

ГЛАВА 8

ОБУЧЕНИЕ БЕЗ УЧИТЕЛЯ

Если вы родитель, все тайны обучения разворачивались прямо на ваших глазах в первые три года жизни ребенка. Новорожденный не умеет говорить, ходить, узнавать предметы и даже не понимает, что то, на что он смотрит, будет существовать и когда он отвернется. Но проходит месяц за месяцем, и маленькими и большими шажками, путем проб, ошибок и больших когнитивных скачков ребенок разбирается, как устроен мир, как ведут себя люди, как с ними общаться. К третьему году плоды обучения сливаются в стабильное «я», в поток сознания, который не прекратится до самой смерти. Более старшие дети и взрослые способны «путешествовать во времени» — вспоминать прошлое, но лишь до этой границы. Если бы мы могли вернуться в младенчество и раннее детство и снова увидеть мир глазами маленького ребенка, многое, что озадачивает нас в механизмах обучения и даже самого бытия, внезапно стало бы очевидным. Но пока величайшая тайна Вселенной — это не ее зарождение или границы и не нити, из которых она соткана, а то, что происходит в мозге маленького ребенка: как из массы серого желе вырастает средоточие сознания.

Хотя наука о механизмах обучения детей все еще молода и исследования начались всего несколько десятилетий назад, ученые уже добились замечательных успехов. Младенцы не умеют заполнять анкеты и не соблюдают протоколов, однако удивительно много информации о том, что происходит у них в голове, можно получить благодаря видеозаписи и изучению их реакций во время эксперимента. Складывается связная картина: разум младенца — это не просто реализация заложенной генетической программы и не биологический прибор для фиксирования корреляций данных, получаемых из органов чувств. Разум ребенка сам активно синтезирует реальность, и со временем она меняется довольно радикально.

Очень удобно, что ученые-когнитивисты все чаще выражают теории детского обучения в форме алгоритмов. Это вдохновляет многих исследователей машинного обучения — ведь все, что нужно, уже есть там, в мозге ребенка, и надо только каким-то образом ухватить суть и записать ее в компьютерном коде. Некоторые ученые даже утверждают, что для создания разумных машин нужно сконструировать робота-ребенка и позволить ему ощутить мир так, как это делают обычные дети. Мы, исследователи, станем ему родителями (может быть, это будет краудсорсинг, и термин «глобальная деревня»¹ приобретет совершенно новое значение). Маленький Робби — давайте назовем его в честь пухлого, но высокого робота из «Запретной планеты»² — единственный робот-ребенок, которого нам надо построить. Как только он обучится всему, что человек знает в три года, проблема искусственного интеллекта будет решена. После этого можно скопировать содержимое его мозга в столько роботов, во сколько захотим, и они будут развиваться дальше: самое сложное уже сделано.

Вопрос, конечно, в том, какие алгоритмы должны работать в мозге Робби в момент рождения. Ученые, находящиеся под влиянием детской психологии, косо смотрят на нейронные сети, потому что работа нейронов на микроскопическом уровне кажется бесконечно далекой от сложности даже простейших действий ребенка: потянуться к предмету, схватить его и рассмотреть широко распахнутыми, полными любопытства глазами. Чтобы за деревьями увидеть планету, обучение ребенка придется моделировать на более высоком уровне абстракции. Самое удивительное, наверное, то, что дети учатся в основном самостоятельно, без надзора, хотя, несомненно, получают огромную помощь от своих родителей. Ни один из алгоритмов, которые мы до сих пор видели, на это не способен, но вскоре мы познакомимся с несколькими вариантами и на шаг приблизимся к Верховному алгоритму.

Как свести рыбака с рыбаком

Мы нажимаем кнопку «Включить», Робби открывает глаза-видеокамеры в первый раз, и его сразу заливает «цветущий и жужжащий беспорядок» мира, как сказал Уильям Джеймс. Новые изображения возникают десятками

¹ Понятие, введенное для понимания современного общества, в котором благодаря развитию электронных коммуникаций расстояние потеряло значение.

² «Запретная планета» (Forbidden Planet) — фильм режиссера Фреда Уилкокка, снятый в 1956 году и ставший классикой жанра кинофантастики.

