Курс лекций «Введение в ИИ.». Часть III. Нейрокомпьтинг. Лекция 18. Стохастические методы

О.Г. Чанышев

Содержание

1 Введение **1**

2 Использование обучения **1**

2.1 Больцмановское обучение 3

2.2 Обучение Коши 4

2.3 Метод искусственной теплоемкости 5

1 Введение

Стохастические методы полезны как для обучения искусственных нейронных сетей, так и для получения выхода от уже обученной сети. Стохастические методы обучения полезны, поскольку позволяют исключать локальные минимумы в процессе обучения. Но с ними также связан ряд проблем. Использование стохастических методов для получения выхода от уже обученной сети рассматривается в работе [2] и обсуждается в следующей лекции. Данная лекция посвящена методам обучения сети.

2 Использование обучения

Имеется два класса обучающих методов: детерминистский и стохастический.

Детерминистский метод обучения шаг за шагом осуществляет процедуру коррекции весов сети, основанную на использовании их текущих значений, фактических и желаемых выходов. Обучение персептрона является примером подобного подхода.

Стохастические методы обучения выполняют псевдослучайные изменения величин весов, сохраняя те изменения, которые ведут к улучшениям. Чтобы увидеть, как это может быть сделано, рассмотрим рис. 1, на котором изображена типичная сеть, в которой нейроны соединены с помощью весов. Выход нейрона является здесь взвешенной суммой его входов, которая, преобразована с помощью нелинейной функции. Для обучения сети может быть использована следующая процедура:

1. Выбрать вес случайным образом и подкорректировать его на небольшое случайное Предъявить множество входов и вычислить получающиеся выходы.

Рис. 1: Двухслойная сеть без обратных связей

Рис. 2: Проблема локальных минимумов.

2. Сравнить эти выходы с желаемыми выходами и вычислить величину разности между ними. Общепринятый метод состоит в нахождении разности между фактическим и желаемым выходами для каждого элемента обучаемой пары, возведение разностей в квадрат и нахождение суммы этих квадратов. Целью обучения является минимизация этой разности, часто называемой целевой функцией.

3. Выбрать вес случайным образом и подкорректировать его на небольшое случайное значение. Если коррекция помогает (уменьшает целевую функцию), то сохранить ее, в противном случае вернуться к первоначальному значению веса.

4. Повторять шаги с 1 до 3 до тех пор, пока сеть не будет обучена в достаточной степени.

Этот процесс стремится минимизировать целевую функцию, но может попасть, как в ловушку, в неудачное решение. На рис. 2 показано, как это может иметь место в системе с единственным весом. Допустим, что первоначально вес взят равным значению в точке A. Если случайные шаги по весу малы, то любые отклонения от точки A увеличивают целевую функцию и будут отвергнуты. Лучшее значение веса, принимаемое в точке B, никогда не будет найдено, и система будет поймана в ловушку локальным минимумом, вместо глобального минимума в точке B. Если же случайные коррекции веса очень велики, то как точка A, так и точка B будут часто посещаться, но то же самое будет иметь место и для каждой другой точки. Вес будет меняться так резко, что он никогда не установится в желаемом минимуме.

Для избежания подобных проблем следует делать большие начальные шаги и постепенно уменьшать их средний случайный размер. Это позволяет сети вырываться из локальных минимумов и в то же время гарантирует окончательную стабилизацию сети. Ловушки локальных минимумов досаждают всем алгоритмам обучения, основанным на поиске минимума, включая персептрон и сети обратного распространения, и представляют серьезную и широко распространенную трудность, которой часто не замечают. Стохастические методы позволяют решить эту проблему. Стратегия коррекции весов, вынуждающая веса принимать значение глобального оптимума в точке B, возможна.

В качестве объясняющей аналогии предположим, что на рис. 2 изображен шарик на поверхности в коробке. Если коробку сильно потрясти в горизонтальном направлении, то шарик будет быстро перекатываться от одного края к другому. Нигде не задерживаясь, в каждый момент шарик будет с равной вероятностью находиться в любой точке поверхности. Если постепенно уменьшать силу встряхивания, то будет достигнуто условие, при котором шарик будет на короткое время «застревать» в точке B. При еще более слабом встряхивании шарик будет на короткое время останавливаться как в точке A, так и в точке B. При непрерывном уменьшении силы встряхивания будет достигнута критическая точка, когда сила встряхивания достаточна для перемещения шарика из точки A в точку B, но недостаточна для того, чтобы шарик мог вскарабкаться из B в A. Таким образом, окончательно шарик остановится в точке глобального минимума, когда амплитуда встряхивания уменьшится до нуля.

Искусственные нейронные сети могут обучаться по существу тем же самым образом посредством случайной коррекции весов. Вначале делаются большие случайные коррекции с сохранением только тех изменений весов, которые уменьшают целевую функцию. Затем средний размер шага постепенно уменьшается, и глобальный минимум в конце концов достигается.

