рекурсивного разбиения подмножеств в узлах дерева решений получил название «разделяй и властвуй!». В его основе лежит следующий принцип. Пусть задано множество T, в котором определены классы $\{C1, C2, \ldots Ck\}$. Тогда существуют три возможных варианта разбиения.

1, Множество T содержит два примера или более, которые относятся к одному классу Cj. Деревом решений для множества T будет лист, идентифицирующий класс. Это тривиальный случай, который на практике не представляет интереса. Графически он может быть интерпретирован, как показано на рис. 1.6.

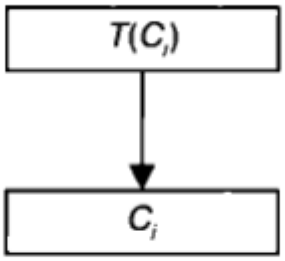


Рис. 1.6. Иллюстрация первого случая

2. Множество T не содержит примеров, то есть является пустым деревом решений, также будет лист, но класс, ассоциированный с данным листом, должен быть определен из другого множества, например родительского, Этот случай иллюстрируется на рис. 1.7.

Рис. 9.7. Иллюстрация второго случая

После разбиения по атрибуту Доход ветвление продолжилось по атрибуту Образование.

Как должен вести себя алгоритм, если в обучающем множестве нет ни одного наблюдения, где Доход = Высокий и Образование = Среднее? Алгоритм должен создать три дочерних узла, и в $T3$ окажется пустое множество. Тогда сгенерируется узел, который будет ассоциирован с классом, наиболее часто встречающимся в родительском множестве.

3. Множество T содержит примеры, относящиеся к различным классам. Задача заключается в разделении T на подмножества, ассоциированные с классами.

Выбирается один из входных атрибутов, принимающий два отличных друг от друга значения $v1$, $v2$ … vn, или более, после чего T разбивается на подмножества $\{T1\ (v1\ ), T2\ (v2\ ) \ldots Tn\ (vn\ )\}$, где каждое подмножество T: содержит все примеры из исходного множества, в которых выбранный атрибут принимает значение vi . Данная процедура будет рекурсивно повторяться до тех пор, пока подмножества не будут содержать примеры только одного класса. Этот случай проиллюстрирован на рис. 1.8.

Пусть требуется определить класс риска, связанный с выдачей кредита клиенту, на основе двух атрибутов Доход и Имущество. Атрибуты принимают по два значения Высокий и Низкий. На первом шаге алгоритм разобьет исходное множество на два подмножества, в одно из которых будут помещены клиенты с высокими доходами, а в другое с низкими. Предположим, что все клиенты с высокими доходами относятся к классу низкого кредитного риска, поэтому

соответствующее подмножество будет объявлено листом и дальнейшее разбиение в нем остановится. Другое подмножество, в котором остались клиенты с низким доходом, подвергается проверке с помощью атрибута Имущество. Логика проста: если клиент не имеет высокого дохода, но имеет достаточно имущества, чтобы покрыть кредит при невозможности его выплатить, то кредитный риск может быть определен как низкий.

Данный случай является наиболее распространенным и практически важным в процессе построения деревьев решений.

Процесс построения дерева решений не является однозначно определенным. Для различных атрибутов и даже для различного порядка их применения могут быть сгенерированы различные деревья решений. В идеальном случае разбиения должны быть такими, чтобы результирующее дерево оказалось наиболее компактны.

Встает вопрос: почему бы тогда не исследовать все возможные деревья и не выбрать самое компактное? К сожалению, во многих реальных задачах перебор всех возможных деревьев решений приводит к комбинаторному взрыву.

Эффективность разбиения оценивается по чистоте полученных дочерних узлов относительно целевой переменной. От ее типа и будет зависеть выбор предпочтительного критерия разбиения. Если выходная переменная является категориальной, то необходимо использовать такие критерии, как индекс Джини, прирост информации или тест хи-квадрат. Если выходная переменная является непрерывной, то для оценки эффективности разбиения используются метод уменьшения дисперсии или F-тест (рис. 1.9).

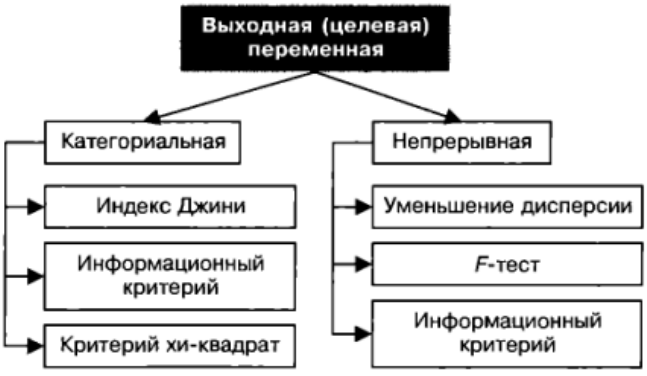


Рис. 1.9. Критерии разбиения

Варианты разбиения.

Смотреть вопрос №5.4, прошлый пункт.

5.5 Дерево решений. Полное дерево решений. Меры эффективности деревьев решений.

Дерево решений.

Смотреть вопрос №5.1.

Полное дерево решений.

Процесс роста дерева решений начинается с разбиения корневого узла на два потомка или более, каждый из которых рекурсивно подвергается дальнейшему разбиению. При этом каждый раз все входные атрибуты рассматриваются как потенциальные атрибуты разбиения, даже те, что уже использовались ранее, кроме атрибутов, все значения в которых одинаковы. Такие атрибуты исключаются из рассмотрения, поскольку с их помощью нельзя разделить при меры различных классов. Когда больше не удается обнаружить разбиения, значимо повышающие чистоту дочерних узлов, или когда число примеров в узле достигает некоторого заданного минимума, процесс разбиения для данной ветви заканчивается. Узел объявляется листом.

Когда найти в дереве какие-либо новые разбиения, повышающие его точность, не удается и разбиение прекращается по всем ветвям, это значит, что построено полное дерево.

Меры эффективности деревьев решений.

В целом эффективность деревьев решений определяется с помощью тестового множества набора примеров, которые не использовались при построении дерева. Дереву предъявляется набор тестовых примеров и вычисляется, для какого процента примеров класс был определен правильно. Это позволяет оценить ошибку классификации, а также качество решения задачи классификации или регрессии отдельных ветвей в дереве.

Каждый узел или лист дерева обладает следующими характеристиками:

- количество примеров, попавших в узел (лист);
- доли примеров, относящихся к каждому из классов;
- число классифицированных примеров (для листьев);
- процент примеров, верно классифицированных данным узлом (листом).

Особый интерес представляет количество правильно классифицированных записей в данном узле или листе. Поэтому для оценки качества классификации вводятся два показателя поддержка (support) и достоверность (confidence).

Поддержка определяется как отношение числа правильно классифицированных примеров в данном узле или листе к общему числу попавших в него примеров, то есть:

$$S = \frac{N_{кл}}{N_{общ}}$$

Очевидно, что значение поддержки может изменяться от 0 до 1.

Достоверность определяется как отношение числа правильно классифицированных примеров к числу ошибочно классифицированных, то есть:

$$C = \frac{N_{кл}}{N_{ош}}$$

Значит, чем больше число правильно классифицированных примеров в узле, тем выше достоверность. Поддержка и достоверность могут использоваться в качестве параметров построения дерева решений. Например, можно задать, что разбиение должно производиться до тех пор, пока в узле не будет достигнут заданный порог поддержки.

5.6 Дерево решений. Критерии выбора наилучших атрибутов ветвления. Индекс Джини.

Дерево решений.

Смотреть вопрос №5.1.

Критерии выбора наилучших атрибутов ветвления.

Существует множество различных подходов к оценке потенциальных разбиений. При этом методы, разработанные в рамках теории машинного обучения, в основном сосредотачиваются на повышении чистоты результирующих подмножеств, в то время как статистические методы фокусируются на статистической значимости различий между распределением значений выходной переменной в узлах. Альтернативные методы разбиения часто приводят к построению совершенно разных деревьев, которые, впрочем, функционируют примерно одинаково. Различные меры оценки чистоты ведут к выбору различных атрибутов разбиения, но, поскольку все меры основаны на одной и той же идее, полученные с их помощью модели будут похожи.

Для выбора атрибута ветвления в случае категориальной целевой переменной используются такие методы, как:

- индекс Джини, или метод разнообразия выборки (Gini-index);
- энтропия, или прирост информации (information gain);
- отношение прироста информации (gain-ratio);
- тест хи-квадрат (chi-square test).

Данные методы применимы и тогда, когда целевая переменная является непрерывной, но в этом случае необходимо предварительно выполнить ее квантование.

Индекс Джини.

Один из популярных критериев разбиения получил название индекса Джини в честь итальянского статистика и экономиста. Эта мера основана на исследовании разнообразия совокупности. Она определяет вероятность того что два объекта, случайным образом выбранные из одной совокупности, относятся к одному классу.

Очевидно, что для абсолютно чистой выборки данная вероятность равна 1. Мера Джини для узла представляет собой простую сумму квадратов долей

классов в узле. В случае, представленном на рис.1.10, родительский узел содержит одинаковое количество светлых и темных кругов.

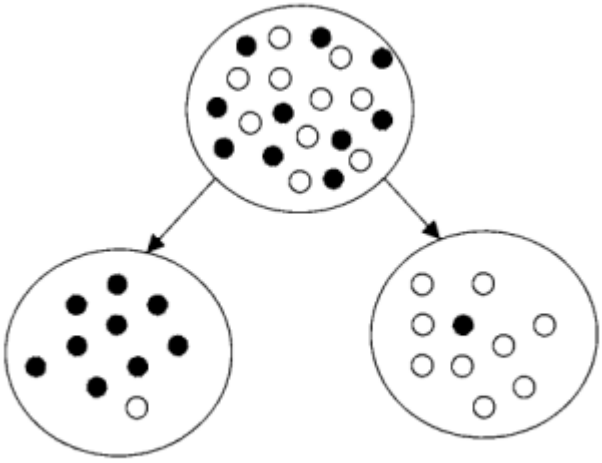


Рис. 1.10. Пример разбиения

Для узла, в котором содержится равное число объектов, относящихся к двум классам, можно записать: $0{,}5^2 + 0{,}5^2 = 0{,}5$. Такой результат вполне ожидаем, поскольку вероятность того, что случайно дважды будет выбран пример, относящийся к одному классу, равна 1/2. Индекс Джини для любого из двух результирующих узлов будет $0{,}1^2 + 0{,}9^2 = 0{,}82$. Идеально чистый узел имеет значение индекса Джини, равное 1. Узел, в котором содержится равное число объектов двух классов, будет иметь значение индекса 0,5. Таким образом, предпочтение следует отдать тому атрибуту, который обеспечит максимальное значение индекса Джини.

5.7 Дерево решений. Критерии выбора наилучших атрибутов ветвления. Уменьшение энтропии, или прирост информации.

Дерево решений.

Смотреть вопрос №5.1.

Критерии выбора наилучших атрибутов ветвления.

Смотреть вопрос №5.6.

Уменьшение энтропии, или прирост информации.

Из теории информации известно, что чем больше состояний может принимать некоторая система, тем сложнее ее описать и тем больше информации для этого потребуется. Примером может служить кодирование цифровых изображений. Для представления каждой точки черно-белого изображения достаточно одного бита. Если он принимает значение 1, точка черная, а если 0 – белая. Когда нужно описать изображение, содержащее 256 оттенков, для задания каждой его точки потребуется $k = log_2 256 == 8$ бит и т. д.

Аналогичный подход можно применить к описанию подмножества в некотором узле дерева решений. Если лист совершенно чистый, то все попавшие в него примеры относятся к одному классу и его описание будет очень простым. Но если лист содержит смесь объектов различных классов, то его описание усложнится и для этого потребуется большее количество информации. В теории существует мера количества информации – энтропия. Она отражает степень неупорядоченности системы.

В контексте нашего рассмотрения энтропия – это мера разнообразия классов в узле. Проще говоря, будем считать, что это мера, определяющая количество вопросов «да/нет», на которые нужно ответить, чтобы определить состояние системы. Если существует 16 возможных состояний, это даст $log_2(16)$, или 4 бита, необходимых для описания всех состояний.

Целью разбиения узла в дереве решений является получение дочерних узлов с более однородным классовым составом. В результате разбиения должны образовываться узлы с меньшим разнообразием состояний выходной переменной. Следовательно, энтропия падает, а количество внутренней информации в узле растет. Уменьшение энтропии эквивалентно приросту информации.

Формально энтропия определённого узла T дерева решений определяется формулой

$$Info(T) = \sum_{i=1}^{k} p_j \, log_2 \, p_j$$

и представляет собой сумму всех вероятностей появления примеров, относящихся к -классу, умноженную на логарифм этой вероятности. Поскольку вероятность меньше или равна 1, значение логарифма всегда будет отрицательным. На практике эта сумма обычно умножается на –1 для получения положительного числа.

Энтропия всего разбиения – это сумма энтропий всех узлов, умноженных на долю записей каждого узла в числе записей исходного множества.

6 ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ: ЛИНЕЙНАЯ РЕГРЕССИЯ

6.1 Регрессия. Линейная и логистическая регрессия. Примеры.

Регрессия.

Классификация и регрессия являются одними из важнейших задач анализа данных. Их объединение не случайно, поскольку в самой постановке задач классификации и регрессии много общего. Действительно, как классификационная, так и регрессионная модель находят закономерности между входными и выходными переменными. Но если входные и выходные переменные модели непрерывные перед нами задача регрессии. Если выходная переменная одна, и она является дискретной (метка класса), то речь идет о задаче классификации.

Линейная и логистическая регрессия.

Задача линейной регрессии заключается в нахождении коэффициентов уравнения линейной регрессии, которое имеет вид:

$$y = b_0 + b_1 x_1 + b_2 x_2 + \cdots + b_n x_n \qquad (4.1)$$

где

y – выходная (зависимая) переменная модели;

$x_1, x_2, \ldots x_n$ – входные (независимые) переменные;

b_i – коэффициенты линейной регрессии, называемые также параметрами модели (b_0 – свободный член).

Задача линейной регрессии заключается в подборе коэффициентов b_i уравнения (4.1) таким образом, чтобы на заданный входной вектор $X = (x_1, x_2, \ldots x_n)$ регрессионная модель формировала желаемое выходное значение y.