в секунду, и одна из первоочередных задач — научиться организовывать их в более крупные элементы: реальный мир состоит не из случайных пикселей, которые каждое мгновение меняются, как им вздумается, а из стабильных во времени объектов. Если мама отошла подальше, вместо нее не появится «уменьшенная мама». Если на стол поставить тарелку, в столе не появится белая дырка. Младенец не отреагирует, если плюшевый мишка скроется за ширмой и вместо него появится самолет, а годовалый ребенок удивится: он каким-то образом уже сообразил, что мишки отличаются от самолетов и не могут просто так превращаться друг в друга. Вскоре после этого он разберется, что некоторые предметы похожи друг на друга, и начнет формировать категории. Если девятимесячному малышу дать гору игрушечных лошадок и карандашей, он и не подумает их разделить, а в полтора года уже догадается.

Организация мира в предметы и категории совершенно естественна для взрослого, но не для младенца, и в еще меньшей степени для робота Робби. Можно, конечно, одарить его зрительной корой в виде многослойного перцептрона и показать подписанные примеры всех предметов и категорий в мире — вот мама рядом, а вот мама далеко, — но так мы никогда не закончим. На самом деле нам нужен алгоритм, который будет спонтанно группировать схожие объекты и разные изображения одного и того же объекта. Это проблема кластеризации, одна из наиболее интенсивно изучаемых тем в науке о машинном обучении.

Кластер — это набор схожих сущностей или как минимум набор сущностей, которые похожи друг на друга больше, чем на элементы других кластеров. Делить все на кластеры — в природе человека, и часто это первый шаг на пути к знанию. Даже глядя в ночное небо, мы невольно видим скопления звезд, а потом придумываем красивые названия формам, которые они напоминают. Наблюдение, что определенные группы веществ имеют очень схожие химические свойства, стало первым шагом к открытию периодической системы элементов: каждая группа в ней заняла свой столбец. Все, что мы воспринимаем — от лиц друзей до звуков речи, — это кластеры. Без них мы бы потерялись. Дети не научатся говорить, пока не приобретут навык определять характерные звуки, из которых состоит речь. Это происходит в первые годы жизни, и все слова, которые они потом узнают, не значат ничего без кластеров реальных вещей, к которым эти слова относятся. Сталкиваясь с большими данными — очень большим количеством объектов, — мы вначале группируем их в удобное число кластеров. Рынок в целом — слишком

общий, отдельные клиенты — слишком мелкие, поэтому маркетологи делят его на сегменты, как они называют кластеры. Даже объекты как таковые, по сути, кластеры наблюдений за ними, начиная с маминого лица под разными углами освещения и заканчивая различными звуковыми волнами, которые ребенок слышит как слово «мама». Думать без объектов невозможно, и, наверное, именно поэтому квантовая механика такая неинтуитивная наука: субатомный мир хочется нарисовать в воображении в виде сталкивающихся частиц или интерферирующих волн, но на самом деле это ни то ни другое.

Кластер можно представить по его элементу-прототипу: образу мамы, который сразу приходит на ум, типичной кошки, спортивного автомобиля, загородного дома и тропического пляжа. Для маркетолога Пеория в штате Иллинойс это средний американский городок. Самый обычный гражданин США — это Боб Бернс, пятидесятитрехлетний завхоз из Уиндема в штате Коннектикут, по крайней мере, если верить книге Кевина О'Кифа *The Average American*. По всем числовым атрибутам — например, росту, весу, объему талии и обуви, длине волос и так далее — можно легко найти среднего члена кластера: его рост — это средний рост всех остальных, вес — средний вес и так далее. Для описательных атрибутов, например пола, цвета волос, почтового индекса и любимого вида спорта, «средним» значением будет просто самое распространенное. Средние члены кластеров, описанные таким набором атрибутов, могут существовать или не существовать в реальности, но это в любом случае удачные точки для ориентации: когда планируешь маркетинг нового продукта, удобнее представить себе Пеорию как место введения его на рынок и Боба Бернса как целевого клиента, а не оперировать абстрактными сущностями вроде «рынка» или «клиента».