Это сильно напоминает отжиг металла, поэтому для ее описания часто используют термин

«имитация отжига». В металле, нагретом до температуры, превышающей его точку плавления, атомы находятся в сильном беспорядочном движении. Как и во всех физических системах, атомы стремятся к состоянию минимума энергии (единому кристаллу в данном случае), но при высоких температурах энергия атомных движений препятствует этому. В процессе постепенного охлаждения металла возникают все более низкоэнергетические состояния, пока в конце концов не будет достигнуто наинизшее из возможных состояний, глобальный минимум. В процессе отжига распределение энергетических уровней описывается следующим соотношением:

$$P(e) = e^{-e/kT} \tag{1}$$

где $P(e)$ – вероятность того, что система находится в состоянии с энергией e; k – постоянная Больцмана; T – температура по шкале Кельвина.

При высоких температурах $P(e)$ приближается к единице для всех энергетических состояний. Таким образом, высокоэнергетическое состояние почти столь же вероятно, как и низкоэнергетическое. По мере уменьшения температуры вероятность высокоэнергетических состояний уменьшается по сравнению с низкоэнергетическими. При приближении температуры к нулю становится весьма маловероятным, чтобы система находилась в высокоэнергетическом состоянии.

2.1 Больцмановское обучение

Этот стохастический метод непосредственно применим к обучению искусственных нейронных сетей:

1. Определить переменную T, представляющую искусственную температуру. Придать T большое начальное значение.

2. Предъявить сети множество входов и вычислить выходы и целевую функцию.

3. Дать случайное изменение весу и пересчитать выход сети и изменение целевой функции в соответствии со сделанным изменением веса.

4. Если целевая функция уменьшилась (улучшилась), то сохранить изменение веса. Если изменение веса приводит к увеличению целевой функции, то вероятность сохранения этого изменения вычисляется с помощью распределения Больцмана:

$$P(c) = e^{-c/kT} \tag{2}$$

где $P(c)$ – вероятность изменения c в целевой функции; k – константа, аналогичная константе Больцмана, выбираемая в зависимости от задачи; T – искусственная температура. Выбирается случайное число r из равномерного распределения от нуля до единицы. Если $P(c)$ больше, чем r, то изменение сохраняется, в противном случае величина веса возвращается к предыдущему значению.

Это позволяет системе делать случайный шаг в направлении, портящем целевую функцию, позволяя ей тем самым вырываться из локальных минимумов, где любой малый шаг увеличивает целевую функцию. Для завершения больцмановского обучения повторяют шаги 3 и 4 для каждого из весов сети, постепенно уменьшая температуру T, пока не будет достигнуто допустимо низкое значение целевой функции. В этот момент предъявляется другой входной вектор и процесс обучения повторяется. Сеть обучается на всех векторах обучающего множества, с возможным повторением, пока целевая функция не станет допустимой для всех них. Величина случайного изменения веса на шаге 3 может определяться различными способами.

Например, подобно тепловой системе весовое изменение w может выбираться в соответствии с гауссовским распределением:

$$P(w) = e^{-w^2/T^2} \tag{3}$$

где P(w) – вероятность изменения веса на величину w, T – искусственная температура.

Такой выбор изменения веса приводит к системе, аналогичной [3].

Так как нужна величина изменения веса Δw, а не вероятность изменения веса, имеющего величину w, то метод Монте-Карло может быть использован следующим образом:

1. Найти кумулятивную вероятность, соответствующую $P(w)$. Это есть интеграл от $P(w)$ в пределах от 0 до w. Так как в данном случае $P(w)$ не может быть проинтегрирована аналитически, она должна интегрироваться численно, а результат необходимо затабулировать.

2. Выбрать случайное число из равномерного распределения на интервале (0,1). Используя эту величину в качестве значения $P(w)$, найти в таблице соответствующее значение для величины изменения веса.

Свойства машины Больцмана широко изучались. В работе [1] показано, что скорость уменьшения температуры должна быть обратно пропорциональна логарифму времени, чтобы была достигнута сходимость к глобальному минимуму. Скорость охлаждения в такой системе выражается следующим образом:

$$T(t) = \frac{T_0}{log(1+t)}(5.4) \tag{4}$$

где $T(t)$ – искусственная температура как функция времени; T_0 – начальная искусственная температура; t – искусственное время.

Этот разочаровывающий результат предсказывает очень медленную скорость охлаждения (и данные вычисления). Этот вывод подтвердился экспериментально. Машины Больцмана часто требуют для обучения очень большого ресурса времени.

