1. Понятие машинного обучения.

Машинное обучение заключается в извлечении знаний из данных. Это научная область, находящаяся на пересечении статистики, искусственного интеллекта и компьютерных наук и также известная как прогнозная аналитика или статистическое обучение. В последние годы применение методов машинного обучения в повседневной жизни стало обыденным явлением. Многие современные веб-сайты и устройства используют алгоритмы машинного обучения, начиная с автоматических рекомендаций по просмотру фильмов, заказа еды или покупки продуктов, и заканчивая персонализированными онлайн- радиотрансляциями и распознаванием друзей на фотографиях.

Артур Самуэль нависал комньютерную программу для игры в шашки. Каждому положению на доске присваивался некий вес, базирующийся на вероят­ ности выигрыша. Изначально вероятность определялась по формуле, в которой учитывались такие факторы, как количество шашек на каждой стороне и количество дамок. Подход работал, но Самуэль придумал, ка­ ким образом можно повысить его эффективность. Сыграв с программой тысячу партий, он использовал их результаты для уточнения позиционных весов. Самуэль написал компьютерную программу, которая могла по мере нако­ пления опыта улучшать собственные результаты. Программа училась - так зародилось машинное обучение.

1. Задачи, решаемые с применением машинного обучения.

Компания Kaggle объявила конкурс на алгоритм, максимально точно от­ личающий собак от кошек. Для тренировки участникам предоставили 25000 изображений с метками, укаэывающими, кто именно изображен на картинке после обучения каждый алгоритм должен был классифи­цировать 12500 не имеющих меток тестовых изображений.

Машинное обучение применяется к широкому кругу экономических задач - от обнаружения мошенничества до выбора целевой аудитории и рекомендаций товара, наблюдения за производством в реальном времени, анализа тональности текстов и медининской лиагностики. Оно может взять на себя задачи, которые невозможно выполнить вручную из-за огромного количества подлежащих обработке данных.

* Классификация – на основе данных определяется дискретный класс для каждого объекта
* Регрессия – на основе данных предсказывается фактическое значение параметра
* Рекомендация – предсказывается альтернатива, которую предпочтет пользователь
* Заполнение пропусков – вывод значений от существующих входных данных

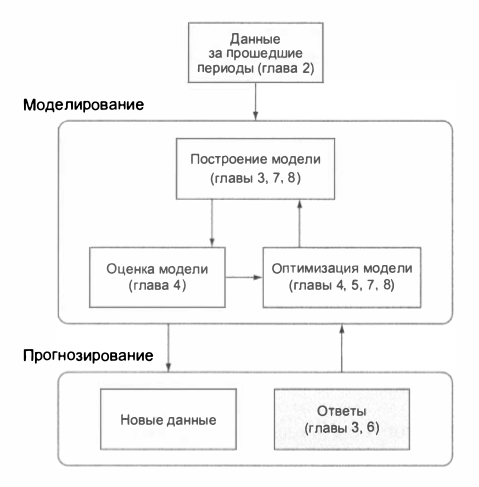
1. Преимущества машинного обучения. Сложности, возникающие в процессе машинного обучения.

* *Точность.* Машинное обучение использует данные для создания принимающей̆ решение программы, оптимизированной под по­ставленную задачу. По мере накопления данных автоматически возрастает точность прогнозов.
* *Автоматизация.* По мере подтверждения и отбрасывания ответов МI.-модель может автоматически обнаруживать новые шаблоны. Это позволяет встраивать машинное обучение непосредственно в автоматизированные рабочие процессы.
* *Скорость*. Машинное обучение дает ответы за доли секунды после поступления новой информации, позволяя системам реагировать в реальном времени.
* *Возможность настройки*. Многие задачи, управляемые данными, можно решить с помощью машинного обучения. Модели строятся на базе ваших собственных данных и донускают настройку вод любую систему мер, нрипятую в вашем бизнесе.
* *Масштабируемость*. При росте бизнеса ML-модель легко приспосабливается к увеличивающимся объемам данных. Некоторые алгоритмы можно использовап, для обработки множества данных на разных вычислительных машинах в облаке.