Одним из наиболее востребованных приложений линейной регрессии является прогнозирование. В этом случае входными переменными модели xi являются наблюдения из прошлого (предикторы), а y – прогнозируемое значение. Несмотря на свою универсальность, линейная регрессионная модель не всегда пригодна для качественного предсказания зависимой переменной.

Когда для решения задачи строят модель линейной регрессии, на значения зависимой переменной обычно не налагают никаких ограничений. Но на практике такие ограничения могут быть весьма существенными. Например, выходная переменная может быть категориальной или бинарной. В таких случаях приходится использовать различные специальные модификации регрессии, одной из которых является логистическая регрессия, предназначенная для пред сказания зависимой переменной, принимающей значения в интервале от 0 до 1. Такая ситуация характерна для задач оценки вероятности некоторого события на основе значений независимых переменных.

Кроме того, логистическая регрессия используется для решения задач бинарной классификации, в которых выходная переменная может принимать только два значения 0 или 1, «Да» или «Нет» и т. д.

Таким образом, логистическая регрессия служит не для предсказания значений зависимой переменной, а скорее для оценки вероятности того, что зависимая переменная примет заданное значение.

Предположим, что выходная переменная y может принимать два возможных значения 0 и 1. Основываясь на доступных данных, можно вычислить, вероятности их появления: $P(y = 0) = 1 - p$; $P(y = 1) = p$. Иными словами, вероятность появления одного значения равна 1 минус вероятность появления другого, поскольку одно из них появится обязательно и их общая вероятность равна 1. Для определения этих вероятностей используется логистическая регрессия:

$$log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n \qquad (4.2)$$

Правая часть формулы (4.2) эквивалентна обычному уравнению линейной регрессии (4.1). Однако вместо непрерывной выходной переменной y в левой части отношения вероятностей двух взаимоисключающих событий (в нашем примере вероятность появления 0 и вероятность появления 1).

Функция вида log (p/(1-p)) называется логит-преобразованием и обозначается logit(p). Использование логит-преобразование позволяет ограничить диапазон изменения выходной переменной в пределах [0; 1].

Примеры.

Из головы.

6.2 Простая линейная регрессия. Линия регрессии. Уравнение регрессии. Коэффициенты регрессии.

Простая линейная регрессия.

В Data Mining существует большой класс задач, где требуется установить зависимость между признаками (атрибутами, показателями), которые описывают исследуемый процесс или объект предметной области. Для этого строятся различные модели, в которых данные признаки выступают в качестве переменных. Если модель будет корректно отражать зависимость между входными и выходными переменными, то с помощью такой модели можно будет предсказывать значения выходной переменной по заданным значениям входных.

Если предположить, что зависимость между переменными линейная, то для построения модели достаточно провести прямую линию, проходящую через «облако» точек, соответствующих наблюдениям. Тогда наклон линии покажет, насколько уменьшатся продажи при увеличении цены. Но таких линий можно построить бесконечно много, и только одна из них обеспечит оптимальную оценку объемов продаж. Естественным было бы провести линию таким образом, чтобы рассеяние вдоль нее точек, соответствующих реальным наблюдениям, было минимальным. На практике линию строят так, чтобы сумма квадратов отклонений наблюдаемых значений от оцененных с помощью данной линейной зависимости была минимальной, то есть:

$$\sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \to min$$

где n – число наблюдений;

$\hat{y}_i$ – оценка выходного значения для -го наблюдения, полученная с помощью модели;

y_i – реально наблюдаемое значение объема продаж.

Данный метод приближения линейной зависимости между входными и выходными переменными известен как метод наименьших квадратов (МНК), а линия, построенная с его помощью, называется линией регрессии.

Линия регрессии.

Линия регрессия – это прямая наилучшего приближения для множества пар значений входной и выходной переменной (x, y), выбираемая таким образом, чтобы сумма квадратов расстояний от точек (xi , yi) до этой прямой, измеренных вертикально (то есть вдоль оси y), была минимальна.

Уравнение регрессии.

Уравнение, описывающее линию регрессии, называется уравнением регрессии:

$$\hat{y} = b_0 + b_1 x \quad (4.3)$$

где $\hat{y}$– оценка значения выходной переменной;

b_0 - коэффициент, определяющий точку пересечения линии с осью y, называемый также свободным членом Коэффициент b_1 определяет наклон линии относительно оси x (иногда его называют *угловым коэффициентом*). Проще говоря,

b_1 – это величина, на которую изменяется значение выходной переменной y при изменении входной переменной x на единицу, Коэффициенты линейного уравнения b_0 и b_1 называются *коэффициентами регрессии*.

Таким образом, задача построения модели линейной регрессии сводится к нахождению таких коэффициентов b_0 и b_1, для которых сумма квадратов ошибок, то есть разностей между реально наблюдаемыми значениями выходной переменной y_i и их оценками $\hat{y}$, была бы минимальна. Уравнение регрессии с учетом ошибки между наблюдаемым и оцененным значениями будет

$$\hat{y} = b_0 + b_1 x + \varepsilon$$

где ε – ошибка.

Коэффициенты регрессии.

Смысл коэффициентов уравнения регрессии следующий: $b0$ – это значение выходной переменной y при значении входной переменной $x = 0$. Значит, при цене картофеля, равной нулю, оценка объемов продаж составит 3213,6 кг. Однако данная формальная интерпретация явно противоречит здравому смыслу, поскольку если раздавать картофель бесплатно, то купят любое его доступное количество. Такая ситуация возникла из-за того, что в исходной выборке наблюдений отсутствуют значения x, близкие к нулю. Отсюда вытекает одно из ограничений линейной регрессии: линию регрессии (и, соответственно, описывающее ее уравнение) следует считать подходящей аппроксимацией

некоторой реальной функции только в том диапазоне изменений входной переменной x, в котором распределены исходные наблюдения. В противном случае результаты могут оказаться непредсказуемыми.

Значение коэффициента наклона линии регрессии $b1$ можно интерпретировать как среднюю величину изменения значения выходной переменной при изменении значения входной переменной на единицу. В нашем примере это означает, что при увеличении цены за один килограмм картофеля на одну денежную единицу можно ожидать уменьшения спроса в среднем на 145,4 кг. Линия регрессии для найденного нами уравнения представлена на рис. 4.2.

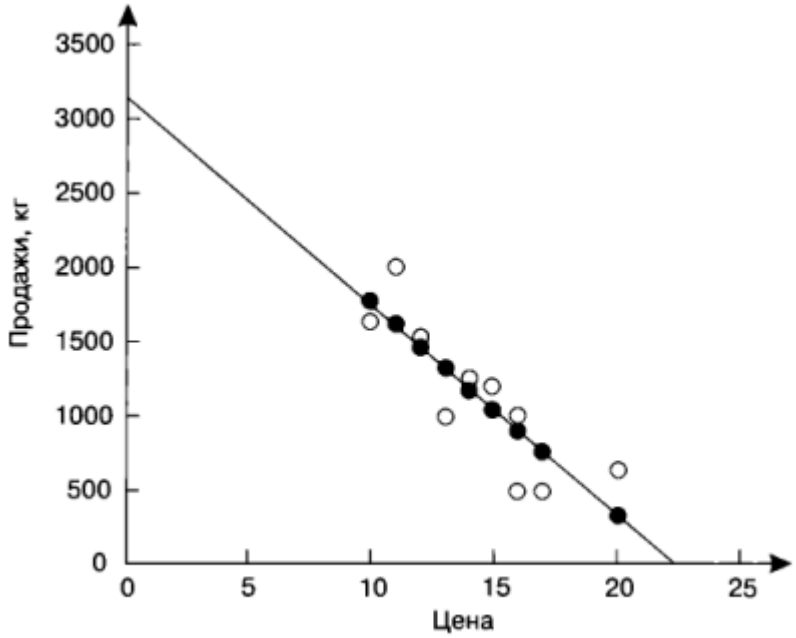


Рис. 4.2. Линия регрессии для примера из табл. 4.1

Для линии регрессии сумма квадратов вертикальных расстояний между точками данных (светлые точки) и линией должна быть меньше, чем аналогичная сумма квадратов для любой другой прямой.

6.3 Регрессия. Оценка соответствия простой линейной регрессии реальным данным. Стандартная ошибка.

Регрессия.

Смотреть вопрос №6.1.

Оценка соответствия простой линейной регрессии реальным данным.

Линия регрессии должна аппроксимировать линейную зависимость между входной и выходной переменными модели, Однако при этом возникает вопрос, насколько линейная аппроксимация соответствует наблюдаемым данным. Чтобы определить это, введем в рассмотрение два показателя стандартную ошибку оценивания $\widehat{E_{ст}}$ и коэффициент детерминации, обозначаемый r^2.

В статистике мерой разброса случайной величины относительно среднего значения является стандартное отклонение. Аналогично в качестве меры разброса точек наблюдений относительно линии регрессии можно использовать стандартную ошибку оценивания, которая показывает среднюю величину отклонения точек исходных данных от линии регрессии вдоль оси y. Стандартная ошибка равна корню квадратному среднеквадратической ошибки ($ECKO$), то есть сумме квадратов разностей между реальным и оцененным значениями, вычисленной по всем наблюдениям и отнесенной к числу степеней свободы выборки:

$$E_{CKO} = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n - m - 1}$$

где m – количество независимых переменных, которое для простой линейной регрессии равно 1.

Стандартная ошибка.

E_{CKO} можно рассматривать как меру изменчивости выходной переменной, объясняемую регрессией. Тогда стандартная ошибка оценивания определится следующим образом:

$$E_{ст} = \sqrt{E_{CKO}}$$

Значение стандартной ошибки Eст позволяет оценить степень рассогласования оценок, полученных с помощью регрессии, и реальных наблюдений аналогично тому, как стандартное отклонение позволяет оценить в статистическом анализе степень разброса случайной величины относительно среднего. Чем меньше стандартная ошибка оценивания, тем лучше работает модель.

6.4 Регрессия. Изменчивость выходной переменной. Меры, характеризующие поведение выходной переменной. Примеры.

Регрессия.

Смотреть вопрос №6.1.

Изменчивость выходной переменной.

Чтобы оценить степень соответствия регрессии реальным данным, полезно ввести меры, количественно характеризующие поведение выходной переменной. В качестве таких мер используются три квадратичные суммы: общая (полная) Q, регрессионная Q_R и ошибки (остаточная) Q_E.

Меры, характеризующие поведение выходной переменной.

Чтобы оценить степень соответствия регрессии реальным данным, полезно ввести меры, количественно характеризующие поведение выходной переменной. В качестве таких мер используются три квадратичные суммы: общая (полная) Q, регрессионная Q_R и ошибки (остаточная) Q_E. Чтобы пояснить смысл данных величин, представим наблюдаемое значение выходной переменной в виде $y = \hat{y} + (y - \hat{y})$, то есть наблюдаемое значение равно сумме euo оценки и ошибки оценивания, Данное выражение может быть записано в виде $y = (b_0 + b_1 x) + (y - b_0 - b_1 x)$.Здесь y – наблюдаемое значение, $(b_0 + b_1 x)$ – член, описывающий долю изменчивости выходной переменной, объясняемой регрессией, $(y - b_0 - b_1 x)$ – член, описывающий отклонение выходной переменной от линии регрессии (остаток). Если все точки данных лежат непосредственно на линии регрессии (идеальное соответствие), то остатки будут равны 0. Если из обеих частей предыдущего выражения вычесть среднее значение $\bar{y}$, то есть:

$$y - \bar{y} = (\hat{y} - \bar{y}) + (y - \hat{y})$$

то можно показать, что суммы квадратов складываются следующим образом:

$$\sum_{i=1}^{n} (y_i - \bar{y})^2 = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^{n} (y - \hat{y})^2$$

или, используя введенные выше обозначения для квадратичных сумм:

$$Q = Q_R + Q_E$$

Таким образом,

$$Q = \sum_{i=1}^{n} (y_i - \bar{y})^2 ;\ Q_R = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 ;\ Q_E = \sum_{i=1}^{n} (y - \hat{y})^2$$

Следовательно, полная изменчивость выходной переменной складывается из части, объясняемой регрессией (линейной зависимостью), и ошибки, то есть части, не объясненной регрессией.

Нужно ли ожидать, что полная квадратичная сумма Q, полученная только при использовании среднего значения , $\bar{y}$ будет больше или меньше, чем остаточная сумма Q_E, полученная при оценивании с учетом входной переменной? Используя данные табл. 4.2, получим, что $Q = 228$ окажется намного больше, чем $Q_E = 12$ (табл.4.3).

Примеры.

Из головы.

6.5 Регрессия. Коэффициент детерминации. Коэффициент корреляции. Примеры.

Регрессия.

Смотреть вопрос №6.1.

Коэффициент детерминации.

Введем понятие коэффициента детерминации r^2, который показывает степень согласия регрессии как приближения линейного отношения между входной и выходной переменными с реальными данными:

$$r^2 = \frac{Q_R}{Q}$$

Значение коэффициента детерминации максимально, когда имеет место идеальное соответствие: все точки данных лежат точно на прямой регрессии. В этом случае квадратичная сумма Q_E оценки, полученной с помощью регрессии, равна 0. Тогда $Q = Q_R$ и $r^2 = Q_R/Q = 1$. Максимальное значение коэффициента детерминации, равное 1, имеет место только тогда, когда уравнение регрессии идеально описывает связь между входной и выходной переменными.

Чтобы определить максимальное значение коэффициента детерминации, предположим, что регрессия совсем не улучшает точность оценки по сравнению с использованием среднего значения, то есть не объясняет изменчивость выходной переменной. В этом случае $Q_R = 0$, а значит, и $r^2 = 0$. Таким образом, коэффициент детерминации может изменяться от 0 до 1 включительно. При этом чем выше значение r^2, тем больше регрессионная модель соответствует реальным данным. Значения r^2, близкие к 1, означают очень хорошее соответствие регрессионной модели реальным данным, а значения, близкие к 0, очень плохое.

Коэффициент корреляции.