Такие усредненные объекты, конечно, полезны, но можно поступить еще лучше: вообще весь смысл больших данных и машинного обучения как раз в том, чтобы избежать грубых рассуждений. Кластеры могут быть очень специализированными группами людей или даже различными аспектами жизни одного и того же человека: Элис, покупающая книги для работы, для отдыха или в подарок на Рождество, Элис в хорошем настроении и грустящая Элис. Amazon заинтересована в том, чтобы отделять книги, которые Элис покупает себе, от тех, которые она покупает для своего молодого человека, потому что тогда можно будет дать ей в нужное время подходящую рекомендацию. К сожалению, покупки не сопровождаются ярлыками «подарок для себя» и «для Боба», поэтому Amazon приходится самой разбираться, как их группировать.

Представьте, что объекты в мире Робби делятся на пять кластеров (люди, мебель, игрушки, еда и животные), но неизвестно, какие вещи к каким кластерам относятся. С проблемой этого типа Робби столкнется, как только мы его включим. Простой вариант сортировки объектов по кластерам — взять пять произвольных предметов, сделать их прототипами кластеров, а затем сравнить новые сущности с каждым прототипом, относя их к кластеру самого схожего. (Как и в аналогическом обучении, здесь важно выбрать меру сходства. Если атрибуты числовые, это может быть просто евклидово расстояние, но это далеко не единственный вариант.)

Теперь прототипы надо обновить. Подразумевается, что прототип кластера должен быть средним его членов: когда кластеры состояли из одного члена, все так и было, но теперь мы добавили к ним новые элементы, и ситуация изменилась. Поэтому мы вычислим средние свойства членов для каждого кластера и сделаем полученный результат новым прототипом. Теперь нужно снова обновить принадлежность объектов кластерам: поскольку прототипы изменились, мог измениться и прототип, наиболее близкий данному объекту. Давайте представим, что прототип одной категории — это мишка, а другой — банан. Если взять крекер в виде животного, при первом подходе он может попасть в группу с медведем, а при втором — с бананом. Изначально крекер выглядел как игрушка, но теперь он будет отнесен к еде. Если переместить крекер в одну группу с бананом, прототип для этой группы тоже может измениться: это уже будет не банан, а печенье. Этот полезный цикл, который относит объекты ко все более и более подходящим кластерам, станет продолжаться, пока кластеры сущностей (а с ними и прототипы кластеров) не прекратят меняться.

Такой алгоритм называется метод k -средних, и появился он еще в 50-е годы XX века. Он простой, красивый, при этом довольно популярный, но имеет ряд недостатков, одни из которых устранить легче, а другие — сложнее. Во-первых, количество кластеров надо зафиксировать заранее, а в реальном мире Робби постоянно натывается на новые виды предметов. Один вариант решения — позволить открывать новый кластер, если объект слишком отличается от имеющихся. Другой — разрешить кластерам делиться и сливаться в процессе работы. Так или иначе, вероятно, будет целесообразно включить в алгоритм приоритеты для меньшего количества кластеров, чтобы избежать ситуации, когда у каждого предмета будет собственный кластер (идеальное решение, если кластеры должны содержать схожие предметы, но смысл явно не в этом).

Более серьезная проблема заключается в том, что метод k -средних работает, только когда кластеры легко различимы: они, как пузыри в гиперпространстве, плавают далеко друг от друга, и у всех схожий объем и схожее количество членов. Если какое-то условие не выполнено, начинаются неприятности: вытянутые кластеры делятся надвое, маленькие поглощаются более крупными соседями и так далее. К счастью, можно поступить лучше.

Допустим, мы пришли к выводу, что разрешить Робби слоняться по реальному миру — слишком медленный и громоздкий способ обучения, и вместо этого посадили его смотреть сгенерированные компьютером изображения, как будущего летчика в авиационном тренажере. Мы знаем, из каких кластеров взяты картинки, но не скажем об этом Робби, а будем создавать их, случайно выбирая кластер (скажем, «игрушки»), а потом синтезируя пример этого кластера (маленький пухлый бурый плюшевый медведь с большими черными глазами, круглыми ушами и галстуком-бабочкой). Кроме того, мы будем произвольно выбирать свойства примера: размер мишки — в среднем 25 сантиметров, мех с вероятностью 80 процентов бурый, иначе — белый и так далее. После того как Робби увидит очень много сгенерированных таким образом картинок, он должен научиться делить их на кластеры «люди», «мебель», «игрушки» и так далее, потому что люди, например, больше похожи на людей, а не на мебель. Возникает интересный вопрос: какой алгоритм кластеризации лучше с точки зрения Робби? Ответ будет неожиданным: наивный байесовский алгоритм — первый алгоритм для обучения с учителем, с которым мы познакомились. Разница в том, что теперь Робби не знает классов и ему придется их угадать!