2.2 Обучение Коши

В работе [6] развит метод быстрого обучения подобных систем. В этом методе при вычислении величины шага распределение Больцмана заменяется на распределение Коши. Распределение Коши имеет, как показано на рис. 3, более длинные «хвосты», увеличивая тем самым вероятность больших шагов. В действительности распределение Коши имеет бесконечную (неопределенную) дисперсию. С помощью такого простого изменения максимальная скорость уменьшения температуры становится обратно пропорциональной линейной величине, а не логарифму, как для алгоритма обучения Больцмана. Это резко уменьшает время обучения. Эта связь может быть выражена следующим образом:

$$T(t) = \frac{T_0}{1+t}(5.5) \tag{5}$$

Распределение Коши имеет вид

$$P(x) = \frac{T(t)}{T^2(t)+x^2}(5.6) \tag{6}$$

где $P()$ есть вероятность шага величины x.

Рис. 3: Распределение Коши и распределение Больцмана

В уравнении (6) P(x) может быть проинтегрирована стандартными методами. Решая относительно x, получаем

$$x_c = \rho T(t) tg(P(x)), \tag{7}$$

где ρ – коэффициент скорости обучения; x_c – изменение веса.

Теперь применение метода Монте Карло становится очень простым. Для нахождения x в этом случае выбирается случайное число из равномерного распределения на открытом интервале $(\breve{}\pi/2, \pi/2)$ (необходимо ограничить функцию тангенса). Оно подставляется в формулу (7) в качестве $P()$, и с помощью текущей температуры вычисляется величина шага.

2.3 Метод искусственной теплоемкости

Несмотря на улучшение, достигаемое с помощью метода Коши, время обучения может оказаться все еще слишком большим. Способ, уходящий своими корнями в термодинамику, может быть использован для ускорения этого процесса. В этом методе скорость уменьшения температуры изменяется в соответствии с искусственной «теплоемкостью», вычисляемой в процессе обучения.

Во время отжига металла происходят фазовые переходы, связанные с дискретными изменениями уровней энергии. При каждом фазовом переходе может иметь место резкое изменение величины, называемой теплоемкостью. Теплоемкость определяется как скорость изменения температуры с энергией. Изменения теплоемкости происходят из-за попадания системы в локальные энергетические минимумы.

Искусственные нейронные сети проходят аналогичные фазы в процессе обучения. На границе фазового перехода искусственная теплоемкость может скачкообразно измениться. Эта псевдотеплоемкость определяется как средняя скорость изменения температуры с целевой функцией. В примере шарика в коробке сильная начальная встряска делает среднюю величину целевой функции фактически не зависящей от малых изменений температуры, т. е. теплоемкость близка к константе. Аналогично при очень низких температурах система замерзает в точке минимума, так что теплоемкость снова близка к константе. Ясно, что в каждой из этих областей допустимы сильные изменения температуры, так как не происходит улучшения целевой функции.

При критических температурах небольшое уменьшение температуры приводит к большому изменению средней величины целевой функции. Возвращаясь к аналогии с шариком, при «температуре», когда шарик обладает достаточной средней энергией, чтобы перейти из A в B, но недостаточной для перехода из B в A, средняя величина целевой функции испытывает скачкообразное изменение. В этих критических точках алгоритм должен изменять температуру очень медленно, чтобы гарантировать, что система не замерзнет случайно в точке A, оказавшись пойманной в локальный минимум. Критическая температура может быть обнаружена по резкому уменьшению искусственной теплоемкости, т. е. средней скорости изменения температуры с целевой функцией. При достижении критической температуры скорость изменения температуры должна замедляться, чтобы гарантировать сходимость к глобальному минимуму. При всех остальных температурах может без риска использоваться более высокая скорость снижения температуры, что приводит к значительному снижению времени обучения.

Список литературы

[1] Geman S., Geman D. 1984. Stohastic relaxation, Gibbs distribution and Baysian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence 6:721-41.

[2] Hinton G. E., Sejnowski T. J. 1986. Learning and relearning in Boltzmann machines. In Parallel distributed processing, vol. 1, p. 282-317. Cambridge, MA: MIT Press.

[3] Metropolis N., Rosenbluth A. W-.Rosenbluth M. N., Teller A. N., Teller E. 1953. Equations of state calculations by fast computing machines. Journal of Chemistry and Physics. 21:1087-91.

[4] Parker D. B. 1987. Optimal algorithms for adaptive networks. Second order Hebbian learning. In Proceedings of the IEEE First International Conference on Neural Networks, eds. M. Caudill and C. Buller, vol. 2, pp. 593-600. San Diego, CA: SOS Printing.

[5] Rumelhart D. E. Hinton G. E. Williams R. J. 1986. Learning internal representations by error propagation. In Parallel distributed processing, vol. 1, pp. 318-62. Cambridg, MA: MIT Press.

[6] Szu H., Hartley R. 1987. Fast Simulated annealing. Physics Letters. 1222(3,4): 157-62.

[7] Wassermann P. D. 1988. Combined backpropagation/Cauchi machine. Neural Networks. Abstracts of the First INNS Meeting, Boston 1988, vol. 1, p. 556. Elmsford, NY. Pergamon Press.

Следующая лекция

Лекция 19. Сети Хопфилда