Еще одной мерой, используемой для количественного описания линейной зависимости между двумя числовыми переменными, является коэффициент корреляции, который определяется следующим образом:

$$r = \frac{\sum(x-\bar{x})(y-\bar{y})}{(n-1)\sigma_x\sigma_y}$$

где σ_x и σ_y стандартные отклонения соответствующих переменных. Значение коэффициента корреляции всегда расположено в диапазоне от −1 до 1 включительно и может быть интерпретировано следующим образом:

- если коэффициент корреляции близок к 1, то между переменными имеет место сильная положительная корреляция. Иными словами, наблюдается высокая степень зависимости входной и выходной переменных (если значения входной переменной x возрастают, то и значения выходной переменной y также будут увеличиваться);
- если коэффициент корреляции близок k − 1, это означает, что между переменными наблюдается отрицательная корреляция: поведение выходной переменной будет противоположным поведению входной (когда значение x возрастает y уменьшается, и наоборот);
- промежуточные значения указывают на слабую корреляцию между переменными и, соответственно, на низкую зависимость между ними: поведение входной переменной x совсем (или почти совсем) не будет влиять на поведение y.

Для приближенной оценки можно воспользоваться следующей шкалой (табл. 4.4).

Таблица 8.4. Шкала соответствия для коэффициента корреляции

Коэффициент корреляции, r	Корреляция
$0{,}6 < r < 1$	Высокая положительная
$0{,}3 \le r \le 0{,}6$	Средняя положительная
$-0{,}3 < r < 0{,}3$	Корреляция отсутствует
$-0{,}6 \le r \le -0{,}3$	Средняя отрицательная
$1 < r < 0{,}6$	Средняя отрицательная

Примеры.

Из головы.

7 ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ: ЛОГИЧЕСКАЯ РЕГРЕССИЯ

7.1 Основы логистической регрессии. Простой пример логистической регрессии.

Основы логистической регрессии.

Линейная регрессия используется для моделирования линейных зависимостей между непрерывной выходной переменной и набором входных переменных. При анализе данных часто встречаются задачи, где выходная переменная является категориальной и тогда использование линейной регрессии затруднено. Поэтому при поиске связей между набором входных переменных и категориальной выходной переменной получила распространение логистическая регрессия. Рассмотрим применение логистической регрессии для случая бинарной выходной переменной (переменной, которая может принимать только два значения), хотя можно использовать данный метод и в случае, когда выходная переменная принимает более чем два значения.

Простой пример логистической регрессии.

Предположим, что врача интересует зависимость между возрастом пациента и наличием (1) или отсутствием (0) некоторого заболевания. Данные, собранные по 20 пациентам, представлены в табл. 1.9, а соответствующий график на рис. 1.12. Таким образом, в задаче используется бинарная выходная переменная y, которая может принимать только два значения: 0 и 1. Иногда такие переменные называют дихотомическими.

Таблица 1.9. Данные о пациентах

№ пациента	Возраст, x	Наличие заболевания, y
1	25	0
2	29	0
3	30	0
4	31	0
5	32	0
6	41	0
7	41	0
8	42	0
9	44	1
10	49	1
11	50	0
12	59	1
13	60	0
14	62	0

На рисунке сплошной линией представлена прямая простой линейной регрессии, построенная для данных из табл. 1.12, а пунктиром кривая логистической регрессии. Также для обеих кривых показана ошибка оценивания для пациента №11 (x = 50, y = 0). Линия логистической регрессии, в отличие от линейной, не является прямой.

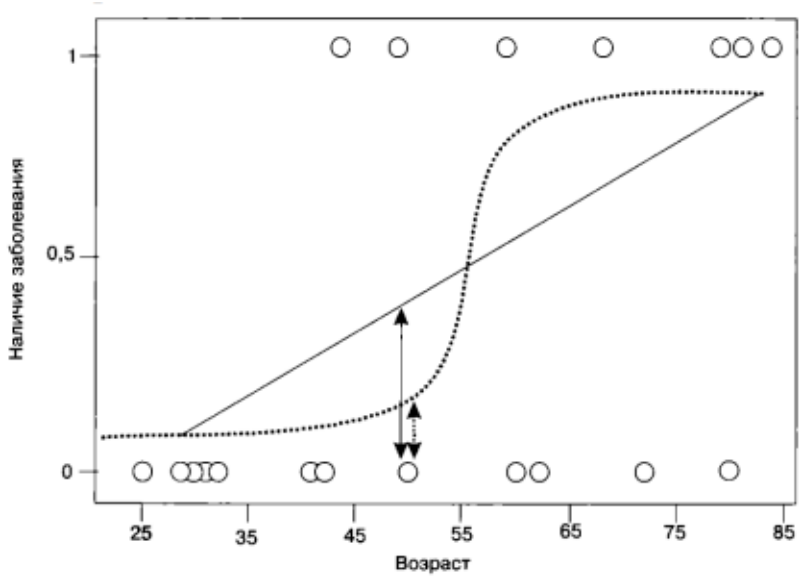


Рисунок 1.12. Диаграмма «Возраст – заболевание», линия регрессии и кривая логистической регрессии

Рассмотрим ошибки оценивания, полученные для пациента №11. Расстояние между точкой данных для пациента №11 и линией регрессии показано сплошной вертикальной стрелкой, а для кривой логистической регрессии пунктирной. Видно, что расстояние будет больше для линейной регрессии, а это означает, что она дает худшую оценку выходной переменной, чем логистическая. Это утверждение также является истинным для большинства других пациентов.

7.2 Основы логистической регрессии. Построение линии логистической регрессии.

Основы логистической регрессии.

Смотреть вопрос №7.1.

Построение линии логистической регрессии.

Введем в рассмотрение условное среднее $E(y|x)$ значений выходной переменной y для заданного значения x входной переменной X. $E(y|x)$ представляет собой ожидаемое значение выходной переменной при заданном значении входной. Напомним, что выходная переменная в линейной регрессии – это случайная переменная, определяемая как $y = \beta 0 + \beta 1 x 1 + \varepsilon$. Поскольку ошибка ε имеет нулевое среднее, для линейной регрессии мы получим, что $E(y|x) = \beta 0 + \beta 1 x 1$ (так же, как и в линейной регрессии, буквой b будем обозначать коэффициенты уравнения регрессии, а β – параметры соответствующей модели).

Для краткости введем обозначение $E(y|x) = \rho(x)$. Условное среднее для логистической регрессии имеет вид:

$$\rho(x) = \frac{e^{\beta_0+\beta_1 x}}{1+e^{\beta_0+\beta_1 x}}$$

Функцию, описываемую уравнением (1.8), называют логистической, а соответствующие кривые – сигмоидами, поскольку они имеют характерную S-образную форму. Эта функция определена на бесконечности и изменяется в диапазоне от 0 до 1. Диапазон изменения $\rho(x)$ также будет от 0 до 1, поэтому

данную функцию можно интерпретировать как вероятность того, что выходная переменная приобрела значение 1 (заболевание имеет место), а $1 - \rho(x)$ – как вероятность появления значения 0 (заболевание отсутствует).

Как уже говорилось при обсуждении модели линейной регрессии, ошибка ε является нормально распределенной случайной величиной с нулевым средним и постоянной дисперсией. Предположения, используемые для логистической регрессии, несколько отличаются. Выходная переменная является бинарной, и принятие выходной переменной одного из двух возможных значений называется исходом.

Исход – явление, показатель или признак, который служит объектом исследования. Например, при проведении клинических испытаний в медицине вероятность исхода служит критерием оценки эффективности лечебного или профилактического воздействия.

Если предположить, что принятие выходной переменной y значения 1 рассматривается как успех, а значения 0 – как неуспех, то $\rho(x)$ можно интерпретировать как вероятность успеха, а $1 - \rho(x)$ – неуспеха. Данная ситуация поясняется на рис. 1.13.

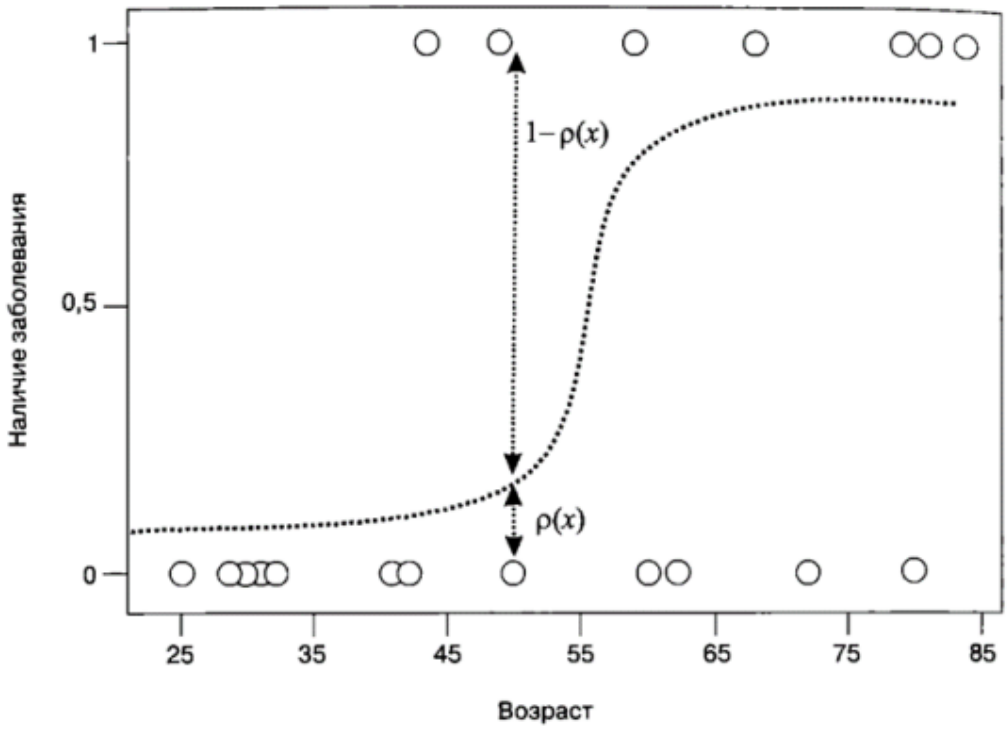


Рисунок 1.13. Геометрическая интерпретация вероятности исхода заболевания

В логистической регрессии используется преобразование вида:

$$g(x) = \ln\frac{\rho(x)}{1-\rho(x)} = \beta_0 + \beta_1 x$$

Оно называется логит-преобразованием и обладает такими полезными свойствами, как линейность, непрерывность и определенность на бесконечности.

7.3 Основы логистической регрессии. Оценки максимального правдоподобия.

Основы логистической регрессии.

Смотреть вопрос №7.1.

Оценки максимального правдоподобия.

Одним из наиболее привлекательных свойств линейной регрессии является то, что коэффициенты регрессии могут быть получены с помощью метода наименьших квадратов. Для оценки коэффициентов логистической регрессии таких решений не существует. Поэтому в ней коэффициенты оцениваются на основе метода максимального правдоподобия, который позволяет найти такие значения коэффициентов, для которых вероятность появления максимальна.

Введем в рассмотрение функцию правдоподобия $l(\beta|x)$. Она определяет вероятность появления значений параметров $\beta = \beta_1, \beta_2 \ \dots \ \beta_\mu$ для заданного значения x. Задача заключается в поиске таких значений этих параметров, которые максимизируют функцию правдоподобия: строятся оценки максимального правдоподобия, для которых значения параметров являются наиболее подходящими для наблюдаемых данных.

Вероятность того, что выходная переменная y приобретет значение 1 для заданного значения x (вероятность успеха), будет $\rho(x) = P(y = 1|x)$, а вероятность того, что $y = 0$ при заданном x, будет $1 - \rho(x) = P(y = 0|x)$. Таким образом, поскольку y_i или 1, вклад i-го наблюдения может быть выражен как $[\rho(x_i)]^{y_i} \times [1 - \rho(x_i)]^{(1-y_i)}$. Предположение, что наблюдения являются независимыми, позволяет представить функцию правдоподобия как произведение двух отдельных членов [1]:

$$l(\beta|x) = \prod_{i=1}^{n} [\rho(x_i)]^{y_i} \times [1 - \rho(x_i)]^{(1-y_i)}$$

В вычислительном плане более удобна логарифмическая функция правдоподобия $L(\beta|x) = \ln[l(\beta|x)]$

$$L(\beta|x) = \ln[l(\beta|x)] = \sum_{i=1}^{n} \{y_i \ln[\rho(x_i)] + (1 - y_i) \ln[1 - \rho(x_i)]\} \qquad (1.9)$$

Оценки максимального правдоподобия могут быть найдены путем дифференцирования $L(\beta|x)$ относительно каждого параметра и приравниванием результирующих выражений к 0.

Проверим результаты логистической регрессии для данных из табл. 1.9. Коэффициенты, то есть оценки максимального правдоподобия неизвестных параметров β_0 и β_1 определятся как $\beta_0 = -4{,}372$, а $\beta_1 = 0{,}067$. С учетом уравнения (1.8) можно записать:

$$\hat{\rho}(x) = \frac{e^{\hat{g}(x)}}{1 + e^{\hat{g}(x)}} = \frac{e^{-4{,}372+0{,}067(\text{возраст})}}{1 + e^{-4{,}372+0{,}067(\text{возраст})}}$$

где $\hat{g}(x) = -4{,}372 + 0{,}067x$ есть логит-преобразование.

Эти уравнения могут использоваться, чтобы оценить вероятность наличия заболевания у пациентов определённого возраста. Например, для пациента в возрасте 50 лет имеем:

$$\hat{g}(x) = -4{,}372 + 0{,}067 \times 50 = -1{,}022;$$

$$\hat{\rho}(x) = \frac{e^{\hat{g}(x)}}{1+e^{\hat{g}(x)}} = \frac{e^{-1{,}022}}{1+e^{-1{,}022}} = 0{,}26.$$

В итоге вероятность того, что пациент 50 лет страдает заболеванием, составляет 26%. Соответственно, вероятность отсутствия заболевания будет 100-26=74 %. Если провести такую же оценку для пациента в возрасте 72 лет, то можно увидеть, что вероятность наличия заболевания составит 61%, а его – отсутствия 39%.

7.4 Основы логистической регрессии. Значимость входных переменных.

Основы логистической регрессии.

Смотреть вопрос №7.1.

Значимость входных переменных.

