Оптимизация сети

<https://habr.com/ru/articles/318970/>

Обучение сети заключается в подборе параметров сети таким образом, чтобы минимизировать значение функции потерь (функции ошибок, целевой функции). Для этого обычно используется метод градиентного спуска.

Пусть – целевая функция, зависящая от параметров нейронной сети (весовые коэффициенты и смещения нейронов сети). Рассмотрим разложение целевой функции в ряд Тэйлора вблизи оптимальных значений параметров :

где

– вектор градиента,

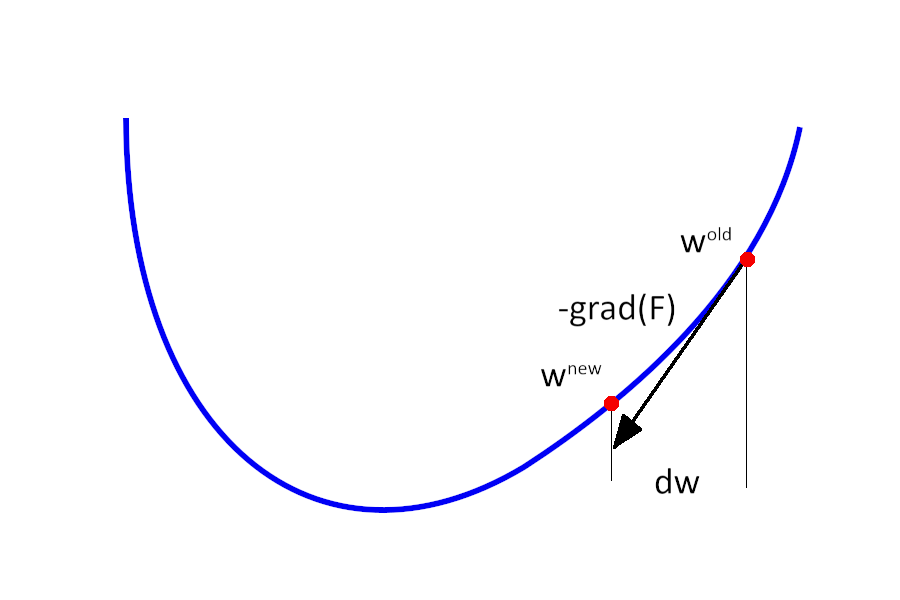
– матрица вторых производных, называемая гессианом.

Ограничимся первыми производными в разложении. В этом случае вектор градиента указывает направление изменения параметров сети:

– параметр обучения. Малое значение параметра обучения – медленное обучение, Большое значение – «перепрыгиваем» через точку минимума.

Способы выбора пакетов из обучающего набора:

1. Batch – целевая функция вычисляется на всех примерах обучающей последовательности.
2. Mini-batch - обучающей последовательность разбивается на группы, вычисляется на всех примерах группы.
3. Стохастический – вычисляется на одном примере, примеры из обучающей последовательности выбираются случайным образом.



***Оптимизация метода градиентного спуска.***

Зачем нужна оптимизация? Обычный метод градиентного спуска выглядит следующим образом:

*–* номер итерации.

Изображение выглядит как диаграмма, линия, График, дизайн

Автоматически созданное описание

Проблемы:

1. Застревание в локальных минимумах или седловых точках целевой функции, для функции большого числа переменных может быть очень много.
2. Сложный ландшафт целевой функции: плато чередуются с регионами сильной нелинейности. Производная на плато практически равна нулю, а внезапный обрыв приводит к резкому изменению градиента.
3. Слишком маленькая скорость обучения заставляет алгоритм сходиться очень долго и застревать в локальных минимумах, слишком большая — «пролетать» узкие глобальные минимумы или вовсе расходиться

**Идея оптимизации:** учитывать не только текущее состояние, но и предыдущие значения параметров сети.

Пусть среднее значение некоторого параметра за предыдущие итерации. Для упрощения вычислений можно использовать экспоненциальное скользящее среднее:

– текущее значение используемого параметра, например .

Коэффициент задаёт размер интервала усреднения, чем меньше значение – тем короче интервал усреднения.

Если накапливать градиент целевой функции, то модифицированный метод градиентного спуска примет вид (**Nesterov Accelerated Gradient**):

Напомним, что – градиент целевой функции в точке . Возможны два варианта: посчитать значение параметра на «старом» значении градиента затем найти градиент или наоборот.

Метод оптимизации **RMSprop** (root mean square propagation):

В этом методе используется усреднённый квадрат градиента:

*Идея ясна. Разница в выборе параметров для усреднения и начальных значениях при старте итераций.*