

04, апрель 2016

УДК 004.8

Использование технологий машинного обучения для создания искусственных нейронных сетей

Слепенков В.О., студент

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Системы обработки информации и управления»*

Бондарь А.В., студент

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Системы обработки информации и управления»*

Научный руководитель: Гапанюк Ю.Е. к.т.н., доцент кафедры

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Системы обработки информации и управления»*

gapuu@bmstu.ru

Машинное обучение, активно развивается с начала XIX века. Апогея развития в этой области происходила в 50-х и 60-х годах. Важный этап произошел в 1958 году, когда Фрэнком Розенблаттом был создан первый в мире нейронный компьютер «Марк-1», основанный на одной из первых моделей искусственных нейронных сетей, перцептроне. Являясь первым представителем искусственных нейронных сетей по своему строению и возможностям, он был необычайно прост, но при этом, перцептрон способен обучаться и решать задачи. Искусственные нейронные сети возникли тогда, когда человечество начало активно заниматься изучением процессов, протекающих в мозге, и при попытке смоделировать эти процессы. **Искусственная нейронная сеть** представляет собой систему соединённых и взаимодействующих между собой простых процессоров, её не программируют в обычном смысле этого слова, она обучается.

Подходя к термину **машинного обучения**, в книге [1] приводится следующее определение: «Машинным обучением называется систематичное обучение алгоритмов и систем, в результате которого их знания или качество работы возрастают по мере накопления опыта». Данная область находится на пересечении многих разделов наук. В задачах обучения часто применяются методы теории вероятностей, линейной алгебры, статистики и оптимизации, а само машинное обучение применяется в широком спектре прикладных сфер:

- распознавание речи;
- компьютерное зрение;
- компьютерная лингвистика и обработка естественных языков;
- медицинская диагностика;
- техническая диагностика;
- финансовые прогнозы;
- информационный поиск;
- интеллектуальные игры.

Существуют разные типы машинного обучения:

1. Обучение с учителем (обучение на размеченных данных), т.е. идёт поиск закономерности на основании различных пар(прецедентов) объект-ответ, исходя из которой можно дать ответ для любого объекта. **Объект** является ситуацией или примером, а **ответ** – это реакция или отклик;
2. Обучение без учителя (обучение на не размеченных данных), т.е. составленный алгоритм ведет поиск связей между объектами, а не ищет сами пары объект-ответ;
3. Обучение с подкреплением (стимулируемое обучение), при котором правильных ответов нет, а алгоритм ведет организацию наиболее выигрышной стратегии;
4. Частичное обучение, часть данных уже известна, т.е. имеется правильная выборка из пар объект-ответ, а часть нет;
5. Активное обучение, при котором алгоритм сам может обучаться на малой выборке данных, которые выбирает сам.

Обучение с учителем и обучение без учителя имеют большую популярность в наше время и широко используются при обучении нейронных сетей, поэтому рассмотрим эти два типа.

Обучение с учителем – это раздел машинного обучения, который ищет решения для задач на поиск и восстановления зависимостей, исходя из которых, уже по новым объектам (входным данным), предсказывать ответы.

Пусть множество X – объекты, примеры, а множество Y – ответы, отклики. Тогда имеется некоторая зависимость, позволяющая по $x \in X$ предсказать (или оценить) $y \in Y$. Если зависимость детерминированная (однозначная взаимосвязь причины и следствия), то существует функция:

$$f(x) : X \rightarrow Y$$

Зависимость известна только на объектах из обучающей выборки:

$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(N)}, y^{(N)})\},$$

где пара $(x^{(i)}, y^{(i)}) \in X \times Y$ – прецедент.

В зависимости от множества Y выделяют разные типы задач обучения.

Задача классификации (задача распознавания образов) – одна из самых распространённых задач машинного обучения, когда X разбивается на K классов:

$$X_k = \{x \in X : f(x) = k\} (k = 1, 2, \dots, K)$$

По x нужно предположить (предсказать), к какому классу он принадлежит. В самом простом варианте есть два класса (рис. 1), положительный (треугольник) и отрицательный (круг), такую классификацию называют бинарной. Её можно легко представить в виде решающего дерева.

