ОСНОВНЫЕ ПРОБЛЕМЫ ВНЕДРЕНИЯ НЕЙРОННЫХ СЕТЕЙ И СПОСОБЫ ИХ РЕШЕНИЯ

Янковский Д.О., Савенко А.Г.

Институт информационных технологий БГУИР,
г. Минск, Республика Беларусь

Савенко А.Г. – м.т.н., старший преподаватель

В статье описаны и проанализированы основные проблемы разработки, обучения и внедрения нейронных сетей. Представлены некоторые пути решения.

Хотя концепция искусственной нейронной сети существует с 1950-х годов, лишь недавно появилось оборудование, позволяющее воплотить теорию в жизнь. Предполагается, что нейронные сети могут имитировать любую непрерывную функцию. Часто встречаются сети, которые не работают на должном уровне, или для получения достойных результатов требуется много времени. Следует подходить к проблеме статистически.

Одним из первых шагов должна быть правильная предварительная обработка данных. Помимо средней нормализации и масштабирования, анализ основных компонентов может быть полезен для ускорения обучения. Если размерность данных уменьшается до такой степени, что всё еще сохраняется надлежащая дисперсия, можно сэкономить пространство без существенного ущерба для качества данных. Кроме того, нейронные сети могут обучаться быстрее, когда им предоставляется меньше данных.

Уменьшение размерности может быть достигнуто путем разложения ковариационной матрицы обучающих данных с использованием разложения по сингулярным значениям на три матрицы. Предполагается, что первая матрица содержит собственные векторы. Кроме того, набор векторов, присутствующих в матрице, является ортонормированным, поэтому их можно рассматривать как базисные векторы. Выбираются первые несколько векторов из этой матрицы, число которых равно числу измерений, в которые необходимо сократить данные. Производя преобразование исходной матрицы (с исходными размерами) с помощью матрицы, полученной на предыдущем шаге, получаем новую матрицу, которая и уменьшена в размерах, и линейно преобразована.

Вышеуказанные шаги имеют математическую природу, но, по сути, происходит «проецирование» данных из более высокого измерения в более низкое измерение, подобно проецированию точек на плоскости на хорошо подобранной линии таким образом, чтобы расстояние, на которое проходит точка, было сведено к минимуму.

Хотя Джордж Кибенко в 1989 году доказал, что нейронные сети, имеющие даже один скрытый слой, могут аппроксимировать любую непрерывную функцию, может потребоваться ввести в сеть полиномиальные характеристики более высокой степени, чтобы получить более точные прогнозы. Также можно увеличить количество скрытых слоев. Фактически, число слоев сети равно наибольшей степени многочлена, который она должна быть в состоянии представить. Хотя это также может быть достигнуто путем увеличения числа нейронов в существующих слоях, для этого потребуется гораздо больше нейронов (и, следовательно, увеличенное время вычислений) по сравнению с добавлением скрытых слоев в сеть, для приближения функции с аналогичным количеством ошибок. С другой стороны, создание «глубоких» нейронных сетей приводит к нестабильным градиентам. Это можно разделить на две части: проблемы исчезновения и взрыва.

Веса нейронной сети, как правило, инициализируются случайными значениями, имеющими среднее значение равное нулю и стандартное отклонение равное единице, расположенными примерно на гауссовом распределении. Это гарантирует, что большинство весов находятся в диапазоне от -1 до 1. Сигмоидальная функция дает нам максимальную производную 0,25 (когда входное значение равно нулю). Это, в сочетании с тем фактом, что веса принадлежат к ограниченному диапазону, помогает убедиться, что абсолютное значение их продукта также составляет менее 0,25. Градиент персептрона включает в себя произведение многих таких терминов, каждый из которых составляет менее 0,25. Чем глубже углубляться в слои, тем больше и больше будет таких терминов, что приведет к исчезающей проблеме градиента.