Напомним, что в простой линейной регрессии модель считалась значимой, если средний квадрат значении оценок регрессии был больше, чем средний квадрат ошибки оценивания. Средний квадрат регрессии представляет собой меру улучшения оценки выходной переменной, если для оценивания мы используем не среднее значение, а входную переменную. Если входная переменная является «полезной» для оценивания значения выходной, то средний квадрат регрессии будет больше и статистика $F = Q_R/Q$ также будет больше. Тогда соответствующую линейную регрессионную модель можно будет рассматривать как значимую.

Значимость коэффициентов логистической регрессии определяется аналогично. В сущности, мы проверяем, обеспечивает ли использование в модели определенной входной переменной лучшую оценку выходной переменной.

Введем понятие насыщенной модели, то есть модели, в которой количество входных переменных равно числу наблюдений данных. Очевидно, что такая модель будет предсказывать значения выходной переменной с абсолютной точностью. Затем мы можем посмотреть на наблюдаемые значения выходной переменной, которые были предсказаны с помощью насыщенной модели. Для сравнения оценок, полученных с помощью обычной модели и насыщенной модели, введем понятие отклонения D:

$$D = -2 \ln \frac{l(\beta|x)}{l_{\text{нас}}(\beta|x)}$$

Здесь мы имеем опошление двух значений функции правдоподобия, поэтому проверка результирующей гипотезы называется проверкой отношения правдоподобия.

Отношение правдоподобия – отношение вероятности получить положительный результат для положительного исхода к вероятности получить положительный результат для отрицательного исхода.

Например, для выборки из табл. 1.9 отношение правдоподобия – это отношение вероятности обнаружить болезнь у больного к вероятности обнаружить болезнь у здорового.

Кроме этого, введем еще два понятия.

Отношение правдоподобия положительного результата – отношение вероятности получить истинноположительный результат к вероятности получить ложноположительный результат.

Отношение правдоподобия отрицательного результата теста – отношение вероятности получить истинноотрицательный результат к вероятности получить ложноотрицательный результат.

Обозначим оценку $\rho(x)$, полученную с помощью обычной модели, как $\hat{\rho}(x)$. Затем, используя уравнение (1.10), для случая логистической регрессии мы можем записать отклонение D в следующем виде [1]:

$$D = -2\sum_{i=1}^{n}\left[y_i \ln\frac{\hat{\rho}_i}{y_i} + (1-y_i)\ln\frac{1-\hat{\rho}_i}{1-y_i}\right]$$

Чтобы определить, является ли переменная значимой, нужно найти разность двух отклонений вычисленного для модели без данной входной переменной D^- и найденного для всей модели D^+, то есть

$$G = D^- - D^+ = -2\ln\left[\frac{\text{правдоподобие без переменной}}{\text{правдоподобие с переменной}}\right]$$

Введем обозначения $n_1 = \sum y_i$ и $n_0 = \sum(1-y_i)$. Тогда для случая единственной входной переменной можно записать [1]:

$$G = 2\left\{\sum_{i=1}^{n}[y_i \ln \hat{\rho}_i + (1 - y_i) \times \ln(1 - \hat{\rho}_i)] - [n_1 \ln n_1 + n_0 \ln n_0 - n \ln n]\right\}$$

Для примера из табл. 1.9 логарифмическое правдоподобие будет – 10,101, тогда:

$$G = 2\{-10{,}101 - [7 \ln(7) + 13 \ln(13) - 20 \ln(20)]\} = 5{,}696$$

При справедливости нулевой гипотезы, состоящей в предположении $\beta_1 = 0$, статистика G имеет распределение X^2 с одной степенью свободы. Следовательно, результирующее ρ-значение для проверки данной гипотезы будет $P(X_1^2 > 5{,}696) = 0{,}017$. Благодаря весьма малому ρ-значению становится очевидно, что возраст очень значимая переменная при определении вероятности наличия заболевания.

Другим методом для проверки значимости определенной входной переменной является тест Вальда. При нулевой гипотезе $\beta_1 = 0$ отношение $Z_W = \left(\frac{b_1}{E_{\text{ст}} b_1}\right)^2$ соответствует распределению хи-квадрат с одной степенью свободы, где $E_{\text{ст}} b_1$ – стандартная ошибка оценивания коэффициента регрессии на основе наблюдаемых данных.

Напомним, что стандартная ошибка оценивания коэффициентов регрессии $E_{\text{ст}}$ использовалась ранее для оценки значимости коэффициентов линейной регрессии. В логистической регрессии она также используется для этих целей. О том, как вычисляются стандартные ошибки оценивания коэффициентов логистической регрессии $E_{\text{ст}}$, будет рассказано дальше, а пока возьмем готовые цифры. Поскольку $b_1 = 0{,}067$, а $E_{b_1} = 0{,}0322$, то $Z_W = 0{,}067^2/0{,}0322^2 = 4{,}33$ и $P(z > 4{,}33) = 0{,}038$. Это неравенство означает, что вероятность справедливости нулевой гипотезы не превышает 3,8%. Данное ρ-значение также достаточно мало, хотя и не настолько, как полученное с помощью отношения правдоподобия. Следовательно, результаты обоих тестов совпадают, и переменная возраста является статистически значимой для оценки вероятности заболевания.

7.5 Основы логистической регрессии. Использование логистической регрессии для решения задач классификации.

Основы логистической регрессии.

Смотреть вопрос №7.1.

Использование логистической регрессии для решения задач классификации.

Постановка задач классификации и регрессии отличается характером выходной переменной. Если выходная переменная является непрерывной, то имеет место задача регрессии, а если дискретной (метка класса) – то классификации. Как было показано выше, логистическая регрессия позволяет работать с дихотомической выходной переменной, что предполагает возможность использования этого метода для решения задач бинарной классификации. В бинарной классификации каждое наблюдение или объект должны быть отнесены к одному из двух классов (например, А и Б). Тогда с каждым исходом связано событие: объект принадлежит к классу А и объект принадлежит к классу Б. Результатом будет оценка вероятности соответствующего исхода.

Если в процессе анализа будет установлено, что вероятность $P(A)$ принадлежности объекта с заданным набором значений признаков (входных переменных) к А больше, чем вероятность P(Б) ero принадлежности к классу Б, то он будет классифицирован как объект класса А. Очевидно, что поскольку события взаимоисключающие, то P(Б) = 1 − $P(A)$. Может быть задан порог вероятности, при превышении которого вероятность, связанная с определенным классом, «перевешивает» и объект относится к этому классу. В простейшем случае это может быть порог равной вероятности, то есть 0,5. Как только вероятность P(Б) становиться 0,51, а $P(A)$ – 0,49, объект относится к классу Б. Иногда порог определяется более сложным образом, например исходя из надежности решения. Так, решение о принадлежности объекта к определенному классу может быть принято только тогда, когда вероятность данного события, оцененная с помощью логистической регрессии, превысит 0,7.

Бинарная классификация на основе логистической регрессии широко применяется при решении задач в медицине, технической диагностике, социальной сфере и других предметных областях.

7.6 Интерпретация модели логистической регрессии. Шансы и отношение шансов.

Интерпретация модели логистической регрессии.

Очень важно не только математически описать модель, но и правильно интерпретировать ее с точки зрения анализа, то есть извлечь всю необходимую информацию об исследуемых объектах и процессах.

Напомним, что в простой линейной регрессии коэффициент b_1 интерпретируется как изменение значения выходной переменной при изменении входной на 1. В логистической регрессии его интерпретация аналогична, но применительно к логистической функции. То есть коэффициент b_1 может быть интерпретирован как изменение значения логистической функции при изменении входной переменной на 1. Формально это можно записать в виде [1]:

$$b_1 = g(x-1) - g(x)$$

Рассмотрим интерпретацию коэффициента $b1$ в простой логистической регрессии для трех случаев:

- когда входная переменная принимает только два значения (дихотомическая входная переменная);
- когда входная переменная может принимать несколько значений (полихотомическая входная переменная);
- для случая непрерывной входной переменной.

Шансы и отношение шансов.

Введем в рассмотрение понятие «шанс», который определяется как вероятность того, что событие произошло (шанс успеха), разделенная на вероятность того, что событие не произошло (шанс неуспеха). Шансы и вероятности содержат одну и ту же информацию, но по-разному выражают ее. Если вероятность того, что событие про изойдет, обозначить ρ, то шансы этого события будут равны $\rho/(1-\rho)$. Например, если вероятность выздоровления составляет 0,3, то шансы выздороветь равны $0{,}3/(1-0{,}3) = 0{,}43$. Если вероятность

вытащить любую карту пиковой масти из колоды составляет 0,25, то шансы этого события равны 0,25/(1 − 0,25) = 0,33.

Ранее мы обнаружили, что вероятность наличия заболевания у 72-летнего пациента составляет 61 %, соответственно, вероятность отсутствия заболевания 39 %. Таким образом, если обозначить шанс как O (Odds), то O = 0,61/0,39 = 1,56.

Мы также нашли, что вероятность наличия или отсутствия заболевания у 50-летнего пациента составляет 26 и 74 % соответственно. Тогда O = 0,26/0,74 = 0,35.

Заметим, что когда вероятность появления события выше, чем вероятность его отсутствия, то O > 1, а когда наоборот O < 1. Когда вероятности появления и отсутствия события равны, O = 1.

В бинарной логистической регрессии с дихотомической входной переменной вероятность того, что выходная переменная принимает значение y = 1 (событие произошло) при x = 1, может быть записано в виде [1]:

$$\frac{\rho(1)}{1-\rho(1)} = \frac{e^{\beta_0+\beta_1}/\left(1+e^{\beta_0+\beta_1}\right)}{1/(1+e^{\beta_0+\beta_1})} = e^{\beta_0+\beta_1}$$

Аналогично вероятность того, что выходная переменная принимает значение y = 1 (событие произошло) для наблюдений, в которых x = 0, будет [1]:

$$\frac{\rho(0)}{1-\rho(0)} = \frac{e^{\beta_0}/\left(1+e^{\beta_0}\right)}{1/(1+e^{\beta_0})} = e^{\beta_0}$$

Введем в рассмотрение так называемое отношение шансов, или отношение несогласия (odds ratio – OR), являющееся отношением шансов того, что событие произойдет, к шансам того, что событие не произойдет, В нашем случае это отношение шанса того, что выходная переменная примет значение 1 (событие произошло), к шансу того, что переменная примет значение 0 (событие не произошло),

То есть [1]:

$$OR = \frac{\rho(1)/(1-\rho(1))}{\rho(0)/(1-\rho(0))} = \frac{e^{\beta_0+\beta_1}}{e^{\beta_0}} = e^{\beta_1}$$

Данное отношение достаточно широко используется аналитиками, поскольку с его помощью хорошо выражается взаимосвязь между OR и коэффициентом b_1. Очевидно, что если отношение шансов равно 1, то есть шансы благоприятного и неблагоприятного исхода равны, то модель оказывается бесполезной, поскольку коэффициент $\beta_1 = 0$ и выход модели будут определяться только константой β_0. Таким образом, чем сильнее отношение шансов отличается от 1, тем более значимой будет модель. Если $OR < 1$, шансы благоприятного исхода меньше, чем шансы неблагоприятного (событие не произойдет), а если больше 1, то наоборот. Значения отношения шансов, близкие к 0, указывают на очень низкую вероятность благоприятного исхода.

Чтобы определить точность полученной оценки отношения шансов, используют стандартную ошибку отношения шансов, которая для случая дихотомической выходной переменной вычисляется с помощью выражения

$$E_{\text{ст}}(OR) = OR \times E_{\text{ст}}(b_1) = OR\sqrt{\frac{1}{n_{11}} + \frac{1}{n_{12}} + \frac{1}{n_{21}} + \frac{1}{n_{22}}} \qquad (1.11)$$

где $E_{\text{ст}}$ – стандартная ошибка оценивания соответствующего коэффициента регрессии, а значения $n_{11}, n_{12}, n_{21}, n_{22}$ – элементы четырехклеточной таблицы сопряженности признаков, отражающей все возможные состояния входной и выходной переменных,

Например, для задачи, в которой исследуется зависимость наличия некоторого заболевания от возраста пациента, таблица будет иметь следующий вид (табл. 8,17),

Таблица 1.10. Четырехклеточная таблица сопряженности признаков

Наличие заболевания, y	Возраст, x	
	$x < 50(0)$	$x \geq 50(1)$
Да (0)	$n_{11} = 21$	$n_{12} = 22$
Нет (1)	$n_{21} = 6$	$n_{22} = 51$

Тогда $E_{\text{ст}}(OR) = OR\sqrt{\frac{1}{21} + \frac{1}{22} + \frac{1}{6} + \frac{1}{51}} = 0{,}529 \times OR$

Границы доверительного интервала отношения шансов будут вычисляться по формулам [1]:

$$exp\left(b_0 \pm z \times E_{\text{ст}}(b_0)\right)$$
$$exp\left(b_1 \pm z \times E_{\text{ст}}(b_1)\right)$$

где z – критическое значение коэффициента Стьюдента, связанное с уровнем достоверности $(1-\alpha) \times 100\%$ (для $\alpha = 0{,}05 z \approx 1{,}96$);

$E_{\text{ст}}(b_0)$, $E_{\text{ст}}(b_1)$ – стандартные ошибки оценивания коэффициента регрессии на основе наблюдаемых данных. Если данный интервал не содержит $e^0 = 1$, то отношение шансов с достоверностью 95% является статистически значимым.

Ошибка $E_{\text{ст}}(b_1)$ для непрерывной переменной определяется как квадратный корень дисперсии оценки и вычисляется в процесс е оценки максимального правдоподобия. Ручные вычисления трудоемки и громоздки, поэтому значения ошибки берут из соответствующей компьютерной программы.

Часто отношение шансов используется для определения понятия относительно риска:

$$\text{относительный риск} = \frac{\rho(1)}{\rho(0)}$$

8 АНАЛИЗ И ПРОГНОЗИРОВАНИЕ ВРЕМЕННЫХ РЯДОВ

8.1 Прогнозирование. Временной ряд и его компоненты. Непрерывный и дискретный временные ряды. Цели и задачи анализа временных рядов.

Прогнозирование.

Прогнозирование одна из самых востребованных задач. И это неудивительно: зная, пусть даже с определенной погрешностью, характер развития событий в будущем, можно принимать более обоснованные управленческие решения, планировать деятельность, разрабатывать соответствующие комплексы мероприятий, эффективно распределять ресурсы и т.д. С точки зрения технологий анализа данных прогнозировании может рассматриваться как определение некоторой неизвестной величины по набору связанных с ней значений. Поэтому прогнозирование выполняется с помощью таких задач Data Mining, как регрессия, классификация и кластеризация.