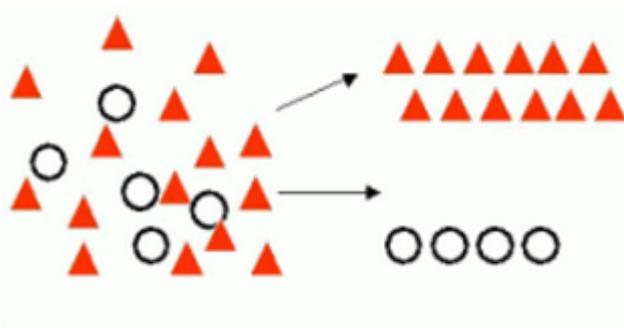


Рис. 1. Классификация

Если обнаруживается в объекте признак 1, тогда он нам подходит, если нет, проверяется наличие другого признака, отвечающего нашим требованиям. Чтобы оценить качество классификации необходимо правильно различать качество классификации каждого класса. В книге [1] даётся такое определение: «Правильно классифицированные положительные и отрицательные объекты называют, соответственно, истинно положительными и истинно отрицательными результатами. Неправильно классифицированные положительные объекты называют ложноположительными, а неправильно классифицированные отрицательные объекты – ложноотрицательными». Тогда получается, ложноположительный означает нечто, что было неправильно определено как положительный и на самом деле являющийся отрицательным. Оценка классификации будет зависеть от доли правильно классифицированных положительных объектов и правильно классифицированных отрицательных объектов. Но что если требуется многоклассовая классификация? Ответ весьма интересный – необходимо обучить бинарные классификаторы, первый из которых будет отделять один класс от

других, второй отделяет следующий от других и т.д. При обучении последнего классификатора мы будем рассматривать все объекты этого класса как положительные, а другие – как отрицательные. Оценка такой классификации может происходить благодаря использованию оценок каждого из классификаторов с последующим обучением модели для многоклассовой оценки, такой метод требует дополнительного обучения, но применим всегда.

Как итог, для решения проблемы с помощью бинарных многоклассовых классификаторов, необходимо разбить проблему на M бинарных проблем обучения, обучить M классификаторов на исходных данных и объединить результаты этих M классификаторов в один результат и воспользоваться методом многоклассовой оценки. Как примеры задач классификации можно привести:

1. Распознавание образов (лиц). Классы – личности;
2. Распознавание рукописного текста. Классы – символы;
3. Определение спама в электронной почте. Классы – спам, не спам.

Восстановление регрессии – это задача, в которой проявляется различие между группирующими и ранжирующими моделями, идея группирующей модели состоит в том, чтобы разумным (логичным) способом разбить всю группу объектов на небольшие группы (сегменты) и в каждой такой группе обучить более простую модель, основываясь на этой группе [1]. Ранжирование же, использует оценку данных на основании выборки, делая список из значений, наиболее подходящих по параметрам. В теории вероятностей и математической статистике, регрессия – это математическое выражение, выражающее зависимость зависимой переменной от независимых переменных при условии, что это математическое выражение будет иметь статистическую значимость. Чтобы решить задачу, необходимо по обучающей выборке построить алгоритм, который аппроксимирует (заменяет близким, приблизительно равным) неизвестную зависимость на данных. Если привести небольшое обособление, то получается так, что задача восстановления регрессии – это задача классификации с множеством классов. Как примеры задач восстановления регрессии можно привести:

1. Кредиты. Оценка максимальной суммы кредита;
2. Продажи. Оценка объёмов продаж.

Задача прогнозирования – это процесс по расчету прогноза или разработки прогнозной модели, т.е. на основе значений, полученных ранее, и их параметров, найти

зависимость, тем самым предсказать будущие значения. Нам дано множество X , являющееся временным рядом (т.е. множество некоторых значений функции во времени). Задача заключается в нахождении значений функции за пределами данных, имеющихся в X . В качестве реальных примеров задачи прогнозирования можно привести:

1. Сейсмические предсказания. Прогнозирование интервала для следующего землетрясения на данной территории исследования;
2. Изменение стоимости. Прогнозирование изменений стоимости продукта в какой-то определенный временной интервал;
3. Нагрузка на центр связи. Прогноз нагрузки через кол-во поступающих звонков на телефонные (и не только) каналы связи.