По существу, градиент персептрона внешнего скрытого слоя (ближе к входному слою) будет определяться суммой произведений градиентов более глубоких слоев и весов, присвоенных каждой из связей между ними. Следовательно, очевидно, что мелкие слои имели бы меньший градиент. Это приведет к тому, что их веса будут меньше меняться во время обучения и со временем станут почти неизменными. Предполагается, что первые слои несут большую часть информации, однако они обучается меньше всего. Следовательно, проблема исчезающего градиента в конечном итоге приводит к гибели сети.

Могут быть обстоятельства, при которых вес может превышать единицу во время тренировки. В этом случае можно задаться вопросом, как исчезающие градиенты могут создавать проблемы. Это может привести к взрывной проблеме градиента, в которой градиент в более ранних слоях становится огромным. Если веса велики и смещение таково, что продукт с производной сигмоидальной функции активации также удерживает его на более высокой стороне, эта проблема возникнет. Но, с другой стороны, этого трудно достичь, так как увеличение веса может привести к более высокой стоимости входных данных для функции активации, где производная сигмоида будет довольно низкой. Это также помогает установить тот факт, что проблему исчезающего градиента трудно предотвратить. Для решения этой проблемы необходимо выбрать другие функции активации, избегая сигмовидной формы.

Хотя сигмоида является популярным выбором, поскольку он сдавливает входное значение между нулем и единицей, а также для его производной можно записать как функцию самой сигмоиды, полагаясь на неё, нейронные сети могут страдать от нестабильных градиентов. Более того, сигмоидальные выходы не центрированы по нулю, все они положительны. Это означает, что все градиенты будут либо положительными, либо отрицательными в зависимости от градиента единиц на следующем слое.

Наиболее рекомендуемая функция активации – Maxout. Maxout поддерживает два набора параметров. Используется тот, который дает более высокое значение для ввода в качестве функции активации. Кроме того, веса могут варьироваться в зависимости от определенных условий ввода. Одна такая попытка приводит к утечкам выпрямленных линейных единиц. В этом особом случае градиент остается 1, когда вход больше 0, и он получает небольшой отрицательный наклон, когда он меньше 0, пропорционально входу.

Другая проблема, которая встречается в нейронных сетях, особенно когда они имеют значительную глубину, — это внутренний ковариантный сдвиг. Статистическое распределение входных данных постоянно меняется в процессе обучения. Это может вызвать значительные изменения в области и, следовательно, снизить эффективность обучения. Решение данной проблемы – выполнить нормализацию для каждой мини-партии. Необходимо вычислить среднее значение и дисперсию для всех таких пакетов, а не для всех данных. Ввод нормализуется перед подачей его почти в каждый скрытый слой. Процесс обычно известен как нормализация партии. Применение нормализации партии может также помочь в преодолении проблемы исчезающих градиентов.

Нейронную сеть можно улучшить, реализовав отсев. Часто определенные узлы в сети случайно отключаются от некоторых или всех слоев нейронной сети. Следовательно, на каждой итерации получается новая сеть, и конечная сеть (полученная в конце обучения) является комбинацией всех таких новых подсетей. Это также помогает в решении проблемы переоснащения.

Какие бы настройки нейронной сети не применялись, необходимо всегда отслеживать процент мертвых нейронов в сети и соответственно регулировать скорость обучения.

Определенная диагностика может быть выполнена для параметров, чтобы получить лучшую статистику. Графики смещения и дисперсии являются двумя важными факторами. Их можно определить путем построения кривых с выводом функции потерь (без регуляризации) для обучения и наборов данных перекрестной проверки в зависимости от количества примеров обучения.

Если нейронная сеть страдает от высокой дисперсии, это означает, что обученные параметры хорошо соответствуют обучающему набору, но плохо работают при проверке на «невидимых» данных (обучающий или проверочный набор). Это может быть связано с тем, что модель «переходит» на тренировочные данные. Получение большой выборки данных может исправить ситуацию. Сокращение количества скрытых слоев в сети также может быть полезным в этом случае.

Хотя было замечено, что огромное количество обучающих данных может повысить производительность любой сети, получение большого количества данных может быть дорогостоящим и занимать много времени. В случае, если сеть страдает от высокого смещения или исчезающей проблемы градиентов, больший объём данных будет бесполезен с точки зрения обучения.