Сложности:

* Чаще всего требуется получить данные в годной к употреблению форме. Было подсчитано, что специалисты по работе с данными тратят на их подготовку 80% времени.
* Фактически наша задача - сформулировать проблему таким образом, чтобы к ней можно было применить методы машинного обучения и по­ лучить имеющие нрактическую ценность и измеримые результаты.
* Когда нужен сложный резулыат, выбор алгоритма и способа его применения сам по себе начинает требовать гигантских усилий. Исследователи, работающие в области кардиологии над прогноэами вероятности после­ операционных осложнений, обладают умопомрачителыным количеством данных на каждого пациента, Проектирование признаков (feature engineering) представляет собой процесс преобразования таких входных данных в подходящие для предсказывающей модели признаки.
* Нельзя не упомянуть и проклятие всех, кто эанимается соэданием про­ гнозирующих моделей, - модель, которая идеально работает на обучающих данных, но демонстрирует полную неспособность к достоверным прогнозам на основании неизвестных ранее данных. Причиной в большинстве случаев становится переобучение (overfitting).

1. Базовый процесс интеграции моделей машинного обучения в приложения или конвейеры данных.



1. Сбор и подготовка данных для систем с машинным обучением.

Сбор и подготовка данных для систем с машинным обучением обычно влечет за собой их представление в виде таблицы, если изначально они имеют другую форму. Представьте, что данные распределены по строкам и столбцам, причем кажлая строка соответствует иэучаемому экземпляру (instance), а столбце - значению этого экземпляра. Столбцы, как правило, содержат данные олного тина, в то время как данные в стро­ках принадлежат разным типам.

Реальные данные могут быть запутаны разными способами. Представьте, что на этапе сбора данных измерить какое-то значение не представляется возможным. Нельзя и вернутся ваэад, чтобы отыскать недостающий фрагмент информации. В подобных случаях некоторые ячейки таблицы останутся незаполненными, что усложнит как ностросние модели, так и последующее прогнозирование. Иногда сбор данных осуществляется вручную, а мы все знаем, как легко делаются ошибки при выполнении повторяющихся задач. В результате часть сведений оказывается некорректной. Вы должны уметь работать с подобными сценариями или, по крайней мере, знать, как конкретный алгоритм ведет себя при наrшчии недостоверных данных.

1. Обучение модели на данных.

Первый этап построения успешной системы с машинным обучением - это формулировка вопроса, ответ на который должны дать наши данные.

Наш ML-алгоритм должен понять каким образом набор входных признаков позволяет успешно предсказывать значение целевой переменной.

Одним из преимуществ алгоритмов машинного обучения в сравнении с другими распространёнными методами является умение обрабатывать множество признаков.

Некоторые алгоритмы в достаточной степени невосприимчивы к неинформативным приэнакам, в то время как другие дают более точные нредсказания, когда такие признаки убираются из рассмотрения. Впрочем, целые сведения норой могут быть извлечены и иэ неинфор­ мативных на первый взгляд признаков. Такой вариант усовершенствования данных, называемый извлечеиием признаков (featшe extraction), крайне важен в реальных МL-проектах.

Предсказанные целевая переменная возвращается в той форме, в которой она фигурировала во взятых для обучения мысли исходных данных. Прогнозирование с помощью модели, по сути, является заполнением пустого столбца новыми значениями. Некоторые МL-алгоритмы также включают в результат своей работы связанные с каждым классом вероят­ ности. В рассматриваемом примере вероятностная модель, выдаст для каждого нового заявителя два эначения: вероятность, что человек состоит в браке, и вероятность того, что он свободен.

1. Оценка производительности модели.

Системы с машинным обучением практически никогла не используются, пока не будет проверена их производительность. И теперь нужен хитрый прием, который позволит понять, насколько готовая мо­дель справляется со стоящими перед ней задачами.

Возьмем набор данных и представим, что целевая переменная неизвестна. Построим на их основе модель и используем их в качестве тестовых данных для нескольких прогнозов.

Теперь можно сравнить результаты прогнозов с известными истинными значениями и оценить точность модели.

1. Оптимизация производительности модели.