В качестве примера машинного обучения с учителем можно рассмотреть одну из первых искусственных нейронных сетей – перцептрон. Перцептрон был впервые реализован в 1958 году на первом нейрокомпьютере «Марк-1». Перцептрон способен работать в одном из двух режимов: распознавание или обобщение. Таким образом, перцептрон способен принимать на вход ранее неизвестные ему объекты и определять к какому классу они принадлежат. Прежде чем рассмотреть процесс обучения перцептрона стоит рассмотреть его структуру.

В общем случае, перцептрон состоит из элементов трёх типов: S-элементы, так же называемые рецепторами или сенсорами, являющиеся по факту входными данными, A-элементы, называемые ассоциативными, и R-элементы, то есть выходные элементы (рис. 2).

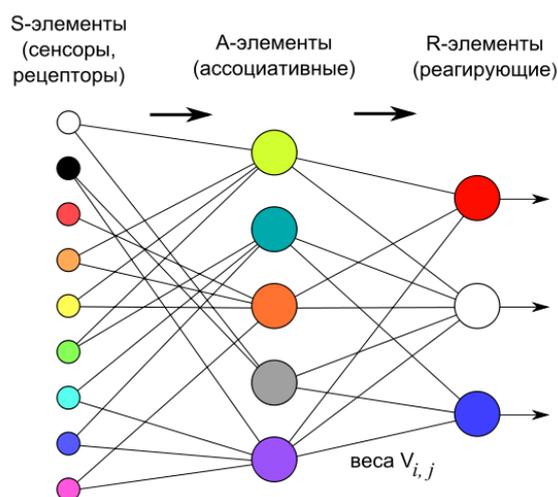


Рис. 2. Перцептрон

Принцип работы перцептрона: каждому A элементу соответствует некоторое множество S элементов, и в зависимости от входных данных возбуждаются определенные

S элементы. При превышении некоторого количества возбуждённых S - элементов, связанных с рассматриваемым A - элементом, данный A элемент активизируется. После этого, сигналы от активизированных A элементов передаются на следующий слой A - элементов, в случае многослойной структуры, или на выходные элементы R, в случае однослойной структуры. При этом, сигнал посылается с определенным коэффициентом, заданным отдельно для каждой связи. На выходном элементе все сигналы, помноженные на коэффициенты связей, так же именуемые весами, суммируются, и если сумма превышает некоторый порог θ , то на выход перцептрона выдаётся единица, то есть объект принадлежит к классу, иначе перцептрон причисляет объект ко второму классу и выдаёт на выходе - 1. Функцию, выполняемую, R - элементом можно записать математически:

$$f(x) = \text{sign}\left(\sum_{i=1}^n \omega_i x_i - \theta\right).$$

Простейший перцептрон, описанный, Фрэнком Розенблаттом, имеет один скрытый слой (то есть один слой A - элементов), способен выполнять задачи распознавания и обобщения. Для работы перцептрона ему требуется обучение, и, хотя, для перцептрона возможен так же и вариант с обучением без учителя, мы рассмотрим классический вариант.

Обучением перцептрона является процесс настройки весов и пороговых значений. Классическим для перцептрона является метод коррекции ошибок, чей основной принцип заключается в следующем: на вход системе подаются объекты, и веса не меняются до тех пор, пока перцептрон правильно определяет класс объекта. При неправильной реакции на объект, мы меняем вес на единицу, а направление изменения противоположно знаку ошибки. То есть если мы ожидали от перцептрона на выходе +1, а получили -1, то требуется изменить вес всех связей, соответствующих возбуждённым элементам, на +1. Таким образом, алгоритм выглядит так:

1. Случайным образом выбираем пороговые значения и устанавливаем связи между S и A элементами;
2. Начальные веса полагаем равными нулю;
3. Предъявляем на вход обучающую выборку, то есть объекты с указанием класса к которому они должны относиться (то есть первый или второй класс);
 - а. При предъявлении объекта первого класса, некоторое количество A элементов возбуждётся. Веса, соответствующие этим возбуждённым элементам, изменяем на +1;

- в. При предъявлении объекта второго класса, изменяем веса, соответствующие возбужденным элементам на -1.
4. Таким образом, предъявив всю обучающую выборку, сформируем веса.