Очевидно: если бы Робби их знал, все пошло бы отлично — как в наивном байесовском алгоритме, каждый кластер определялся бы своей вероятностью (17 процентов сгенерированных объектов — игрушки) и распределением вероятности каждого атрибута среди членов кластера (например, 80 процентов игрушек коричневые). Робби мог бы оценивать вероятности путем простого подсчета числа игрушек в имеющихся данных, количества коричневых игрушек и так далее, но для этого надо знать, какие предметы — игрушки. Эта проблема может показаться крепким орешком, но, оказывается, мы уже знаем, как ее решить. Если бы в распоряжении Робби имелся наивный байесовский классификатор и ему необходимо было определить класс нового предмета, нужно было бы только применить классификатор и вычислить вероятность класса при данных атрибутах объекта. Маленький, пухлый,

коричневый, похож на медведя, с большими глазами и галстуком-бабочкой? Вероятно, игрушка, но, возможно, животное.

Итак, Робби сталкивается с проблемой «курица или яйцо»: зная классы предметов, он мог бы получить модели классов путем подсчета, а если бы знал модели, мог бы сделать заключение о классах объектов. Если вы думаете, что опять застряли, это далеко не так: чтобы стартовать, надо просто начать угадывать классы для каждого предмета каким угодно способом, даже произвольно. На основе этих классов и данных можно получить модели классов, на основе этих моделей — вновь сделать вывод о классах и так далее. На первый взгляд это кажется безумием: придется бесконечно кружиться между выводами о классах на основе моделей и моделей на основе их классов, и даже если это закончится, нет причин полагать, что кластеры получатся осмысленные. Но в 1977 году трое статистиков из Гарварда — Артур Демпстер, Нэн Лэрд и Дональд Рубин — показали, что сумасшедший план работает: после каждого прохождения по этой петле модель кластера улучшается, а после достижения моделью локального максимума схожести повторения заканчиваются. Они назвали эту схему *EM*-алгоритмом, где *E* — ожидания (expectation, заключение об ожидаемых вероятностях), а *M* — максимизация (maximization, оценка параметров максимальной схожести). Еще они показали, что многие предыдущие алгоритмы были частными случаями *EM*. Например, чтобы получить скрытые модели Маркова, мы чередуем выводы о скрытых состояниях с оценкой вероятностей перехода и наблюдения на их основе. Когда мы хотим получить статистическую модель, но нам не хватает какой-то ключевой информации (например, классов примеров), всегда можно использовать *EM*-алгоритм, что делает его одним из самых популярных инструментов в области машинного обучения.

Вы, возможно, заметили определенное сходство между методом *k*-средних и *EM*-алгоритмом, поскольку оба чередуют отнесение сущностей к кластерам и обновление описаний кластеров. Это не случайность: метод *k*-средних сам по себе — частный случай *EM*-алгоритма, который получается, если у всех атрибутов «узкое» нормальное распределение, то есть нормальное распределение с очень маленькой дисперсией. Если кластеры часто перекрываются, объект может относиться, скажем, к кластеру *A* с вероятностью 0,7 и к кластеру *B* с вероятностью 0,3, и нельзя просто отнести его к кластеру *A* без потери информации. *EM*-алгоритм учитывает это путем частичного приписывания объекта к двум кластерам и соответствующего обновления описаний этих кластеров, однако, если распределения очень

сконцентрированы, вероятность, что сущность принадлежит к ближайшему кластеру, всегда будет приблизительно равна единице, и нужно только распределить объекты по кластерам и усреднить их в каждом кластере, чтобы вычислить среднее — то есть получится алгоритм k -среднего.