* Редактироваиие параметров модели. Каждый МL-алгоритм об­ ладает набором параметров, оптимальные значения которых зачастую зависят от типа и структуры данных. При этом на производительность модели может влиять значение как любого отдельного параметра, так и их произвольной комбинации.
* Выбор подмножества признаков. Зачастую задачи, сняэанные с машинным обучением, включают множество признаков, и вносимые ими помехи порой мешают алгоритму обнаружить, верную закономерность, даже если сами по себе эти признаки являются ин­формативными. Во многих случаях наличие большого количества данных - благо, но иногда это становится нроклятием. А так как заранее никогда не известно, как это скажется на эффективности модели, следует тщательно выбирать признаки, которые дадут наиболее универсальную и точную модель.
* Предварительная обработка данных. В интернете можно найти наборы данных для машинного обучения, которые без проблем скармливаются многим МL-алгоритмам. Реальные же данные в большинстве случаев находятся далеко не в таком чистом виде и подлежат очистке и обработке. Этот процесс называют выпасом данных (data munging или data wrangling). Набор может включать имена, которые п ишутся по-разному, хотя и относятся к одной и той же сущности, а также отсутствующие или недостоверные значения. Все это негативным образом сказывается на эффективности модели. Может показаться, что речь идет о редких случаях, но вы удивитесь, когда увидите, насколько часто подобные вещи встречаются даже в тщательно раэработанных управляемых дан­ ными структурах.

1. Предварительная обработка данных и проектирование признаков в моделях машинного обучения.

Можно не просто очистить данные, а сделать еще один шаг и извлечь из них дополнительное значение, положительно влияющее на производительность модели.

В любой предметной области требуются специальные знания, поэволя­ ющие решать, какие именно данные подлежат сбору. Именно эти энания применяются для извлечения из собранных данных ценной информации, которая добавляется к признакам строящейся модели.

Примеры признаков для проектирования:

* Дата и время
* Местоположение
* Цифровые средства коммуникации



1. Непрерывное совершенствование моделей машинного обучения. Масштабирование моделей.

Желательно, чтобы модель постепенно совершенствовалась и адаптировалась к изменениям в этих данных. Существуют МL-алгоритмы, поддерживающие данный тип дииамического обучения (online learning).



1. Определение набора входных признаков в задачах машинного обучения.

Всем известно, что современные наборы данных увеличиваются в раэ­ мерах быстрее, чем когда-либо раныне. Наборы для методов машинного обучения с учителем, в которых целевые ответы входят в обучающую выборку, были традиционно небольшими, так как для получения ответов прибегали к помощи человека. В настоящее время множество данных (включая ответы) продуцируется непосредственно измерительными элементами, машинами или компьютерами, и для обработки таких объ­емов требуются уже масштабируемые ML-алгоритмы.

1. Наблюдаемое значение целевой переменной в задачах машинного обучения.

Одна из самых больших трудностей на нервом этане машинного обучения с учителем - формирование обучающей выборки с известными значени­ ями целевых переменных. Зачастую для этого требуется работа существу­ ющей, условно-оптимальной системы до тех пор, пока не будут собраны все данные.

Сбор экземпляров известных целевых переменных зачастую - тяжелый и кропотливый труд, но выгоды от переход к МL-решению, скорее всего, многократно компенсируют временные и денежные затраты.

Способы получения достоверных значений целевой переменной:

* нанять аналитика, который будет вручную просматривать текущие данные и данные эа прошлые периоды с целью определения или оценки значений целевой переменной;
* привлечь к оценке значений целевой переменной широкий круг лиц;
* по результатам проведенных мероприятий опрашивать клиентов и проводить с ними другие практические эксперименты;
* запускать управляемые эксперименты (например, А/В-тестирование) и следить за ответами.

1. Определение размера обучающей выборки для построения и запуска модели машинного обучения.

Какой размер обучающей выборки достаточен для построения и запуска МL-модели? К сожале­нию, ответ настолько зависит от конкретной задачи, что универсальных рекомендаций попросту не существует. Нет и общих правил.

Вот факторы, от которых зависит количество необходимых данных;

* Сложность задачи. Можно ли описать связь между входными признаками и целевой неременной простым шаблоном или же она запутанна и не имеет линейной зависимости?
* Требования к точности. Если достаточно всего 60% успешных результатов, вы обойдётесь меньшей обучающей выборкой, чем в случае, когда необходимо получить, 95% успешных результатов.
* Размерность пространства признаков. Если доступно всего два входных признака, обучающих данных потребуется меньше, чем при наличии 2000 таких признаков.

Нужно запомнить следующий руководящий принцип: по мере увеличения обучающей выборки модели (в среднем) становятся более точными. (Предполагается, что мы получаем образцы продолжающегося процесса генерации данных, как будет подробно рассмотрено в следующем разде­ле.) Большая обучающая выборка обеспечивает более высокую точность из-за того, что МL-модели базируются именно на данных. Взаимосвязь между признаками и целевой переменной выводится целиком из обучаю­ щей выборки, соответственно чем больше размер этой выборки, тем выше способность модели распознавать и фиксировать, менее выраженные шаблоны и соотношения.