Обучение без учителя – это обучение выявляет по-настоящему новые знания, поэтому оно зачастую находится на пересечении машинного обучения и добычи данных. Это такой способ решения задач, при котором система обучается выполнять поставленную задачу, без вмешательства эксперта. Такое обучение пользуется большой популярностью при обучении искусственных нейронных сетей. Главная черта, делающая обучение без учителя привлекательным, – это его "самостоятельность".

Кластеризация. Пусть дано некоторое множество объектов X . Задача заключается в построении алгоритма, способного относить произвольный объект $x \in X$ к некоторому кластеру. Кластером будем называть класс в той же задаче классификации, но с такой разницей, что кластеры изначально неизвестны. Кластеры, в обычных обстоятельствах, обладают следующими свойствами:

1. Непересекаемость (Никакие два кластера не имеют общих элементов);
2. Компактность (под компактностью достаточно понимать, как конечность каждого кластера);
3. Связность (никакой кластер невозможно разбить на два непустых непересекающихся открытых подмножества).

Некоторые алгоритмы кластеризации строят прогностическую модель, т.е. на основе обучающих данных они порождают модель кластеризации, которую можно использовать для отнесения новых данных к кластерам. По существу, ставится задача обучить подходящую модель кластеризации для имеющихся данных. Хорошую кластеризацию отличает от плохой тот факт, что данные разбиваются на компактные группы, или кластеры (рис. 3).

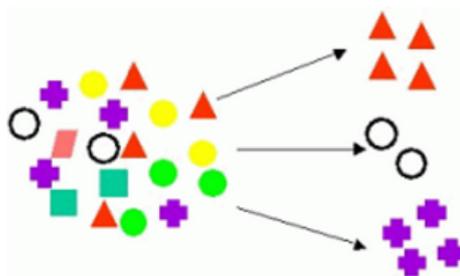


Рис. 3. Пример кластеризации

Под компактностью подразумевается, что в среднем два объекта из одного кластера имеют между собой больше общего, чем два объекта из разных кластеров. Предполагается что существует какой-то способ оценить схожесть или расхождение произвольной пары. Таким образом, хорошим считается кластеризатор, в котором сумма расхождений по всем кластерам (внутрикластерный разброс) намного меньше расхождений всего набора данных. Основная проблема состоит в том, что очень трудно (почти невозможно) найти глобальное решение задачи для большого набора данных.

Выявление ассоциативных правил. Дано некоторое множество объектов X . Задача заключается в построении алгоритма, способного находить взаимосвязи между элементами X . Часто встреченные наборы данных легко можно найти перебором с помощью алгоритма поиска в ширину, если разложить объекты на параметры и сделать решётку для наборов параметров (рис. 4), где связи – это отношение «является подмножеством». В овалах с пунктирной границей встречаются в одном объекте, со штриховой границей – в двух объектах, в треугольниках – в трёх объектах, в n -угольниках – в n объектах. Максимальные наборы параметров с опорой 3 и больше в фигурах с зеленой границей.