Продажи, поставки, заказы — это процессы, распределенные во времени. Данные, собираемые и используемые для разработки прогнозов, чаще всего представляют собой временные ряды, то есть описывают развитие того или иного бизнес процесса во времени. Следовательно, прогнозирование в области продаж, сбыта и спроса, управления материальными запасами и потоками обычно связано именно с анализом временных рядов. Это объясняет, почему обработке временных рядов в бизнес аналитике и Data Mining уделяется такое большое внимание.

Все методы прогнозирования можно разделить на три большие группы формализованные, эвристические и комплексные.

Формализованные методы позволяют получать в качестве прогнозов количественные показатели, описывающие состояние некоторого объекта или

процесса. При этом предполагается, что анализируемый объект или процесс обладает свойством инерционности, то есть в будущем он продолжит развиваться в соответствии с теми же законами, по которым развивался в прошлом и существует в настоящем.

Недостатком формализованных методов является то, что для прогноза они могут использовать только исторические данные, находящиеся в пределах эволюционного цикла развития объекта или процесса. Поэтому такие методы пригодны лишь для оперативных и краткосрочных прогнозов. К формализованным методам относятся экстраполяционные и регрессионные методы, методы математической статистики, факторный анализ и др.

Эвристические методы основаны на использовании экспертных оценок. Эксперт (группа экспертов), опираясь на свои знания в предметной области и практический опыт, способен предсказать качественные изменения в поведении исследуемого объекта или процесса. Эти методы особенно полезны в тех случаях, когда поведение объектов и процессов, для которых требуется дать прогноз, характеризуется большой степенью неравномерности. Если формализованные методы в силу присущих им ограничений используются для оперативных и краткосрочных прогнозов, то эвристические методы чаще применяются для среднесрочных и перспективных.

Комплексное прогнозирование использует комбинацию формализованного подхода с экспертными оценками, что в некоторых случаях позволяет добиться наилучшего результата.

Независимо от методики на эффективность прогноза в наибольшей степени влияет то, насколько он полезен для планирования и ведения бизнеса. Прогноз полезен только тогда, когда его компоненты тщательно продуманы и ограничения, содержащиеся в нем, представлены явно.

Прогнозирование очень широкое понятие, как отмечалось ранее, в большинстве случаев оно связывается с предвидением во времени, с предсказанием дальнейшего развития событий.

Временной ряд и его компоненты.

Временной ряд представляет собой последовательность наблюдений за изменениями во времени значений параметров некоторого объекта или процесса, строго говоря, каждый процесс непрерывен во времени, то есть некоторые значения параметров этого процесса существуют в любой момент времени. Например, если вы запросите в банке текущий курс валют, вам никогда не ответят, что в настоящее время курса валют нет. Возможно, новый курс установился минуту назад, возможно, через час он изменится, но в настоящее время какое-то его значение непременно существует.

Для задач анализа не нужно знать значения параметров объектов в любой момент времени. Интерес представляют временные отсчеты значения, зафиксированные в некоторые, обычно равноотстоящие моменты времени. Отсчеты могут браться через различные промежутки: через минуту, час, день, неделю, месяц или год в зависимости от того, насколько детально должен быть проанализирован процесс.

Непрерывный и дискретный временные ряды.

В задачах анализа временных рядов мы имеем дело с дискретным временем, когда каждое наблюдение за параметром образует временной отсчет. Непрерывный и дискретный временные ряды иллюстрируются на рис. 1.

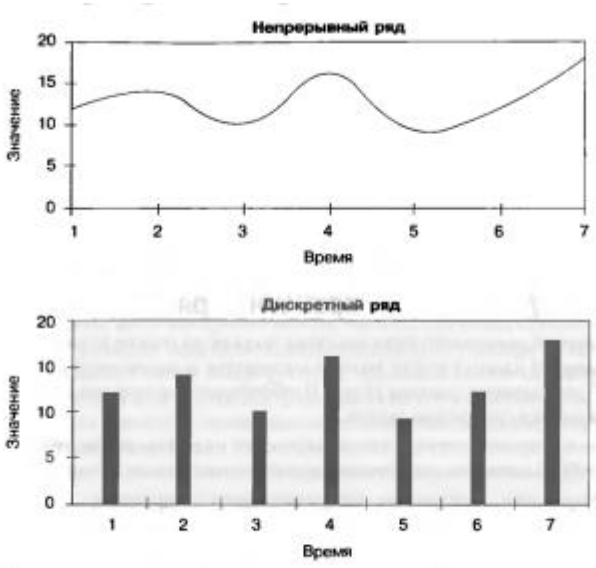


Рис. 1. Непрерывный и дискретный временные ряды

Непрерывный ряд – это процесс, значения которого известны в любой момент времени, а дискретный ряд процесс, значения которого известны только

в заданные моменты времени (временные отсчеты). Непрерывные данные – это данные, которые могут принимать бесконечное множество значений. Непрерывными могут быть только числовые данные. Дискретные данные могут принимать ограниченный набор заранее определенных значений (категорий).

Временные ряды бывают одномерные и многомерные. Одномерные ряды содержат наблюдения за изменением только одного параметра исследуемого процесса или объекта, а многомерные за двумя параметрами или более.

Цели и задачи анализа временных рядов.

При изучении временного ряда аналитик должен на основе некоторого отрезка ряда конечной длины сделать выводы о характере и закономерностях процесса, который описывается данным рядом. Наиболее часто в ходе анализа временных рядов решаются следующие задачи:

- описание характеристик и закономерностей ряда. На основе этого описания могут быть выявлены свойства соответствующих бизнес-процессов;
- моделирование построение модели исследуемого процесса;
- прогнозирование предсказание будущих значений временного ряда;
- управление. Зная свойства временных рядов, можно выработать методы воздействия на соответствующие бизнес-процессы для управления ими.

Наиболее востребованными в бизнес аналитике являются задачи моделирования и прогнозирования. Они взаимосвязаны: чтобы составить прогноз, сначала нужно построить соответствующую модель, проверить степень ее адекватности и правильно применить к имеющимся наблюдениям.

При решении данных задач можно столкнуться с такими проблемами, как:

- недостаточное количество наблюдений, содержащихся во временном ряду;
- статистическая изменчивость ряда, которая приводит к тому, что прошлые значения ряда устаревают и обесцениваются с точки зрения

выявления текущих закономерностей и возможности прогнозирования.

8.2 Прогнозирование. Детерминированная и случайная составляющая временного ряда. Модели временных рядов.

Прогнозирование.

Смотреть вопрос №8.1.

Детерминированная и случайная составляющая временного ряда.

Процесс разрабатывается и направляется осмысленно в соответствии с теми или иными целями, поэтому в его поведении должны присутствовать определенные закономерности. Некоторые процессы протекают равномерно, не отклоняясь от намеченных показателей. Например, завод должен производить ровно столько изделий, сколько требуется заказчикам, поэтому, если спрос стабильный, то и производство необходимо поддерживать на одном уровне. Увеличивать или снижать его нет смысла, поскольку в первом случае будет иметь место работа «на склад», а во втором упущенная прибыль и потерянные заказчики.

Если спрос устойчиво возрастает с течением времени, то и производство должно наращиваться пропорционально. Если спрос характеризуется периодичностью, вызванной сезонными колебаниями, то и производство должно подстраиваться под них. Возьмем мороженое и шипованные автопокрышки: первое лучше продается в теплое время, а вторые в зимнее. Попытка пойти против данной закономерности вряд ли будет способствовать коммерческому успеху.

Но любой бизнес-процесс сталкивается с воздействием различных случайных факторов. Диапазон таких факторов очень широк от стихийных бедствий и пожаров до банального хищения на складе готовой продукции. При этом случайные факторы могут как препятствовать, так и способствовать бизнесу. Например, аномально холодная зима будет способствовать взлету

продаж отопительных приборов, а аномально теплая испортит бизнес продавцам зимних автопокрышек.

Общим свойством всех случайных факторов является то, что они не поддаются прогнозированию и вносят во временные ряды изменения, не соответствующие их основным закономерностям.

Можно выделить две составляющие временного ряда закономерную (детерминированную) и случайную (стохастическую).

Закономерная (детерминированная) составляющая временного ряда последовательность значений, элементы которой могут быть вычислены в соответствии с определенной функцией. Закономерная составляющая временного ряда отражает действие известных факторов и величин.

Зная функцию, описывающую закономерность, в соответствии с которой развивается исследуемый процесс, мы можем вычислить значение детерминированной составляющей в любой момент времени.

Случайная (стохастическая) составляющая временного ряда последовательность значений, которая является результатом воздействия на исследуемый процесс случайных факторов. Случайная составляющая и ее влияние на временной ряд могут быть оценены только с помощью статистических методов.

Пример соотношения детерминированной и случайной составляющих представлен на рис. 2. Здесь детерминированная составляющая изображена плавной темной линией, а случайная быстро изменяющейся светлой линией.

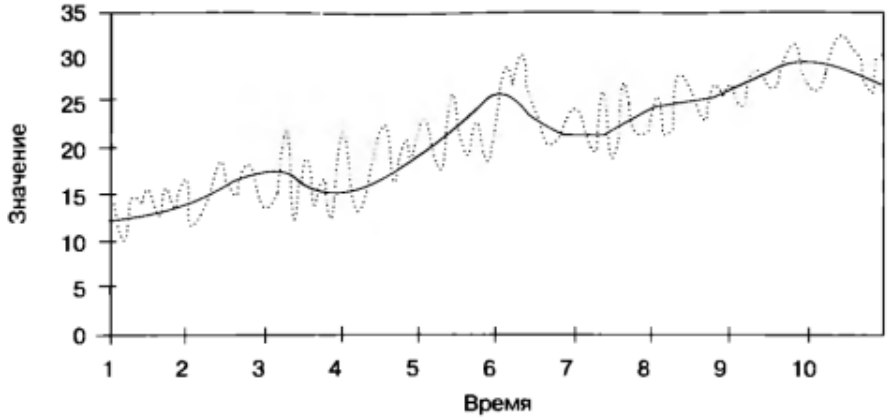


Рис. 2. Детерминированная и случайная составляющие

Случайная составляющая не существует отдельно от детерминированной. Она проявляется только как результат воздействия набора случайных факторов на исследуемый процесс и обычно выражается в повышенной изменчивости временного ряда, а также в отклонении значений детерминированной составляющей. Проще говоря, результирующее значение временного ряда – это результат взаимодействия детерминированной и случайной составляющих. Простейшим видом такого взаимодействия является случай, когда каждое значение временного ряда можно рассматривать как сумму (разность) двух значений, одно из которых обусловлено детерминированной составляющей, а другое случайной.

Модели временных рядов.

Наблюдаемые значения временного ряда представляют собой результат взаимодействия детерминированной и случайной составляющих. Различают два вида такого взаимодействия:

- аддитивное значения временного ряда получаются как результат сложения детерминированной и случайной составляющих;
- мультипликативное значения временного ряда получаются как результат умножения детерминированной и случайной составляющих.

Соответственно модели временных рядов также бывают аддитивные и мультипликативные.

Аддитивная модель имеет вид:

$$x_i = d_i + p_i \text{ ИЛИ } X = D + \text{P}$$

где $i = 1, \dots, n$ – номер временного отсчета.

Мультипликативная модель имеет вид:

$$x_i = d_i \times p_i \text{ ИЛИ } X = D \times \text{P}$$

Очень важно знать характер взаимодействия детерминированной и случайной составляющих, поскольку методика анализа временных рядов зависит от используемой модели.

8.3 Прогнозирование. Компоненты временного ряда. Тренд. Сезонная и циклические компоненты.

Прогнозирование.

Смотреть вопрос №8.1.

Компоненты временного ряда.

Количество разнообразных процессов в экономике, управлении, бизнесе, социальной и государственной сфере весьма велико, и поведение временных рядов, описывающих эти процессы, может существенно различаться. Поэтому для описания поведения временных рядов были введены три компоненты, своего рода типовые структуры, которые можно выделить во временном ряду – тренд, сезонная компонента и циклическая компонента.

С учетом указанных компонент детерминированная составляющая ряда может быть записана в виде:

$$d_i = t_i + s_i + c_i$$

где t_i – тренд; s_i – сезонная компонента; c_i – циклическая компонента; $i = 1, \dots, n$ – номер временного отсчета.

Тренд.

Тренд – наиболее важная компонента временного ряда. Именно с выделения тренда чаще всего и начинается анализ временного ряда.

Тренд – медленно меняющаяся компонента временного ряда, которая описывает влияние на временной ряд долговременно действующих факторов, вызывающих плавные и длительные изменения ряда.

Действительно, среди всех факторов, влияющих на экономические и бизнес-процессы, выделяются быстродействующие и медленнодействующие. Быстродействующие факторы, такие как стихийное бедствие, «обвал» фондового рынка и т. д, могут изменить ситуацию в течение нескольких дней или даже часов. Медленнодействующие факторы могут изменять ситуацию в течение нескольких месяцев и даже лет.

Чтобы представить характер тренда, обычно бывает достаточно взглянуть на график временного ряда. Для описания тренда используются различные модели, наиболее популярными из которых являются следующие:

- простая линейная модель: $t_i = a + b \times i$, где $i = 1, \ldots, n$ – номер временного отсчета (элемента ряда). Пример линейного тренда представлен на рис. 3.

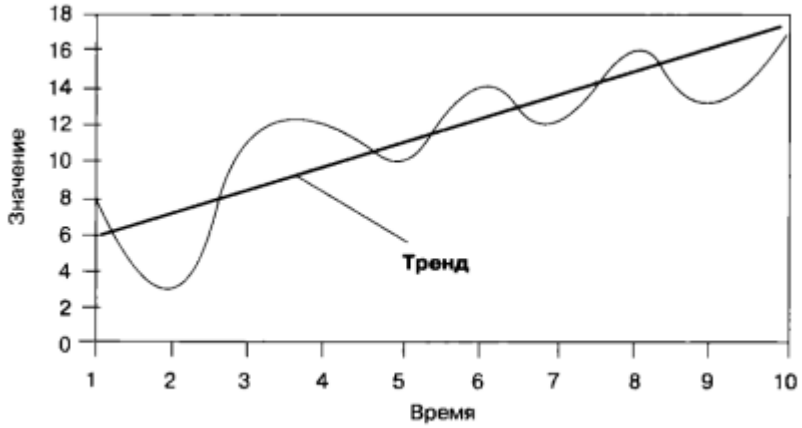


Рис. 3. Линейный тренд

Несмотря на свою простоту, линейная модель тренда часто оказывается полезной при решении реальных задач анализа, поскольку многие бизнеспроцессы линейны по своей природе.