1. Репрезентативность обучающей выборки при построении моделей машинного обучения.

Так как целью машинного обучения с учителем является генерания точных предсказаний для новых данных, крайне важно, чтобы обучающая выборка включала в себя акзсмпляры тех ти­ пов, для которых будет создаваться прогноз. Выборка с эюемплярами, не имеющими отношения к будущим данным, называется систематической.

Отсутствие репрезентативности в обучающей выборке может быть об­ условлено рядом факторов:

* Получить экспериментальные значения целевой переменной удалось только для определенного, содержащего ошибку подмноже­ства данных.
* Свойства экземпляров со временем менялись.
* Набор входных признаков меняется со временем.

В каждом из вышеуказанных случаем МL-модель, хорошо работающая с обучающей выборкой, не сможет хорошо выполнить экстраполяцию для новых данных. Другими словами, если модель обучали на данных о яблоках, врядли она сможет точно предсказывать урожаи апельсинов.

Чтобы избежать, подобных проблем. обучающая выборка должна максимально репрезентативно представлять будущие данные. Значит, требует­ ся процесс структурирования набора тренировочных данных, убирающий систематические ошибки.

1. Категориальные признаки в ML-наборах данных.

Вторым наиболее распространенным типом признаков является ка­ тегориальный. Признак считается таковым, сели его значения можно отнести к какой-то группе, при этом нам не важен их порядок. Иногда принадлежность к этому типу определяется достаточно легко (напри­мер, когда параметры принимают всего несколько строковых значений, скажем, спам и обычная почта). Но бывает и так, что раэница между численным (целым) и категориальным приэнаком не очевидна. Любой из типов может оказаться правильным представлением, и при этом выбор влияет на производительность модели. Скажем, день недели допустимо представить как в виде числа (пронумеровав дни, начиная с воскресения), так и в виде категории (взяв за основу названия).

1. Преобразование категориальных признаков в численные.

Можно присвоить, каждой категории номер, но такие зашифрованные данные нельзя использовать в качестве истинных категориальных признаков, так как при этом добавляется случайным образом выбранный вами порядок категорий. Ведь одним из свойств категориальных признаков является их неупорядоченность. Куда лучше преобразовать каждую категорию в отдельный двоичный нризнак, имеющий значение 1 для экземпляров, попадающих в категорию, и О для не понадающих. В результате каждый категориальный признак преобразу­ется в набор двоичных признаков, по одному на категорию. Сконструи­рованные таким способом признаки иногда называют вспомогательными переменными (dummy variables).

Но есть и алгоритмы (например, с деревьями принятия решений, такие как «случайный лее»), которые используют категориальные признаки в исходном виде. Для исключительно категориальных наборов данных они часто дают очень хорошие результаты.

1. Подходы к работе с отсутствующими данными.

Обработка двух основных типов отсутствующих данных происходит по­ разному. Во-первых, иногда сам факт отсутствия данных может быть информацией, полезной для МL-алгоритма. Разумеется, если измерение эначений было попросту невоэможным, этот факт не несет никакого глубинного смысла.

Рассмотрим сначала ииформативные отсутствующие данные. Если нам кажется, что в данных не хватает какой-то информации, значит, мы хотим, чтобы МL-алгоритм смог ею воспользоваться и теоретически увеличить точность предсказаний. Для этого отсутствующее :шачение нужно привести к формату остальных значений в столбце. Для число­вых столбцов это осуществляется путем присваивания значения -1 или -999, в зависимости от того, какие типичные значения имеют ненулевые параметры. Выберите для обозначения отсутствующих значений число с одного из концов числового снектра и не забывайте, что для численных столбцов важен порядок следования. При этом нельзя выбирать вариант из центральной части диапазона значений.

Для категориальных столбцов с потенциально информативными отсутствующими данными можно создать новую категорию, назвав ее Мissing, None или другим подобным образом, после чего новый категориальный признак обрабатывается обычным способом (например, с помощью опи­санной в предыдущем разделе техники)

Если же отсутствие значений отдельных экземпляров не несет информационной нагрузки, порядок действий будет другим. Добавлять специальное число или категорию уже нельзя, так как есть риск, что значение окажется некорректным. Скажем, вставив в пустые ячейки столбца «Возраст» значение -1, вы, скорее всего, навредите модели, с непонятной целью внеся путанину в диапазон возрастов. Некоторые ML-алгоритмы нросто игнорируют такие пропуски. Если же алгоритм этот нс умеет, данные следует предварительно обработать, вообще убрав отсутствующие значения или заменив их неким оценочным значением. Такой подход к обработке отсутствующих данных называется заполнением пропусков (imputation).