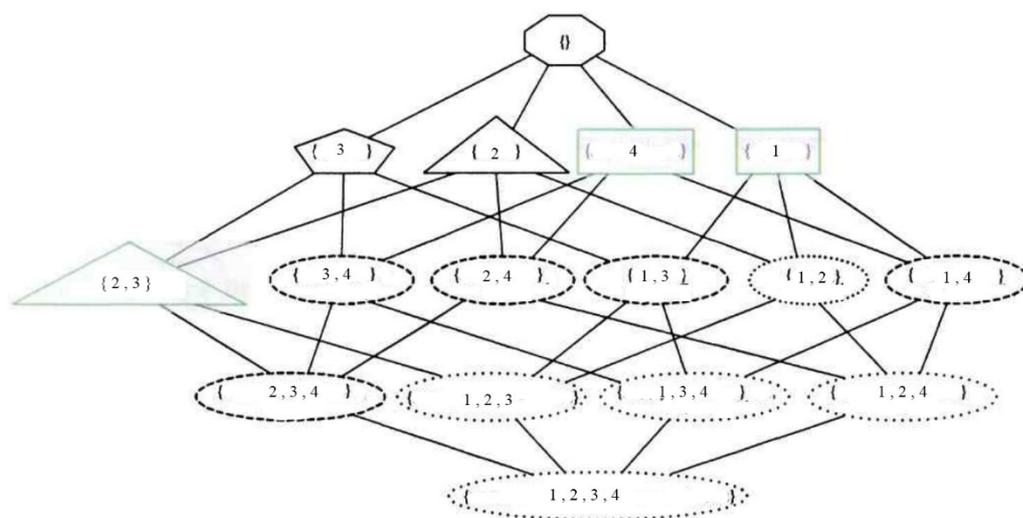


Рис. 4. Решётка для набора (адаптировано из [1])

Возьмём ребро, соединяющее наборы $\{2\}$ и $\{2,4\}$, опора первого равна 3, а второго – 2, т.е. параметр 2 имеется в трёх объектах, а 2 и 3 в двух, в каждом из которых есть параметр 2. Получается, что уровень доверия правила «if 2 then 3» равен $2/3$. Далее происходит отброс данных с меньшим уровнем доверия, лишние правила, частные случаи, уровень доверия которых не выше, чем у общего.

В качестве реальных примеров задачи поиска ассоциативных правил можно привести:

1. Анализ поведения пользователя;
2. Анализ корзины покупателя;
3. Анализ возможных операций пользователя;
4. Анализ пути передвижения.

Сокращение размерности заключается в том, чтобы по уже известным (данным) признакам используя некоторые функции преобразования прийти к наименьшему числу новых признаков, не потеряв при этом важной информации об объектах выборки. Сокращение размерности может потребоваться, когда данные избыточны в информационном плане, т.е. задачу можно решить с тем же уровнем эффективности и точности, но используя меньший объем данных. Такое действие позволяет урезать время и вычислительные затраты на решение задачи. Сокращение объема должно происходить за счет данных, которые имеют наименьшую ценность, например, сначала за счет наименее значимых признаков, затем – похожих записей и т.д., пока размерность не окажется достаточной точки зрения объема вычислений. Решается данная задача с помощью:

- Метода главных компонент;
- Метода независимых компонент;
- Многомерного шкалирования.

В рамках данной статьи рассматривать эти решения мы не будем, а перейдем к обзору искусственных нейронных сетей и машинного обучения которое задействовано в них.

Для демонстрации машинного обучения без учителя рассмотрим нейронную сеть Когнитрон, одну из первых самоорганизующихся сетей, созданную в 1975 году Кунихикой Фукусимой.

Когнитрон – нейронная сеть, симулирующая строение зрительной коры, главной целью которой является распознавание объектов. Когнитрон работает на принципах самоорганизации, соответственно, он способен обучаться без учителя и разделять образы на классы самостоятельно. Рассмотрим структуру Когнитрона.

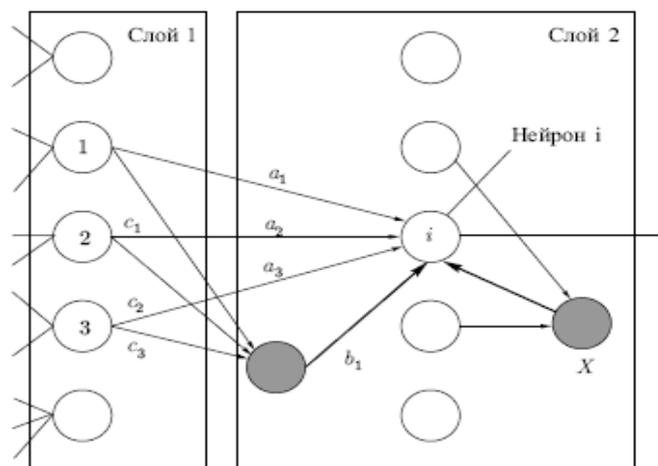


Рис. 5. Когнитрон (адаптировано из [2])

