گرادیان تصادفی نزولی (SGD)[[1]](#footnote-1):

گرادیان تصادفی نزولی یک راه‌کار ساده و در عین حال موثر برای مسائل کمینه‌سازی[[2]](#footnote-2) است.

روش‌های کمینه‌سازی‌ای که به صورت دسته‌ای صورت می‌گیرند مانند روش BFGS با حافظه‌ی محدود[[3]](#footnote-3) [] که از کل مجموعه داده‌ی آموزش برای بروزرسانی پارامتر‌های مساله در هر تکرار استفاده می‌کنند به سرعت به سمت نقطه‌ی بهینه‌ی محلی[[4]](#footnote-4) همگرا[[5]](#footnote-5) می‌شوند.همچنین این روش‌ها روی مجموعه داده‌های بزرگ که در حافظه‌ی اصلی[[6]](#footnote-6) جا نمی‌گیرند کارایی مناسبی ندارند زیرا هزینه‌ی محاسبات روی کل مجموعه‌ی داده زیاد بوده و بسیار کند انجام می‌شود و در برخی موارد روی یک ماشین غیرقابل انجام است. مشکل دیگری که روش‌های بهینه‌سازی دسته‌ای[[7]](#footnote-7) با آن مواجه هستند این است که به خوبی با داده‌های جدیدی که به صورت برخط[[8]](#footnote-8) به مجموعه داده اضافه می‌شوند هماهنگ نمی‌شوند و نمی‌توانند آن‌ها را در مساله‌ی بهینه‌سازی شرکت دهند.

گرادیان تصادفی نزولی سعی بر حل هر دو مشکل مطرح شده دارد. ایده‌ی روش SGD بخاطر هزینه‌ی زیاد عمل انتشار معکوس در روش‌های شبکه‌ی عصبی و حل این مشکل ایجاد شد.

نکته روش گرادیان تصادفی نزولی این است که نیاز نیست در هر تکرار کل مجموعه داده مورد بررسی قرار بگیرد و این روش با یک یا چند نمونه از مجموعه‌ی داده در هر تکرار قابل اجرا است. فرض کنید تتا پارامتری است که می‌خواهیم تابع هدف Q را کمینه کند و قصد داریم آن را تخمین بزنیم. در هر تکرار از حل مساله، یک دسته از داده‌های مجموعه داده را انتخاب می‌کنیم و در هر تکرار میزان تغییر پارامتر تتا (Δ*w*) را از طریق رابطه‌ی زیر بدست می‌آوریم.

../../../../../Desktop/Screen%20Shot%201396-04-01%20at%2017.

که در آن پارامتر الفا میزان تغییر پارامتر تتا و پارامتر بتا که به آن نرخ یادگیری[[9]](#footnote-9) گفته می‌شود میزان همگرائی را تنظیم می‌کنند. بعد از هر تکرار از حل مساله پارامتر تتا را طبق رابطه‌ی + بروزرسانی می‌کنیم.

../../../../../Desktop/Screen%20Shot%201396-04-01%20at%2018.

1. Stochastic gradient descent [↑](#footnote-ref-1)
2. minimization [↑](#footnote-ref-2)
3. limited memory BFGS [↑](#footnote-ref-3)
4. local optima [↑](#footnote-ref-4)
5. converge [↑](#footnote-ref-5)
6. main memory [↑](#footnote-ref-6)
7. batch optimization [↑](#footnote-ref-7)
8. online [↑](#footnote-ref-8)
9. learning rate [↑](#footnote-ref-9)