В случае большого набора данных с небольшим количеством отсутствующих значений самое простое отбросить их. Если же значения отсутствуют в изрядной части наблюдений, потеря данных уменьшит прогностическую силу вашей модели. Более того, если распределение наблюдений с отсутствующими значениями по набору данных не является случайным, отбрасывание может внести неожиданную систематическую ошибку.

Есть и другой простой подход. Можно предположить, что экземпляры данных обладают некой временной упорядоченностью, и вставить на места пропусков значения из предшествующей строки этого же столбца. Нам неоткуда получить другую информацию, поэтому мы просто пред­ полагаем, что при переходе от одного экземпляра к другому результат измерений не изменился. Излишне упоминать, что зачастую подобное нредположение некорректно. Но еще менее корректно, например, запол­ нять пустые ячейки нулями, особенно если данные представляют собой набор последовательных наблюдений (вчерашняя температура вполне может применяться и для оценки сегодняшней). Кроме того, с огромными наборами данных далеко не всегда получается использовать более слож­ные методы, и тогда на помощь придут эти простые приемы.

Когда это возможно, лучше использовать большие фрагменты существующих данных для приблизительной оценки отсутствующих зна­ чений. Их можно заменить средним значением или медианой столб­ца. При отсутствии другой информации предполагается, что среднее значение ближе всего к истине. Но в зависимости от распределения значений в столбце иногда лучше использовать медиану; среднее зна­ чение чувствительно к выбросам. Перечисленные техники широко при­ меняются в современном машинном обучении и прекрасно работают во многих случаях. Но, заменяя все отсутствующие данные одним новым значением, вы снижаете наблюдаемость потенциальной корреляции с другими переменными, которая может помочь алгоритму распознать имеющиеся закономерности.

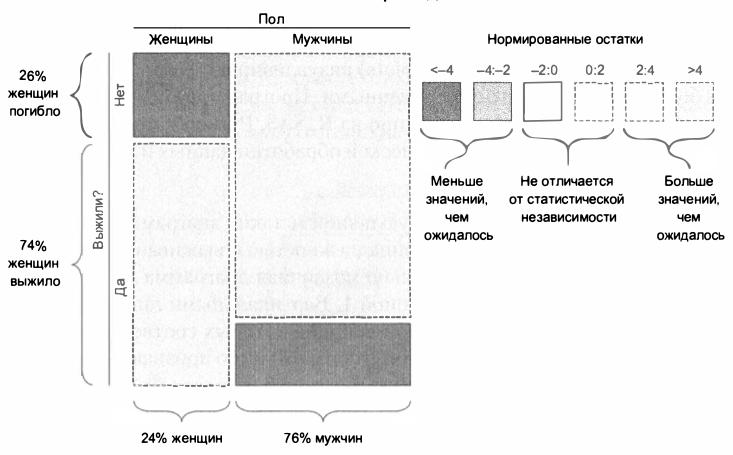
1. Нормализация данных для ML-алгоритмов.

Некоторые ML-алгоритмы требуют, чтобы данные были нормализованными. Это означает, что каждый признак обрабатывается с целью его подгонки под единую числовую шкалу. Диапазон значений признака может влиять на его важность относительно прочих признаков. Если значения одного признака варьируются от 0 до 10, а второго - от 0 до 1, вес первого признака по отношению ко второму составит 10. Иногда вес конкретного признака регулируется вручную, по, как правило, лучше оставить определение относительных весов на откуп МL-алгоритму. Но чтобы гарантировать, что все признаки учитываются одинаково, данные требуется нормализовать. Зачастую нормализация сводится к переводу в диапазон от 0 до 1 или от -1 до 1.

1. Визуализация данных в задачах машинного обучения.

Между сбором/предварительной обработкой данных и построением МL-модели находится важный этап визуализации. Он предназначен для про­верки работоспособности обучающих признаков и целевой переменной. Простые техники визуализации позволяют увидеть, как входные призна­ки связаны с целевой переменной. Эта информация служит ориентиром при построении модели и помогает лучше понять модель и генерируемые ею прогнозы. Более того, визуализация зачастую дает прелставление о степени репрезентативности обучающей выборки и о том, экземпляров каких типов там, возможно, не хватает.