- Полиномиальная модель: $t_i = a + b_1 \times i + b_2 \times i^2 + \cdots + b_n \times i^n$. В большинстве реальных задач степень полинома не превышает 5.
- Экспоненциальная модель: $t_i = exp(a + b \times i)$. Используется в случаях, когда процесс характеризуется равномерным увеличением темпов роста. Пример экспоненциального тренда представлен на рис. 4 (сплошная линия).

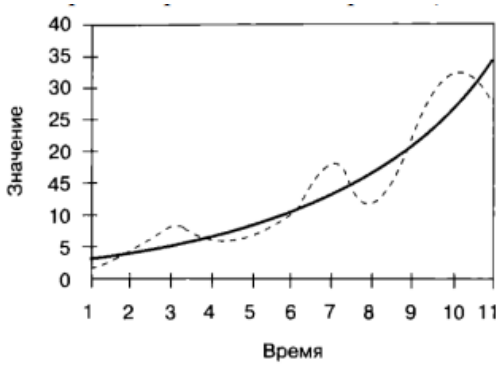


Рис. 4. Экспоненциальный тренд

- Логистическая модель: $t_i = a/(1 + b \times e^{-ki})$, где k – константа, управляющая крутизной логистической функции (рис. 5).

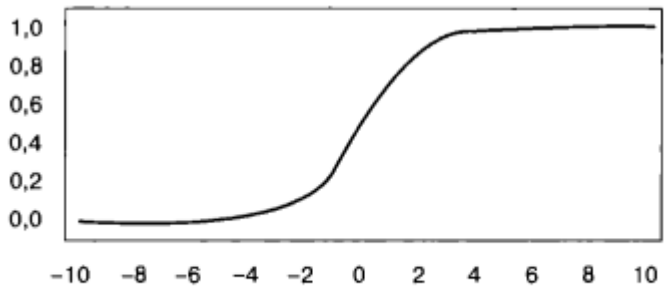


Рис. 5. Логистическая функция (сигмоида)

Такого типа кривые, имеющие s-образную форму, часто называют сигмоидами. Они хорошо описывают процессы с непостоянными темпами роста.

Сезонная и циклические компоненты.

Многим процессам свойственна повторяемость во времени, причем периодичность таких повторений может изменяться в очень широком диапазоне. Например, для экономики стран и регионов характерны подъемы и спады, которые могут длиться десятилетиями. В супермаркете продажи ежедневно увеличиваются в вечернее время, когда люди идут с работы. Подобных примеров можно привести множество.

Очевидно, что для описания таких периодических изменений, присутствующих во временных рядах, тренд непригоден. Поэтому вводится еще одна компонента, называемая сезонностью, или сезонной компонентой.

Сезонная компонента – составляющая временного ряда, описывающая регулярные изменения его значений в пределах некоторого периода и представляющая собой последовательность почти повторяющихся циклов.

Сезонная компонента может быть привязана к определенному календарному временному интервалу: дню, неделе, кварталу, месяцу или году – либо к какому-нибудь событию, которое прямо не соотносится с конкретными календарными интервалами.

Сезонную компоненту с изменяющимся периодом иногда называют «плавающей».

Утверждать, что сезонная компонента описывается периодической функцией, было бы неверно. Понятно, что регулярно повторяющиеся всплески продаж, образующие сезонную компоненту, не будут воспроизводиться абсолютно точно (как, например, значения функций синуса или косинуса). Тем не менее сезонная компонента обычно хорошо прослеживается на графике временного ряда.

Часто временные ряды содержат изменения, слишком плавные и заметные для случайной составляющей. В то же время такие изменения нельзя отнести ни к тренду, поскольку они не являются достаточно протяженными, ни к сезонной компоненте, поскольку они не являются регулярными. Подобные изменения называются циклической компонентой временного ряда. Она занимает промежуточное положение между детерминированной и случайной составляющими временного ряда.

Циклическая компонента временного ряда – интервалы подъема или спада, которые имеют различную протяженность, а также различную амплитуду расположенных в них значений.

Наличие в рядах данных циклических компонент связано с тем, что в пределах интервалов более глобальных изменений (например, сезонных) могут наблюдаться не имеющие периодичности временные подъемы и спады, которые, в отличие от случайной компоненты, не вызваны действием случайных факторов, а являются особенностями бизнеса и обусловлены общеэкономической ситуацией. На графике 6 циклическая компонента выглядит как плавные волнообразные флуктуации вокруг тренда.

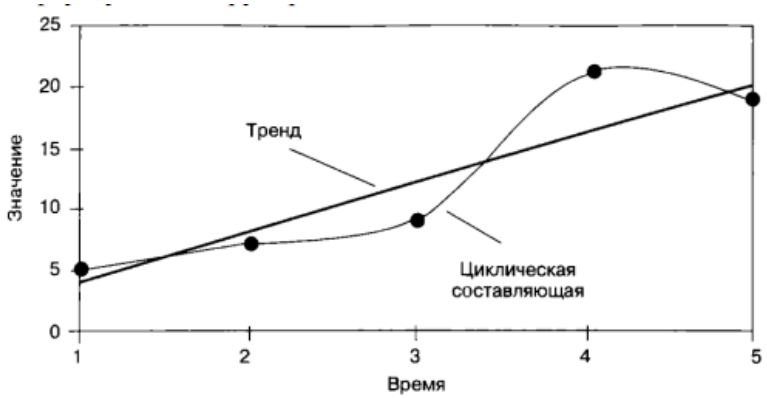


Рис. 6. Циклическая компонента

Выделить во временных рядах циклическую компоненту формальными методами довольно сложно. Обычно для этого используют информацию из других временных рядов, которые связаны с исследуемым. Например, объяснить колебания ряда в пределах сезонных интервалов можно инфляцией, изменениями на рынке трудовых ресурсов и т. д. Но для этого необходимо привлечь соответствующую информацию.

Изучение циклической компоненты часто оказывается полезным для прогнозирования, особенно краткосрочного. Таким образом, временной ряд можно представить, как композицию, состоящую из двух составляющих случайной и детерминированной. Детерминированная составляющая, в свою очередь, содержит три компоненты тренд, сезонную и циклическую.

8.4 Прогнозирование. Исследование временных рядов и автокорреляция. Коэффициент автокорреляции. Примеры.

Прогнозирование.

Смотреть вопрос №8.1.

Исследование временных рядов и автокорреляция.

Цель анализа временного ряда построение его математической модели, с помощью которой можно обнаружить закономерности поведения ряда, а также построить прогноз его дальнейшего развития. Главной проблемой при построении таких моделей является нестационарность ряда.

Временной ряд называется стационарным, если его статистические свойства (математическое ожидание и дисперсия) одинаковы на всем протяжении ряда.

Если статистические свойства для различных интервалов ряда существенно различаются, то такой ряд называется нестационарным. Применение к нестационарным рядам различных методов анализа, в том числе статистических, затруднено. Поэтому, прежде чем приступать к построению модели ряда, его стараются свести к стационарному.

При исследовании временного ряда, как правило, ищут ответы на несколько вопросов.

- Является ли ряд данных случайным?
- Содержит ли временной ряд тренд и сезонную компоненту?
- Является ли временной ряд стационарным?

Для ответа используется аппарат корреляционного анализа. Корреляция – это понятие математической статистики, которое характеризует степень статистической взаимосвязи между элементами данных. Если взаимосвязь между элементами данных присутствует, то такие данные называются коррелированными, в противном случае некоррелированный.

Если определяется корреляция между двумя временными рядами, то говорят о взаимной корреляции. Когда устанавливается степень статистической зависимости между значениями одного временного ряда, имеет место автокорреляция. В этом случае вычисляется корреляция между временным рядом и его копией, сдвинутой на один или несколько временных отсчетов.

Смысл корреляционного анализа заключается в следующем. Детерминированная составляющая временного ряда, которая описывает закономерности, присущие связанному с ним процессу, характеризуется плавными изменениями значений ряда. То есть соседние значения ряда не должны сильно отличаться, и, следовательно, между ними присутствует взаимная зависимость (элементы ряда коррелированы между собой). Если значения ряда в большей степени обусловлены случайной составляющей и соседние значения могут существенно отличаться друг от друга, то степень их взаимозависимости и, следовательно, корреляция будут меньше.

Рассмотрим понятие автокорреляции на следующем примере. Пусть дан ряд, который содержит последовательность ежемесячных наблюдений за продажами (табл.1).

Таблица 1. Продажи по месяцам

Месяц	**Продажи**
Январь	125

Февраль	130
Март	140
Апрель	132
Май	145
Июнь	150
Июль	148
Август	155
Сентябрь	157
Октябрь	160
Ноябрь	158
Декабрь	165

Для того чтобы вычислить автокорреляцию ряда, будем использовать его копию, сдвинутую в сторону запаздывания на определенное количество отсчетов (табл. 2).

Таблица 2. Данные для расчета автокорреляционной функции (АКФ)

X	125	130	140	132	145	150	148	155	157	160	158	165
X_{i-1}		125	130	140	132	145	150	148	155	157	160	158
X_{i-2}			125	130	140	132	145	150	148	155	157	160
…	…	…	…	…	…	…	…	…	…	…	…	…
X_{i-n}												125

Тогда ряд X_{i-1} будет представлять собой копию исходного ряда, сдвинутую на один временной отсчет, X_{i-2} – на два временных отсчета и т. д. Для определения степени взаимной зависимости элементов ряда используется величина, называемая коэффициентом автокорреляции r_k, где k – количество отсчетов, на которое был сдвинут временной ряд при вычислении данного коэффициента.

Коэффициент автокорреляции.

Лаг (сдвиг во времени) определяет порядок коэффициента автокорреляции. Если L = 1, то имеем коэффициент автокорреляции 1-го порядка $r_{k,k-1}$. Если L = 2, то коэффициент автокорреляции 2-го порядка $r_{k,k-2}$ и т.д.

Коэффициент автокорреляции вычисляется в соответствии с формулой:

$$r_k = \frac{\sum_{i=k+1}^{n}(x_i - \bar{x})(x_{i-k} - \bar{x})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 (x_{i-k} - \bar{x})^2}}$$

где x_i – значение l-го отсчета; x_{i-k} – наблюдение x_i со сдвигом на k временных отсчетов; $\bar{x}$ – среднее значение ряда, которое рассчитывается по формуле

$$\bar{x}_{k-l} = \frac{\sum_{t=k}^{n} x_i}{n-l}$$

Коэффициент корреляции изменяется в диапазоне [-1; 1], где $r_k = 1$ указывает на полную корреляцию.

Примеры.

Из головы.

8.5 Модели прогнозирования. Обобщенная модель прогноза. Метод скользящего окна.

Модели прогнозирования.

На современном этапе главным инструментом прогнозирования являются прогностические модели. От того, насколько модель прогноза адекватна условиям, в которых работает компания, насколько полно в ней учитываются внешние и внутренние факторы, воздействующие на те или иные бизнес-процессы, зависит точность и достоверность прогноза.

Обобщенная модель прогноза.

Структура прогностической модели похожа на структуры моделей, используемых для решения других задач анализа, например распознавания, идентификации и т. д. Модель прогноза отличается только характером используемых данных и алгоритмами их обработки. Обобщенная структура прогностической модели представлена на рис. 1.

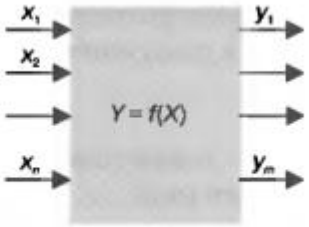


Рис. 1. Обобщенная модель прогноза

Здесь набор входных переменных ($i = 1. . . n$), образующих вектор X, исходные данные для прогноза. Набор выходных переменных y_j ($j = 1. . . m$), образующих вектор результата Y, есть набор прогнозируемых величин. Когда решается задача прогнозирования значений временного ряда, описывающего динамику изменения некоторого бизнес-процесса, входные значения – наблюдения за развитием процесса в прошлом, а выходные прогнозируемые значения процесса в будущем. При этом временные интервалы прошлых

наблюдений и временные интервалы, по которым требуется получить прогноз, должны соответствовать друг другу. Например, требуется получить прогноз по продажам на будущую неделю. Наблюдения, на основе которых будет строиться прогноз, также должны быть взяты за неделю. Если в базе данных история продаж представлена по дням, то получить данные по неделям можно с помощью соответствующего агрегирования. Обучающая выборка строится путем преобразования временного ряда с помощью скользящего окна.

Метод скользящего окна.

Метод скользящего окна - алгоритм трансформации, позволяющий сформировать из членов временного ряда набор данных, который может служить обучающим множеством для построения модели прогнозирования. Под окном в данном случае понимается временной интервал, содержащий набор значений, которые используются для формирования обучающего примера. В процессе работы алгоритма окно смещается по временной последовательности на единицу наблюдения, и каждое положение окна образует один пример.

Кроме того, количество наблюдений за историей развития процесса в прошлом, на основе которого строится прогноз, должно быть больше, чем число прогнозируемых интервалов, то есть $n > m$. Иначе говоря, если мы хотим получить прогноз на неделю, то для этого должны взять наблюдения за несколько прошедших недель.

8.6 Методы прогнозирования: формализованные, эвристические и комплексные.

Методы прогнозирования: формализованные, эвристические и комплексные.

Все методы прогнозирования можно разделить на три большие группы формализованные, эвристические и комплексные.

Формализованные методы позволяют получать в качестве прогнозов количественные показатели, описывающие состояние некоторого объекта или

процесса. При этом предполагается, что анализируемый объект или процесс обладает свойством инерционности, то есть в будущем он продолжит развиваться в соответствии с теми же законами, по которым развивался в прошлом и существует в настоящем.

Недостатком формализованных методов является то, что для прогноза они могут использовать только исторические данные, находящиеся в пределах эволюционного цикла развития объекта или процесса. Поэтому такие методы пригодны лишь для оперативных и краткосрочных прогнозов. К формализованным методам относятся экстраполяционные и регрессионные методы, методы математической статистики, факторный анализ и др.