Когнитрон состоит из иерархически связанных слоев нейронов двух типов — тормозящих (на рис. 5 – тёмные нейроны) и возбуждающих (на рис. 5 – светлые нейроны). Состояние возбуждения каждого нейрона определяется суммой его тормозящих и возбуждающих входов. Синаптические связи идут от нейронов одного слоя (далее слоя 1) к следующему (слою 2). Относительно данной синаптической связи соответствующий нейрон слоя 1 является пресинаптическим, а нейрон второго слоя — постсинаптическим. Постсинаптические нейроны связаны не со всеми нейронами 1-го слоя, а лишь с теми, которые принадлежат их локальной **области связей**. Области связей близких друг к другу постсинаптических нейронов перекрываются, поэтому активность данного пресинаптического нейрона будет сказываться на все более расширяющейся области постсинаптических нейронов следующих слоев иерархии.

Суммарное воздействие на постсинаптический нейрон определяется как отношение суммы его возбуждающих ходов (E) и суммы его тормозящих входов (I), которые можно найти по следующим формулам:

$$E = \sum_j a_j u_j, \quad I = \sum_j b_j u_j.$$

Соответственно, суммарное воздействие на нейрон находится по формуле:

$$NET_i = ((1 + E)/(1 + I)) - 1.$$

Выходная активность нейрона устанавливается равной NET_i , если $NET_i > 0$, в противном случае, выходная активность устанавливается равной нулю.

Как уже было сказано выше, Когнитрон способен обучаться без учителя, то есть является самоорганизующейся сетью. Получив обучающий набор данных, сеть самоорганизуется посредством изменения весов связей. Так как в процессе обучения

Когнитрона вес связей только увеличивается, можно предположить проблему произвольного разрастания выходного сигнала, но его удаётся избежать благодаря формуле вычисления выходного сигнала приведённой выше.

Одной из главных проблем обучения Когнитрона является его многослойность, то есть при необходимости внести малейшие изменения в один из слоёв, требуется полностью переучивать другой слой.

Модель Когнитрона является весьма простой концепцией, во многом ставшей основополагающей в исследованиях нейросетей.

Заключение

Искусственные нейронные сети и машинное обучение приближают нас к полному пониманию процессов человеческого мозга и, что более важно, к созданию его аналога – искусственному интеллекту. Машинное обучение включает в себя множество разных методик, многие из которых не связаны с искусственными нейронными сетями, и, пока, мы не можем предположить, какое из направлений окажется наиболее перспективным. Тем не менее, уже сейчас машинное обучение позволяет решать большое разнообразие задач, и с дальнейшим его развитием количество таких задач будет только увеличиваться, позволяя автоматизировать многие задачи человечества, в которых сегодня человек выглядит незаменимым.

Список литературы

- [1] Флах П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных / пер. с англ. А.А. Слинкин. М.: ДМК Пресс, 2015. 400с.
- [2] Национальный открытый университет ИНТУИТ. Основы теории нейронных сетей. Лекция 13: Когнитрон. Режим доступа: <http://www.intuit.ru/studies/courses/88/88/lecture/20551> (дата обращения 27.11.15).
- [3] Федоренко Ю. С. Анализ данных на основе алгоритмов класса SOINN // Молодежный научно-технический вестник. МГТУ им. Н. Э. Баумана. Электрон. журн. 2015. № 4. Режим доступа: <http://sntbul.bmstu.ru/doc/777163.html> (дата обращения 28.11.15).
- [4] Розенблатт Ф. Принципы нейродинамики: Перцептроны и теория механизмов мозга. М.: Мир, 1965. 480с.