До сих пор мы рассматривали получение всего одного уровня кластеров, но мир, конечно, намного богаче, и одни кластеры в нем вложены в другие вплоть до конкретных предметов: живое делится на растения и животных, животные — на млекопитающих, птиц, рыб и так далее до домашнего любимца — пса Фидо. Но проблем это не создает: получив набор кластеров, к ним можно относиться как к объектам и, в свою очередь, объединять их в кластеры все более высокого уровня, вплоть до кластера всех объектов. Или же можно начать с грубой кластеризации, а затем все больше дробить кластеры на подкластеры: игрушки Робби делятся на мягкие игрушки, конструкторы и так далее. Мягкие игрушки — на плюшевых медведей, котят и так далее. Дети, видимо, начинают изучение мира где-то посередине, а потом идут и вверх, и вниз. Например, понятие «собака» они усваивают до того, как узнают о «животных» и «гончих». Для Робби это может стать хорошей стратегией.

Открытие формы данных

Независимо от того, поступают ли данные в мозг Робби из его органов чувств или в виде потока миллионов кликов клиентов Amazon, сгруппировать множество в меньшее число кластеров — лишь половина дела. Второй этап — сократить описание объектов. Первый образ мамы, который видит Робби, будет состоять, может быть, из миллиона пикселей, каждый своего цвета, однако человеку вряд ли нужен миллион переменных, чтобы описать лицо. Аналогично каждый товар, на который вы кликнули на сайте Amazon, дает частицу информации о вас, но на самом деле Amazon интересны не ваши клики, а ваши вкусы. Вкусы довольно стабильны и в какой-то мере подразумеваются в кликах, количество которых растет безгранично во время пользования сайтом и должно понемногу складываться в картину предпочтений точно так же, как пиксели складываются в картинку лица. Вопрос в том, как реализовать это сложение.

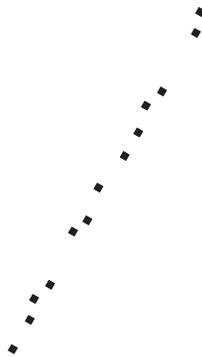
У человека есть примерно 50 лицевых мышц, поэтому 50 чисел должно с лихвой хватить для описания всех возможных выражений лица. Форма глаз, носа, рта и так далее — всего того, что помогает отличить одного человека от другого, — тоже не должна занимать больше нескольких десятков чисел.

В конце концов, художникам в полиции достаточно всего десяти вариантов каждой черты лица, чтобы составить фоторобот, позволяющий опознать подозреваемого. Можно добавить еще несколько чисел для описания освещения и наклона, но на этом все. Поэтому, если вы дадите мне примерно сотню чисел, этого должно хватить для воссоздания лица, и наоборот: мозг Робби должен быть способен взять картинку лица и быстро свести ее ко все той же сотне по-настоящему важных чисел.

Специалисты по машинному обучению называют этот процесс понижением размерности, потому что он уменьшает множество видимых измерений (пикселей) до нескольких подразумеваемых (выражение и черты лица). Понижение размерности важно для того, чтобы справиться с большим объемом данных, например данными, поступающими каждую секунду из органов чувств. Может быть, действительно лучше один раз увидеть, чем сто раз услышать, но обрабатывать и запоминать изображения в миллион раз сложнее, чем слова. Тем не менее зрительная кора головного мозга каким-то образом довольно хорошо справляется с уменьшением такого объема информации до приемлемого, достаточного, чтобы ориентироваться в мире, узнавать людей и предметы и помнить увиденное. Это великое чудо познания настолько естественно для нас, что мы его даже не замечаем.

Наводя порядок в своей библиотеке, вы тоже выполняете своего рода понижение размерности от обширного пространства тем до одномерной полки. Некоторые тесно связанные книги неизбежно окажутся далеко друг от друга, но все равно можно расставить их так, чтобы такие случаи были редкими. Алгоритм понижения размерности делает именно это.

Представьте, что я дал вам координаты GPS всех магазинов в Пало-Альто в Калифорнии и вы нанесли их на листок бумаги:



Наверное, взглянув на эту схему, вы сразу поймете, что главная улица городка ведет с юго-запада на северо-восток. Хотя вы не рисовали саму улицу, интуиция подсказывает, где она проходит, потому что все точки лежат на прямой линии (или рядом с ней — магазины могут быть по разные стороны улицы). Догадка верна: эта улица — Университи-авеню, и, если вы окажетесь в Пало-Альто и захотите перекусить и сделать покупки, туда и надо идти. Еще лучше, что, когда магазины сконцентрированы на одной улице, для описания их расположения нужно уже не два числа, а всего одно — номер дома, а для большей точности — расстояние от магазина до пригородной железнодорожной станции в юго-западном углу, откуда начинается Университи-авеню.