Эвристические методы основаны на использовании экспертных оценок. Эксперт (группа экспертов), опираясь на свои знания в предметной области и практический опыт, способен предсказать качественные изменения в поведении исследуемого объекта или процесса. Эти методы особенно полезны в тех случаях, когда поведение объектов и процессов, для которых требуется дать прогноз, характеризуется большой степенью неравномерности. Если формализованные методы в силу присущих им ограничений используются для оперативных и краткосрочных прогнозов, то эвристические методы чаще применяются для среднесрочных и перспективных.

Комплексное прогнозирование использует комбинацию формализованного подхода с экспертными оценками, что в некоторых случаях позволяет добиться наилучшего результата.

Независимо от методики на эффективность прогноза в наибольшей степени влияет то, насколько он полезен для планирования и ведения бизнеса. Прогноз полезен только тогда, когда его компоненты тщательно продуманы и ограничения, содержащиеся в нем, представлены явно.

Прогнозирование очень широкое понятие, как отмечалось ранее, в большинстве случаев оно связывается с предвидением во времени, с предсказанием дальнейшего развития событий.

8.7 Модели прогнозирования. «Наивная» модель прогнозирования. Экстраполяция.

Модели прогнозирования.

Смотреть вопрос №8.5.

«Наивная» модель прогнозирования.

«Наивная» модель предполагает, что последний период прогнозируемого временного ряда лучше всего описывает будущее этого ряда. В таких моделях прогноз, как правило, является довольно простой функцией от наблюдений прогнозируемой величины в недалеком прошлом.

Простейшая модель описывается выражением:

$$y(t+1) = y(t)$$

где $y(t)$ – последнее наблюдаемое значение;

$y(t+1)$ – прогноз.

В основу этой модели заложен принцип: «Завтра будет как сегодня».

Ждать от такой примитивной модели точного прогноза не стоит. Она не только не учитывает закономерности прогнозируемого процесса (что в той или иной степени свойственно многим статистическим методам прогнозирования), но и не защищена от случайных изменений в данных, а также не отражает сезонного колебания и тренды.

Чтобы модель учитывала наличие возможных трендов, ее можно несколько усложнить, например преобразовав к виду $y(t+1) = y(t) + [y(t) - y(t-1)]$ или $y(t+1) = y(t) \times [y(t)/y(t-1)]$. При необходимости учета сезонных колебаний «наивная», модель модифицируется следующим образом: $y(t+1) = y(t-s)$, где s - показатель, учитывающий сезонные изменения прогнозируемого временного ряда.

Экстраполяция.

Экстраполяция представляет собой попытку распространить закономерность поведения некоторой функции из интервала, в котором известны ее значения, за его пределы. Иными словами, если значения функции $f(x)$ известны в некотором интервале $[x_0, x_n]$, то целью экстраполяции является определение наиболее вероятного значения в точке x_{n-1}.

Экстраполяция применима только в тех случаях, когда функция $f(x)$ (а соответственно, и описываемый с ее помощью временной ряд) достаточно стабильна и не подвержена резким изменениям. Если это требование не выполняется, скорее всего, поведение функции в различных интервалах будет подчиняться разным закономерностям.

К существенным факторам, определяющим эффективность применения метода экстраполяции, относится надежность данных, лежащих в основе анализа. Экстраполяция тенденций получила широкое применение в нормативном прогнозировании. В частности, с помощью этого метода устанавливается, можно ли, используя существующие технологии, достичь заданных производственных или других бизнес-показателей.

Наиболее популярным методом экстраполяции в настоящее время является экспоненциальное сглаживание. Основной его принцип заключается в том, чтобы учесть в прогнозе все наблюдения, но с экспоненциально убывающими весами.

Метод позволяет принять во внимание сезонные колебания ряда и предсказать поведение трендовой составляющей.

8.8 Модели прогнозирования. Прогнозирование методом среднего и скользящего среднего.

Модели прогнозирования.

Смотреть вопрос №8.5.

Прогнозирование методом среднего и скользящего среднего.

Наиболее простая модель этой группы обычное усреднение набора наблюдений прогнозируемого ряда:

$$y(t+1) = \frac{(y(t) + y(t-1) + y(t-2) + \cdots + y(1))}{t}$$

Принцип модели простого среднего: «Завтра будет как в среднем за последнее время». Преимущество такого подхода по сравнению с «наивной» моделью очевидно: при усреднении сглаживаются резкие изменения и выбросы данных, что делает результаты прогноза более устойчивыми к изменчивости ряда. Но в целом эта модель прогноза столь же примитивна, как и «наивная», и ей присущи те же недостатки. В формуле прогноза на основе среднего предполагается, что ряд усредняется по достаточно длительному интервалу времени (в пределе по всем наблюдениям).

С точки зрения прогноза это не вполне корректно, так как старые значения временного ряда могли формироваться на основе иных закономерностей и утратить актуальность. Поэтому свежие наблюдения из недалекого прошлого лучше описывают прогноз, чем более старые значения того же ряда. Чтобы повысить точность прогноза, можно использовать скользящее среднее:

$$y(t+1) = \frac{(y(t) + y(t-1) + y(t-2) + \cdots + y(t-T))}{T+1}$$

Смысл данного метода заключается в том, что модель «видит» только ближайшее прошлое на T отсчетов по времени и прогноз строится только на этих наблюдениях.

Метод скользящего среднего весьма прост, и его результаты довольно точно отражают изменения основных показателей предыдущего периода. Иногда он оказывается даже эффективнее, чем методы, основанные на долговременных наблюдениях.

И так, чем меньшее количество наблюдений используется для вычисления скользящего среднего, тем точнее будут отражены изменения показателей, на основе которых строится прогноз. Однако, если для прогнозируемого скользящего среднего используется только одно или два наблюдения, такой прогноз может быть слишком упрощенным. Чтобы определить, сколько наблюдений желательно включить в скользящее среднее, нужно исходить из

предыдущего опыта и имеющейся информации о наборе данных. Необходимо соблюдать равновесие между повышенным откликом скользящего среднего на несколько самых поздних наблюдений и большой изменчивостью скользящего среднего. Одно отклонение в наборе данных для трёхкомпонентного среднего может исказить весь прогноз.

8.9 Модели прогнозирования. Регрессионные модели. Метод декомпозиции временного ряда.

Модели прогнозирования.

Смотреть вопрос №8.5.

Регрессионные модели.

К числу наиболее мощных, развитых и универсальных моделей прогнозирования относятся регрессионные модели. Регрессия – это технология статистического анализа, целью которой является определение лучшей модели, устанавливающей взаимосвязь между выходной (зависимой) переменной и набором входных (независимых) переменных.

Применение регрессионных моделей оказывается особенно полезным в следующих случаях:

- Входные переменные задачи известны или легко поддаются измерению, а выходные нет.
- Значения входных переменных известны из начально, и на их основе требуется предсказать значения выходных переменных.
- Требуется установить причинно-следственные связи между входными и выходными переменными, а также силу этих связей.

В технологиях прогнозирования наиболее широко используется такой вид регрессионной модели, как обобщенная линейная модель. Ее популярность вызвана тем, что многие процессы в управлении, экономике и бизнесе линейны по своей природе. Кроме того, существуют методы, которые позволяют привести нелинейную модель к линейной с минимальными потерями точности.

Обобщенная линейная модель регрессии описывается уравнением:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

где x_i – значение i-го наблюдения;

i – коэффициент регрессии.

Метод декомпозиции временного ряда.

Одним из методов прогнозирования временных рядов является определение факторов, которые влияют на каждое значение временного ряда. Для этого выделяется каждая компонента временного ряда, вычисляется ее вклад в общую составляющую, а затем на его основе прогнозируется будущие значения временного ряда. Данный метод получил название декомпозиции временного ряда.

Термин «декомпозиция» означает, что исходный временной ряд представляется как композиция компонент – тренда, сезонной и циклической. Для построения прогноза выполняется выделение этих компонент из ряда, то есть декомпозиция или разложение ряда по компонентам. Методы декомпозиции могут использоваться для построения как краткосрочных, так и долгосрочных прогнозов.

Фактически декомпозиция – это выделение компонент временного ряда и их проекция на будущее с последующей комбинацией для получения прогноза. Метод был разработан довольно давно, однако сейчас его использует все реже и реже из-за присущих ему ограничений. Проблема заключается в том, что обеспечить достаточно высокую точность прогноза для отдельных компонент очень затруднительно.

Рассмотрим прогнозирование методом декомпозиции с помощью тренда (рис. 2).

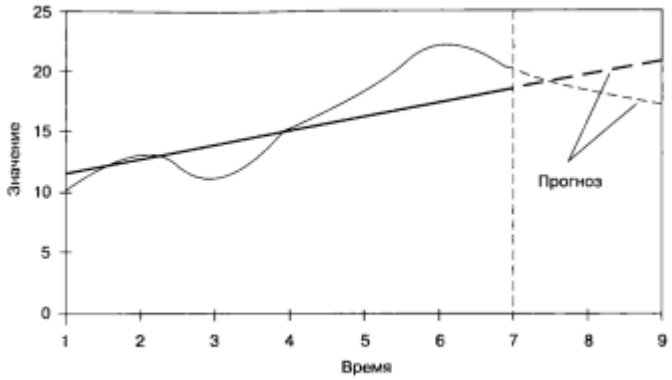


Рис. 2. Прогнозирование с помощью тренда

Если тренд линейный, что типично для многих реальных временных рядов, то он представляет собой прямую линию, описываемую уравнением:

$$y = a + b \times t$$

где y – значение ряда;

a, b – коэффициенты, определяющие расположение и наклон линии тренда;

t – время.

Если уравнение линии тренда известно, с его помощью можно рассчитать значения тренда в любой, в том числе будущий, момент времени. Достаточно воспользоваться уравнением:

$$y_{i+k} = a + b \times (t' + k)$$

где t' – начало прогноза;

k – горизонт прогноза.

При использовании сезонности для прогнозирования методом декомпозиции сначала из временного ряда убирается тренд и слаживается возможная циклическая компонента. Тогда можно считать, что оставшиеся данные будут обусловлены в основном сезонными колебаниями. На основе этих данных вычисляются так называемые сезонные индексы, которые характеризуют изменения временного ряда во времени. Например, временной ряд содержит наблюдения по месяцам в течении года. Сезонный индекс, равный 1, будет установлен для месяца, ожидаемое значение в котором составляет 1/12 от общей суммы по всем месяцам. Если для некоторого месяца устанавливается индекс 1,2, то ожидаемое значение для этого месяца составляет 1/12 + 20 %, а

если 0,8 – то 1/12 - 20 % и т. д. Ясно, что сумма месячных сезонных индексов за год должна равняться 12.

Диаграмма сезонных индексов позволяет оценить вероятный относительный вклад месяцев будущего года в общегодовой показатель на основе сезонной компоненты. Использовать сезонность для прогнозирования можно тогда, когда сезонные колебания имеют хорошую повторяемость.

8.10 Ансамбли моделей. Комбинирование решений. Виды ансамблей.

Ансамбли моделей.

Если обученной модели хорошо удается разделить классы и она допускает мало ошибок классификации, то такая модель может рассматриваться как сильная. Слабая модель, напротив, не позволяет надежно разделять классы или давать точные предсказания, допускает в работе большое количество ошибок. Если в результате обучения мы получили слабую модель, то ее необходимо усовершенствовать, подбирая тип классификатора, алгоритмы и параметры обучения. Но часто встречаются ситуации, когда все возможности совершенствования единственной модели исчерпаны, а качество ее работы по-прежнему неудовлетворительно. Это может быть связано со сложностью решаемой задачи и искомых закономерностей, с низким качеством обучающих данных и другими факторами.

Неизбежно возникает вопрос: как усилить слабую модель, что сделать, для повышения эффективности классификации? Вполне логичным выходом из ситуации является попытка применить к неудачным результатам работы первой модели еще одну модель, задача которой классифицировать те примеры, что остались нераспознанными.

Если и после этого результаты неудовлетворительны, можно применить третью модель и так далее до тех пор, пока не будет получено достаточно точное решение. Таким образом, для решения одной задачи классификации или

регрессии мы применили несколько моделей, при этом нас интересует не результат работы каждой отдельной модели, а результат, который дает весь набор моделей. Такие совокупности моделей называются ансамблями моделей.

Набор моделей, применяемых совместно для решения единственной задачи, называется ансамблем (комитетом) моделей.

Комбинирование решений.

Цель объединения моделей очевидна улучшить (усилить) решение, которое дает отдельная модель. При этом предполагается, что единственная модель никогда не сможет достичь той эффективности, которую обеспечит ансамбль. Использование ансамблей вместо отдельной модели в большинстве случаев позволяет повысить качество решений, однако такой подход связан с рядом проблем, основными из которых являются:

- увеличение временных и вычислительных затрат на обучение нескольких моделей;
- сложность интерпретации результатов;
- неоднозначный выбор методов комбинирования результатов, выдаваемых отдельными моделями.

Перечисленные проблемы аналогичны тем, что возникают при работе нескольких людей экспертов. Действительно, приходится собирать группу экспертов, предоставлять им необходимую информацию, обсуждать задачу и т. д. Все это отнимает намного больше времени, чем принятие решения одним человеком. Сложность интерпретации результатов экспертных оценок также имеет место, ведь каждый эксперт оперирует терминами своей предметной области, формулирует выводы на уровне своего понимания проблемы. И наконец, выбор метода обобщения отдельных заключений экспертов, позволяющего получить наилучшие результаты, не является однозначным.

Виды ансамблей.

В последнее десятилетие ансамбли моделей стали областью очень активных исследований в машинном обучении, что привело к разработке большого числа разнообразных методов формирования ансамблей.

Первым вопросом при формировании ансамбля является выбор базовой модели (base model). Ансамбль в целом может рассматриваться как сложная, составная модель (multiple model), состоящая из отдельных (базовых) моделей.

Здесь возможны два случая.

1. Ансамбль состоит из базовых моделей одного типа, например, только из деревьев решений, только из нейронных сетей и т. д. (рис. 3).

