Stochastic gradient descent

From Wikipedia, the free encyclopedia

Stochastic gradient descent (often shortened in **SGD**), also known as **incremental**gradient descent, is a stochastic approximation of the gradient descent optimizationmethod for minimizing an objective function that is written as a sum ofdifferentiable functions. In other words, SGD tries to find minimums or maximums by iteration.

Contents

[hide]

Background

Both statistical estimation and machine learning consider the problem of minimizing an objective function that has the form of a sum:

$$Q(w) = \sum_{i=1}^{n} Q_i(w),$$

where the parameter w which minimizes $Q(w)$ is to be estimated. Each summand function Q_i is typically associated with the i-th observation in the data set (used for training).

In classical statistics, sum-minimization problems arise in least squares and inmaximum-likelihood estimation (for independent observations). The general class of estimators that arise as minimizers of sums are called M-estimators. However, in statistics, it has been long recognized that requiring even local minimization is too restrictive for some problems of maximum-likelihood estimation.[1] Therefore, contemporary statistical theorists often consider stationary points of the likelihood function (or zeros of its derivative, the score function, and other estimating equations).

The sum-minimization problem also arises for empirical risk minimization: In this case, $Q_i(w)$ is the value of the loss function at i-th example, and $Q(w)$ is the empirical risk.

When used to minimize the above function, a standard (or "batch") gradient descentmethod would perform the following iterations :

$$w := w - \eta \nabla Q(w) = w - \eta \sum_{i=1}^{n} \nabla Q_i(w),$$

where η is a step size (sometimes called the *learning rate* in machine learning).

In many cases, the summand functions have a simple form that enables inexpensive evaluations of the sum-function and the sum gradient. For example, in statistics,one-parameter exponential families allow economical function-evaluations and gradient-evaluations.

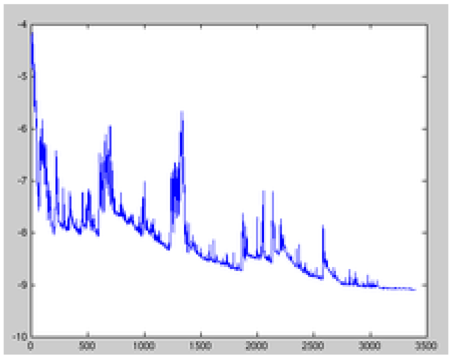
However, in other cases, evaluating the sum-gradient may require expensive evaluations of the gradients from all summand functions. When the training set is enormous and no simple formulas exist, evaluating the sums of gradients becomes very expensive, because evaluating the gradient requires evaluating all the summand functions' gradients. To economize on the computational cost at every iteration, stochastic gradient descent samples a subset of summand functions at every step. This is very effective in the case of large-scale machine learning problems.[2]

Iterative method

In stochastic (or "on-line") gradient descent, the true gradient of $Q(w)$ is approximated by a gradient at a single example:

$$w := w - \eta \nabla Q_i(w).$$

As the algorithm sweeps through the training set, it performs the above update for each training example. Several passes can be made over the training set until the algorithm converges. If this is done, the data can be shuffled for each pass to prevent cycles. Typical implementations may use an adaptive learning rate so that the algorithm converges.



Fluctuations in the total objective function as gradient steps with respect to mini-batches are taken.

In pseudocode, stochastic gradient descent can be presented as follows:

```
- Choose an initial vector of parameters $w$ and learning rate $\eta$.
- Repeat until an approximate minimum is obtained:
    - Randomly shuffle examples in the training set.
    - For$i = 1, 2, \ldots, n$ , do:
        - $w := w - \eta \nabla Q_i(w).$
```

A compromise between computing the true gradient and the gradient at a single example, is to compute the gradient against more than one training example (called a "mini-batch") at each step. This can perform significantly better than true stochastic gradient descent because the code can make use of vectorization libraries rather than computing each step separately. It may also result in smoother convergence, as the gradient computed at each step uses more training examples.

The convergence of stochastic gradient descent has been analyzed using the theories of convex minimization and of stochastic approximation. Briefly, when the learning rates η decrease with an appropriate rate, and subject to relatively mild assumptions, stochastic gradient descent converges almost surely to a global minimum when the objective function is convex or pseudoconvex, and otherwise converges almost surely to a local minimum.[3][4] This is in fact a consequence of the Robbins-Siegmund theorem.[5]