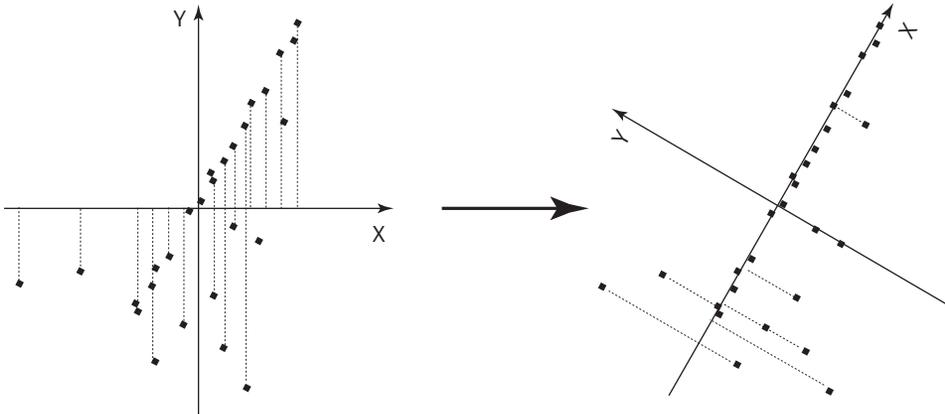
Если нанести на карту еще больше магазинов, вы, вероятно, заметите, что часть из них находится на перекрестках, чуть в стороне от Университи-авеню, а некоторые — вообще в других местах:



Тем не менее большинство магазинов все равно расположены довольно близко к центральной улице, и, если разрешено использовать для описания положения магазина только одно число, расстояние от вокзала вдоль этой улицы будет довольно удачным вариантом: пройдя этот отрезок и оглядевшись, вы с достаточной вероятностью найдете нужный магазин. Итак, вы только что понизили размерность «расположения магазинов в Пало-Альто» с двух измерений до одного.

У Робби, однако, нет преимуществ, которые дает человеку сильно развитая зрительная система, поэтому, если вы попросите его забрать белье из химчистки Elite Cleaners и учтете на его карте только одну координату,

ему нужен будет алгоритм, чтобы «открыть» Университи-авеню на основе GPS-координат магазинов. Ключ к решению проблемы — заметить, что, если поставить начало координат плоскости x, y в усредненное расположение магазинов и медленно поворачивать оси, магазины окажутся ближе всего к оси x при повороте примерно на 60 градусов, то есть когда ось совпадает с Университи-авеню:



Это направление — так называемая первая главная компонента данных — будет направлением, вдоль которого разброс данных наибольший. (Обратите внимание: если спроецировать магазины на ось x , на правом рисунке они будут находиться дальше друг от друга, чем на левом.) Обнаружив первую главную компоненту, можно поискать вторую, которой в данном случае станет направление наибольшей дисперсии под прямым углом к Университи-авеню. На карте остается только одно возможное направление (направление перекрестков). Но если бы Пало-Альто находился на склоне холма, одна или две главные компоненты частично были бы расположены непосредственно на холме, а третья — последняя — оказалась бы направлена в воздух. Ту же идею можно применить к тысячам и миллионам измерений данных, как в случае изображений лиц: нужно последовательно искать направления наибольшей дисперсии, пока оставшаяся вариабельность не окажется наименьшей. Например, после поворота осей на рисунке выше координата y большинства магазинов будет равна нулю, поэтому среднее y окажется очень маленьким, и, если его вообще проигнорировать, потеря информации получится незначительной. А если мы все же решим сохранить

у, то z (направленная вверх) наверняка будет несущественна. Как оказалось, линейная алгебра позволяет провести процесс поиска главных компонент всего за один цикл, но еще лучше то, что даже в данных с очень большим количеством измерений значительную часть дисперсии зачастую дают всего несколько измерений. Если это не так, все равно визуальный поиск двух-трех важнейших измерений часто оказывается очень успешным, потому что наша зрительная система дает удивительные возможности восприятия.