Рис. 3. Однородный ансамбль

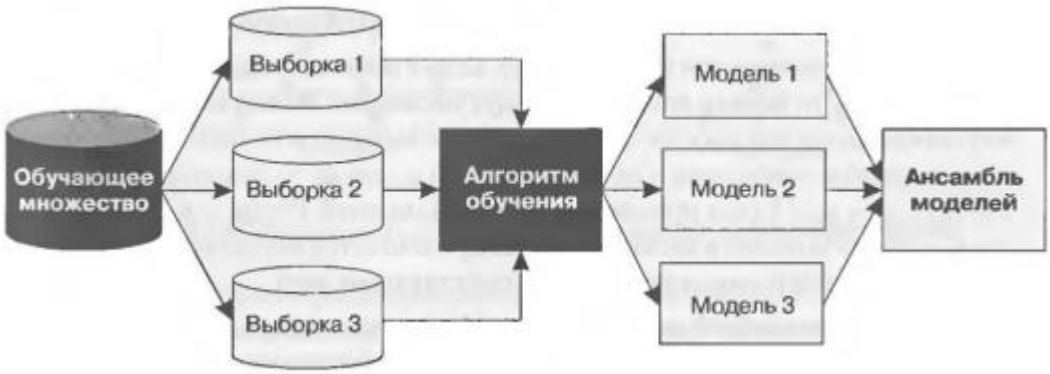
2. Ансамбль состоит из моделей различного типа нейронных сетей, деревьев решений, регрессионных моделей и т. д. (рис. 4).

Рис. 4. Ансамбль, состоящий из моделей различного типа

Каждый подход имеет свои преимущества и недостатки. Использование моделей различных типов дает классификатору дополнительную гибкость. Но, поскольку выход одной модели применяется для формирования обучающего множества для другой, возможно, потребуются дополнительные преобразования, чтобы согласовать входы и выходы моделей.

Второй вопрос: как использовать обучающее множество при построении ансамбля? Здесь также существуют два подхода.

1. Перевыборка (resampling). Из исходного обучающего множества извлекается несколько подвыборок, каждая из которых используется для обучения одной из моделей ансамбля. Данный подход иллюстрируется с помощью рис. 5. Если ансамбль строится на основе моделей различных типов, то для каждого типа будет свой алгоритм обучения.



2. Использование одного обучающего множества для обучения всех моделей ансамбля (рис. 6).

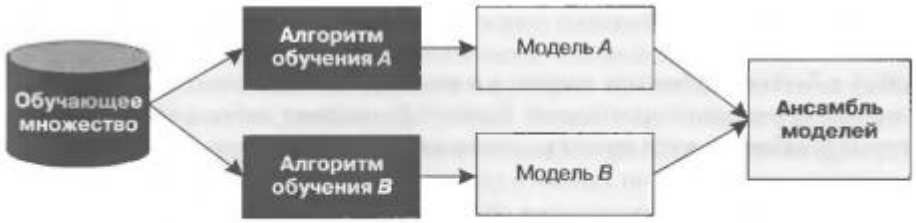


Рис. 6. Использование одного обучающего множества для всех моделей ансамбля

Третий вопрос касается метода комбинирования результатов, выданных отдельными моделями: что будет считаться выходом ансамбля при определенных состояниях выходов моделей? Обычно используются следующие способы комбинирования:

1. Голосование. Применяется в задачах классификации, то есть для категориальной целевой переменной. Выбирается тот класс, который был выдан простым большинством моделей ансамбля. Пусть, например, решается задача бинарной классификации с целевыми переменными, Да и Нет, для чего используется ансамбль, состоящий из трех моделей. Если две модели выдали выход Нет и только одна Да, то общий выход ансамбля будет Нет.

2. Взвешенное голосование. В ансамбле одни модели могут работать лучше, а другие хуже. Соответственно, к результатам одних моделей доверия больше, а к результатам других меньше. Чтобы учесть уровень достоверности результатов, для моделей ансамбля могут быть назначены веса (баллы). Например, в случае, рассмотренном в предыдущем пункте, для моделей, выдавших результат Нет, установлены веса 30 и 40, указывающие на невысокую

достоверность этих результатов, В то же время единственная модель, которая выдала Да, имеет вес 90. Тогда голосование будет производиться с учетом весов моделей: 30 (Нет) + 40 (Нет) == 70 (Нет) < 90 (Да). Таким образом, модель с выходом Да перевесила обе модели с выходом Нет и общий выход ансамбля будет Да.

3. Усреднение (взвешенное или невзвешенное). Если с помощью ансамбля решается задача регрессии, то выходы его моделей будут числовыми. Выход всего ансамбля может определяться как простое среднее значение выходов всех моделей. Например, если в ансамбле три модели и их выходы равны $y1$, $y2$ и $y3$, то выход ансамбля будет $Y = (y1 + y2 + y3)/3$ (для произвольного числа моделей $Y = (y1 + y2 + \cdots + yk)/K$ где K – число моделей в ансамбле)/ Если производится взвешенное усреднение, то выходы моделей умножаются на соответствующие веса.

Исследования ансамблей моделей в Data Mining стали проводиться относительно недавно. Тем не менее к настоящему времени разработано множество различных методов и алгоритмов формирования ансамблей. Среди них наибольшее распространение получили такие методы, как бэггинг и бустинг.

8.11 Ансамбли моделей. Бэггинг - основная идея. Алгоритм и схема процедуры бэггинга.

Ансамбли моделей.

Смотреть вопрос №8.10.

Бэггинг - основная идея.

Сначала на основе исходного множества данных путем случайного отбора формируется несколько выборок. Они содержат такое же количество примеров, что и исходное множество. Но, поскольку отбор производится случайно, набор примеров в этих выборках будет различным: одни примеры могут быть отобраны по нескольку раз, а другие ни разу. Затем на основе каждой выборки строится классификатор и выходы всех классификаторов комбинируются (агрегируются)

путем голосования или простого усреднения. Ожидается, что полученный результат будет намного точнее любой одиночной модели, построенной на основе исходного набора данных. Обобщенная схема процедуры бэггинга представлена на рис.7 (на примере дерева решений).

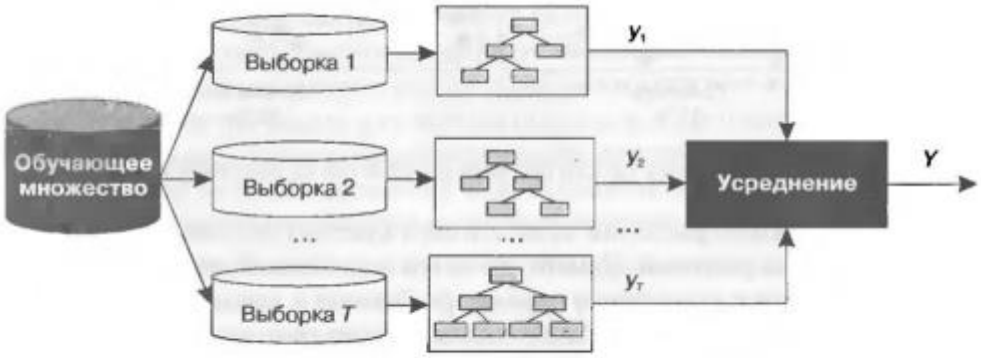


Рис. 7. Схема процедуры бэггинга

Таким образом, бэггинг включает следующие шаги.

1. Из обучающего множества извлекается заданное количество выборок одинакового размера.

2. На основе каждой выборки строится модель.

3. Определяется общий результат путем голосования или усреднения выходов моделей.

Остановка процедуры бэггинга производится на основе следующих критериев:

- На некоторой итерации t ошибка ε^t классификатора C^t становится равной 0 или больше либо равной 0,5. В этом случае процедура бэггинга останавливается, а последний классификатор удаляется: T = $t - 1$. Таким образом, бэггинг не пускает в ансамбль «плохие» классификаторы.
- Число итераций достигло заданного пользователем предела T. Как и для большинства других итеративных алгоритмов Data Mining, однозначного ответа на вопрос, каково достаточное число итераций, не существует. Оно подбирается эмпирическим путем.
- Бэггинг, хотя и в меньшей степени, чем отдельные модели, склонен к переобучению. Поэтому критерием для остановки процедуры может

служить возрастание ошибки на тестовом множестве.э

Случайный лес – алгоритм машинного обучения, заключающийся в использовании ансамбля решающих деревьев каждое из которых само по себе даёт очень невысокое качество классификации, но за счёт их большого количества получается необходимый результат. Точно так же, как инвестиции с низкими корреляциями (например, акции и облигации) объединяются, чтобы сформировать портфель больший, чем сумма его частей.

Алгоритм и схема процедуры бэггинга.

Построим ансамбль алгоритмов, где базовый алгоритм — это решающее дерево. Будем строить по следующей схеме:

1. Для построения i-го дерева:

а. Сначала, как в обычном бэггинге, из обучающей выборки X выбирается с возвращением случайная подвыборка $X\ i$ того же размера, что и X/

б. В процессе обучения каждого дерева в каждой вершине случайно выбираются $n < N$ признаков, где N – полное число признаков (метод случайных подпространств), и среди них ищется оптимальный сплит (разделение). Такой приём как раз позволяет управлять степенью скоррелированности базовых алгоритмов.

2. Чтобы получить предсказание ансамбля на тестовом объекте, усредняем отдельные ответы деревьев (для регрессии) или берём самый популярный класс (для классификации).

3. В результате построили Random Forest (случайный лес) — комбинацию бэггинга и метода случайных подпространств над решающими деревьями.

Достоинства:

- Способность эффективно обрабатывать данные с большим числом признаков и классов;
- Нечувствительность к любым монотонным преобразованиям значений признаков;
- Одинаково хорошо обрабатываются как непрерывные, так и

дискретные признаки;

- Существуют методы оценивания значимости отдельных признаков;
- Внутренняя оценка способности модели к обобщению;
- Высокая параллелизуемость и масштабируемость;
- Случайные леса очень гибки и обладают очень высокой точностью.

Недостатки:

- Большой размер получающихся моделей;
- Построение леса сложнее и отнимает больше времени;
- Чем больше объем, тем меньше интуитивное понимание.

8.12 Ансамбли моделей. Бустинг – основная идея. Процедура бустинга. Отличие от бэггинга.

Ансамбли моделей.

Смотреть вопрос №8.10.

Бустинг – основная идея.

По сравнению с бэггингом бустинг (boosting) несколько более сложная процедура, но во многих случаях работает эффективнее. Как и бэггинг, бустинг использует неустойчивость алгоритмов обучения и начинает создание ансамбля на основе единственного исходного множества. Но если в бэггинге модели строятся параллельно и независимо друг от друга, то в бустинге каждая новая модель строится на основе результатов ранее построенных моделей, то есть модели создаются последовательно.

Бустинг создает новые модели таким образом, чтобы они дополняли ранее построенные, выполняли ту работу, которую другие модели сделать не смогли на предыдущих шагах. И наконец, последнее отличие бустинга от бэггинга заключается в том, что всем построенным моделям в зависимости от их точности присваиваются веса (бэггинг, напомним, использует взвешенное голосование или усреднение).

В настоящее время разработано большое количество различных модификаций бустинга. Рассмотрим один из наиболее популярных алгоритмов - AdaBoost.M1, который предназначен для решения задач классификации.

Вместо извлечения выборок из исходного множества данных бустинг в качестве возмущающего фактора применяет взвешивание примеров. Вес каждого примера устанавливается в соответствии с его влиянием на обучение классификатора.

На каждой итерации вектор весов подстраивается таким образом, чтобы отражать эффективность данного классификатора. В результате вес неправильно классифицированных примеров увеличивается. Итоговый классификатор также агрегирует обученные классификаторы путем голосования, но теперь голос классификатора является функцией его точности.

Обозначим вес примера x на итерации t как ω_x^t, при этом вес на первой итерации задается как $\omega_x^t = \frac{1}{N}$ для каждого x (N – число примеров). На каждой итерации $t = 1,2 \ldots T$ классификатор C^t конструируется из данных примеров в Соответствии с распределением их весов ω^t (то есть как будто вес ω_x^t отражает вероятность появления примера x). Ошибка ε^t классификатора t также измеряется относительно весов и представляет собой сумму весов примеров, которые были классифицированы неправильно. Когда ε^t становится больше 0,5, итерации прекращаются последний классификатор удаляется и T изменяется на $t-1$. Наоборот, если $\varepsilon^t = 0$ (классификатор C^t правильно классифицировал все примеры), итерации останавливаются и $T = t$. В остальных случаях вектор весов ω^{t+1} для следующей итерации генерируется путем умножения текущих весов примеров, которые были правильно распознаны классификатором C^t, на коэффициент $\beta^t = \varepsilon^t/(1-\varepsilon^t)$, а затем нормируется так, чтобы $\sum_x \omega^{t+1} = 1$. Тогда:

$$\omega^{t+1} = \omega^t \frac{\varepsilon^t}{(1-\varepsilon^t)} \quad (1)$$

Итоговый классификатор C^* получается путем суммирования голосов классификаторов C^1, C^2 ... C^T , где голос классификатора C^t определяется как $log(1/\beta^t)$ единиц.

Если ошибка отдельного классификатора ε^t всегда меньше 0,5. То значение ошибки итогового классификатора C^* экспоненциально стремится к 0 с увеличением числа итераций t. Последовательность слабых классификаторов C^1, C^2 ... C^T может быть усилена до классификатора C^*, который обычно получается более точным, чем отдельные классификаторы, Конечно, при этом нельзя гарантировать высокую обобщающую способность C^*.

Процедура бустинга.

1. Для всех примеров исходного множества данных устанавливаются равные начальные веса ω_0.

2. На основе взвешенного набора примеров строится классификатор C^t, вычисляется и запоминается выходная ошибка данного классификатора ε^t.

3. Рассчитывается коррекция весов примеров обучающего множества, и веса корректируются по формуле (1).

4. Если ошибка $\varepsilon^t = 0$ или $\varepsilon^t \geq 0{,}5$ то классификатор C^t удаляется, и процедура бустинга останавливается.

5. В противном случае осуществляется переход на шаг 2 и начинается следующая итерация.

Таким образом, параметрами, настраиваемыми на каждой итерации, являются веса примеров. При этом чем больше раз пример был неправильно распознан предыдущими моделями, тем выше его вес. Вес можно рассматривать как вероятность попадания примера на следующую итерацию.

Отличие от бэггинга.

Да.