Мозаичные диаграммы визуализируют связи между двумя и более категориальными переменными. Программное обеспечение для создания таких диаграмм достунно на R, SAS, Pytlюn и прочих языках, предназначенных для статистической обработки данных и работы с графикой.



Диаграммы размаха, или «ящики с усами», - это стандартная статистическая техника визуализации распределения численных переменных. Для одной переменной такая диаграмма показывает квартили ее распределения: минимум, 25-й процентиль, мелиану, 75-й процентиль и максимум значения. Диаграмма размаха одной переменной дает представление о центре, разбросе и асимметрии в распределении значений, а также о наличии выбросов.

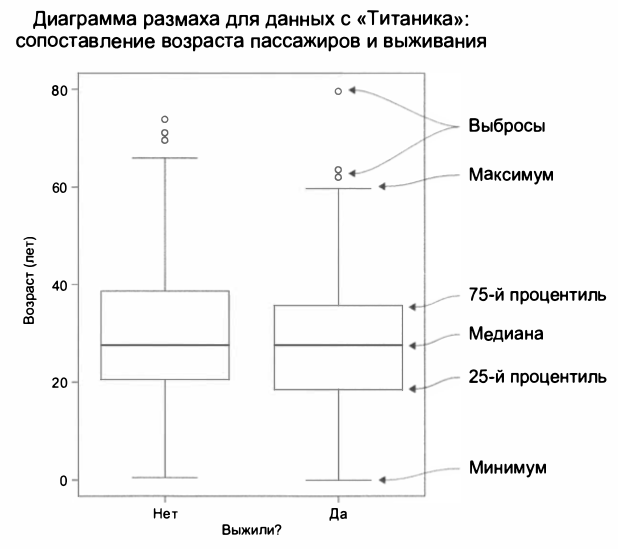


График плотности (density plot) отображает распределение одной переменной куда детальнее, чем диаграмма размаха. Во-первых, рассчитыва­ ется сглаженная оценка распределения вероятности переменной (обычно для этого используется техника, называемая ядериым илажuваиuем (kernel smoothing). Во-вторых, распределение отображается в виде кри­вой, показывая значения, которые, скорее всего, будет иметь переменная. Создав график нлотности для неременной отклика из каждой катеюрии входного признака, вы сможете легко визуализировать любые расхож­дения значений для разных категорий. Графики шюпюсти аналогичны гистограммам, но их сглаженная природа упрощает процесс визуализации набора распределений на одном рисунке.

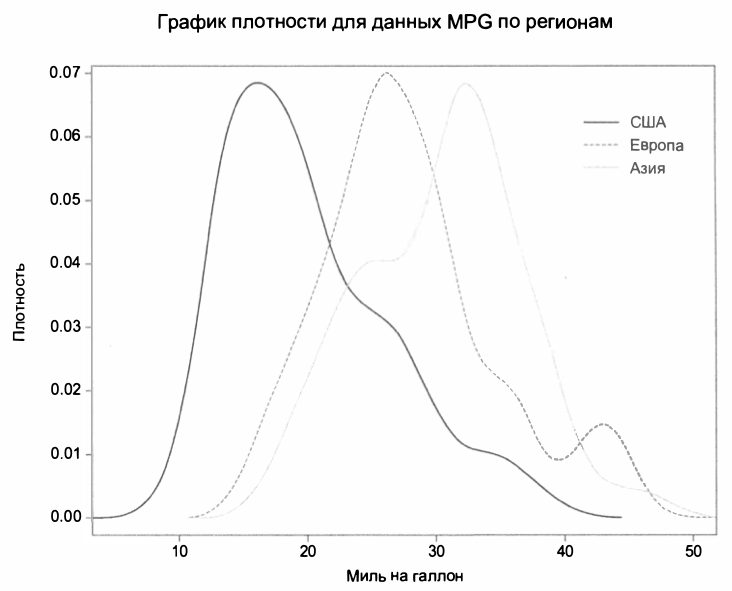


Диаграмма рассеяния (scatter plot) даст простую визуализанию соотношения между двумя численными переменными. Это один из самых популярных графических инструментов. Значение признака на такой диаграмме отмечается относительно значения неременной отклика, и каждый экземпляр представляется в виде точки. Несмотря на свою простоту, диаграммы рассеяния обнаруживают как линейные, так и нели­ нейные соотношения между входной нсременной и переменной отклика.