Метод главных компонент (Principal Component Analysis, PCA), как называют этот процесс, — один из важнейших инструментов в арсенале ученого. Можно сказать, что для обучения без учителя это то же самое, что линейная регрессия для контролируемого множества. Знаменитая «клюшкообразная» кривая глобального потепления, например, была получена в результате нахождения главной компоненты различных рядов данных, связанных с температурой (годовые кольца деревьев, ледяные керны и так далее), и допущения, что это запись температуры как таковой. Биологи используют метод главных компонент, чтобы свести уровни экспрессии тысяч различных генов в несколько путей. Психологи обнаружили, что личность можно выразить пятью факторами — это экстраверсия, доброжелательность, добросовестность, нейротизм и открытость опыту, — которые оценивают по твитам и постам в блогах. (У шимпанзе, предположительно, есть еще одно измерение — реактивность, — но их с помощью Twitter не оценишь.) Применение метода главных компонент к голосам на выборах в Конгресс и данным избирателей показывает, что, вопреки расхожему мнению, политика в основном не сводится к противостоянию либералов и консерваторов. Люди отличаются в двух основных измерениях — экономических и социальных вопросах, — и, если спроецировать их на одну ось, либертарианцы смешаются с популистами, хотя их позиции полярно противоположны, и возникнет иллюзия, что в центре много умеренных. Попытка апеллировать к ним вряд ли окажется выигрышной стратегией. С другой стороны, если либералы и либертарианцы преодолеют взаимную неприязнь, они могут стать союзниками в социальных вопросах, где и те и другие выступают за свободу личности.

Когда Робби подрастет, он сможет применять один из вариантов метода главных компонент для решения проблемы «эффекта вечеринки», то есть чтобы выделить из шума толпы отдельные голоса. Схожий метод может помочь ему научиться читать. Если каждое слово — измерение, тогда текст — точка в пространстве слов, и главные направления этого пространства

окажутся элементами значения. Например, «президент Обама» и «Белый дом» в пространстве слов далеко отстоят друг от друга, но в пространстве значений близки, потому что обычно появляются в схожих контекстах. Хотите верьте, хотите нет, но такой тип анализа — все, что требуется и компьютерам, и людям для оценки сочинений на экзаменах SAT (стандартизованный тест для приема в высшие учебные заведения США). В Netflix используется похожая идея. Вместо того чтобы рекомендовать фильмы, которые понравились пользователям со схожими вкусами, система проецирует и пользователей, и фильмы в «пространство вкуса» с низкой размерностью и рекомендует картины, расположенные в этом пространстве рядом с вами. Это помогает найти фильмы, которые вы никогда не видели, но обязательно полюбите.

Тем не менее главные компоненты набора данных о лице вас, скорее всего, разочаруют. Вопреки ожиданиям, это будут, например, не черты лица и выражения, а скорее размытые до неузнаваемости лица призраков. Дело в том, что метод главных компонент — линейный алгоритм, поэтому главными компонентами могут быть только взвешенные пиксель за пикселем средние реальных лиц (их еще называют «собственные лица», потому что они собственные векторы центрированной ковариационной матрицы этих данных, но я отхожу от темы). Чтобы по-настоящему понять не только лица, но и большинство форм в мире, нам понадобится кое-что еще, а именно нелинейное понижение размерности.

Представьте, что вместо карты Пало-Альто у нас есть GPS-координаты важнейших городов всей Области залива Сан-Франциско:



Глядя на эту схему, можно, вероятно, сделать предположение, что города расположены на берегу залива и, если провести через них линию, положение каждого города можно определить всего одним числом: как далеко он находится от Сан-Франциско по этой линии. Но метод главных компонент такую кривую найти не может: он нарисует прямую линию через центр залива, где городов вообще нет, а это только затуманит, а не прояснит форму данных.

Теперь представьте на секунду, что мы собираемся создать Область залива с нуля. Бюджет позволяет построить единую дорогу, чтобы соединить все города, и нам решать, как она будет проложена. Естественно, мы проложим дорогу, ведущую из Сан-Франциско в Сан-Бруно, оттуда в Сан-Матео и так далее, вплоть до Окленда. Такая дорога будет довольно хорошим одномерным представлением Области залива, и ее можно найти простым алгоритмом: «построй дорогу между каждой парой близлежащих городов». Конечно, у нас получится целая сеть дорог, а не одна, проходящая рядом с каждым городом, но можно получить и одну дорогу, если сделать ее наилучшим приближением сети, в том смысле, что расстояния между городами по этой дороге будут как можно ближе расстоянию вдоль сети.

Именно это делает Isomap — один из самых популярных алгоритмов нелинейного понижения размерности. Он соединяет каждую точку данных в высокоразмерном пространстве (пусть это будет лицо) со всеми близлежащими точками (очень похожими лицами), вычисляет кратчайшие расстояния между всеми парами точек по получившейся сети и находит меньшее число координат, которое лучше всего приближает эти расстояния. В отличие от метода опорных векторов, координаты лиц в этом пространстве часто довольно осмысленны: одна координата может показывать, в каком направлении смотрит человек (левый профиль, поворот на три четверти, анфас и так далее), другая — как лицо выглядит (очень грустное, немного грустное, спокойное, радостное, очень радостное) и так далее. Isomap имеет неожиданную способность сосредотачиваться на самых важных измерениях комплекса данных — от понимания движения на видео до улавливания эмоций в речи.

Вот интересный эксперимент. Возьмите потоковое видео из глаз Робби, представьте, что каждый кадр — точка в пространстве изображений, а потом сведите этот набор изображений к единственному измерению. Какое это будет измерение? Время. Как библиотекарь расставляет книги на полке, время ставит каждое изображение рядом с самыми похожими. Наверное, наше восприятие времени — просто естественный результат замечательной

способности головного мозга понижать размерность. В дорожной сети нашей памяти время — главная автострада, и мы ее быстро находим. Другими словами, время — главная компонента памяти.

Жизнелюбивый робот

Кластеризация и понижение размерности приближают нас к пониманию человеческого обучения, но чего-то очень важного все равно не хватает. Дети не просто пассивно наблюдают за миром. Они активно действуют: замечают предметы, берут их в руки, играют, бегают, едят, плачут и задают вопросы. Даже самая продвинутая зрительная система будет бесполезна, если она не поможет Робби взаимодействовать со средой. Ему нужно не просто знать, где что находится, но и что надо делать в каждый момент. В принципе можно научить его выполнять пошаговые инструкции, соотнося показания сенсоров с соответствующими ответными действиями, но это возможно только для узких задач. Предпринимаемые действия зависят от цели, а не просто от того, что вы в данный момент воспринимаете, при этом цели могут быть весьма отдаленными. И потом, в любом случае нужно обойтись без пошагового контроля: дети учатся ползать, ходить и бегать сами, без помощи родителей. Ни один рассмотренный нами обучающий алгоритм так учиться не умеет.

У людей действительно есть один постоянный ориентир: эмоции. Мы стремимся к удовольствиям и избегаем боли. Коснувшись горячей плиты, вы непроизвольно отдернете руку. Это просто. Сложнее научиться не трогать плиту: для этого нужно двигаться так, чтобы избежать острой боли, которую вы еще не почувствовали. Головной мозг делает это, ассоциируя боль не просто с моментом прикосновения к плите, но и с ведущими к этому действиями. Эдвард Торндайк¹ назвал это законом эффекта: действия, которые ведут к удовольствию, станут с большей вероятностью повторяться в будущем, а ведущие к боли — с меньшей. Удовольствие как будто путешествует назад во времени, и действия в конце концов могут начать ассоциироваться с довольно отдаленными результатами. Люди освоили такой поиск косвенных наград лучше, чем любое животное, и этот навык критически важен для успеха в жизни. В знаменитом эксперименте детям давали зефир и говорили, что, если они выдержат несколько минут и не съедят его, им дадут целых два.

¹ Эдвард Ли Торндайк (Edward Lee Thorndike, 1874–1949) — американский психолог и педагог. Президент Американской психологической ассоциации (1912–1939).



[Почитать описание, рецензии
и купить на сайте](#)

Лучшие цитаты из книг, бесплатные главы и новинки:

