تخمین توابع غیر خطی بر پایه شبکه‌های عصبی با آموزش گرادیان مرتبه اول و دوم

اعظم ربیعی دولت آبادی[۱]، محمد تشنه لب[۲]

[۱] دانشگاه آزاد اسلامی واحد دولت‌آباد
اصفهان، ایران
rabiee@iauda.ac.ir

۲ دانشگاه خواجه نصیر طوسی
تهران، ایران
teshnehlab@eetd.kntu.ac.ir

چکیده

الگوریتم گرادیان نزولی پایهٔ بسیاری از الگوریتم‌های بهینه‌سازی است و از آن در یادگیری شبکه‌های عصبی و حداقل‌سازی مقدار خطای شبکه برای تنظیم پارامترهای شبکه استفاده می‌شود. دسته وسیعی از تحقیقات برای افزایش سرعت این الگوریتم در آموزش شبکه‌های عصبی چندلایه پیشرو انجام شده است؛ که از نتایج آن می‌توان به انواع روش‌های گرادیان مرتبه اول و دوم اشاره کرد. این مقاله‌ی مروری به جزئیات دقیق این روش‌ها نمی‌پردازد؛ بلکه هدف آن مشخص کردن ویژگی‌های اصلی این روش‌ها و ارتباطشان است. در این گزارش، از چهارروش گرادیان مرتبه اول و سه روش مرتبه دوم برای چهار تابع استاندارد جهت تخمین استفاده شد. در بین روش‌های گرادیان مرتبه اول، همگرایی سریعتر در روش RPROP و در روش‌های مرتبه دوم در الگوریتم لونمارکوت (LM) دیده شد. واضح است که روش‌های مرتبه اول حجم محاسبات کمتری دارند و مقدار فضای کمتری نیاز دارند؛ در عوض برای بسیاری از مسائل بزرگ قابل استفاده نیستند.

کلمات کلیدی

تخمین توابع، شبکه عصبی چندلایه پیشرو، گرادیان مرتبه اول و دوم.

۱- مقدمه

برای آموزش شبکه‌های عصبی مصنوعی از روش‌های گرادیان نزولی به گستردگی استفاده می‌شود. منظور از گرادیان نزولی، تغییر وزن‌ها با توجه به مشتقات خطای شبکه، به گونه‌ای است که خطای شبکه به حداقل برسد. مسئلهٔ یادگیری یک نگاشت ورودی-خروجی از یک مجموعه مثال P تایی را می‌توان به یک مسئله تخمین تابع و حداقل‌سازی مقدار تابع خطا روی مجموعه مثال‌ها تبدیل کرد. از بین توابع خطای متفاوتی که می‌توان برای این مسئله تعریف کرد، تابع مجموع مربعات خطا، از پرکاربردترین آن‌ها، بصورت رابطه (۱) تعریف می‌شود[۱-۴].

$$E(w)=\frac{1}{2}\sum_{p=1}^{P}E_p=\frac{1}{2}\sum_{p=1}^{P}\sum_{i=1}^{n_o}(t_{pi}-o_{pi}(w))^2 \quad (۱)$$

که در این رابطه، P تعداد الگوها، t_{pi} و o_{pi} خروجی‌های مطلوب و خروجی نرون i-اُم لایه آخر برای الگوی p-اُم، و n_o تعداد نرون‌های لایه آخر است. روش‌های گرادیان مرتبه اول از مشتقات جزئی مرتبه اول تابع خطای فوق استفاده می‌کنند. در روش‌های گرادیان مرتبه دوم برای سرعت دادن بیشتر به همگرایی از مشتقات جزئی مرتبه دو تابع خطای فوق استفاده می‌شود. در بسیاری از موارد تخمین تابع با استفاده از روش‌های گرادیان مرتبه اول به تعداد تکرار زیادی از الگوریتم یادگیری نیاز دارد که این مسئله در مسائل برخط و کنترلی مناسب نیست. روش‌های بر اساس گرادیان مرتبه دوم به عنوان

روش‌های جایگزین بر گرادیان مرتبه اول برای سرعت بخشیدن به مرحله یادگیری ارائه شده‌اند. الگوریتم‌های متنوعی بر اساس گرادیان مرتبه اول یا دوم تعریف شده‌اند. این گزارش به بررسی این الگوریتم‌ها پرداخته و نتایج تجربی این روش‌ها را با چند مسئله‌ی تخمین تابع ارائه می‌کند.

۲- روش‌های مبتنی بر گرادیان مرتبه اول

۲-۱- پس انتشار خطا و شیب‌دارترین نزول

فرایند یادگیری پس‌انتشار خطا در دو مرحله انجام می‌شود:

- محاسبه گرادیان‌های تابع خطا برای هر الگو
- تغییر وزن‌ها به دو صورت
  - بر خط؛ بعد از هر الگو
  - خارج خط؛ با جمع گرادیان‌ها روی تمام الگوها

اگر گرادیان تابع خطا روی تمام الگوها را با g_k نشان‌دهیم؛ رابطه تغییر وزن خارج خط بصورت رابطه (۲) و رابطه تغییر وزن برخط بفرم رابطه (۳) است.

$$w_{k+1} = w_k - \varepsilon g_k \qquad (۲)$$

$$w_{k+1} = w_k - \varepsilon \nabla E_p(w_k) \qquad (۳)$$

که در این روابط ε، نرخ یادگیری یا گام است؛ برای نرخ یادگیری کوچک، روش‌های برخط و خارج خط بسیار نزدیک بهم هستند ولی سرعت همگرایی کم است. در نرخ یادگیری بزرگتر، تفاوت روش‌های برخط و خارج خط بیشتر به چشم می‌خورد. با افزایش نرخ یادگیری تا جایی که همگرایی تضمین شود، سرعت همگرایی افزایش می‌یابد.

زمانی که تمام الگوها قبل از شروع یادگیری در دسترس نباشند، از روال برخط استفاده می‌شود. از طرف دیگر وقتی تمام الگوها در دسترس هستند، اطلاعات گرادیان کل قبل از تصمیم‌گیری برای مرحله بعد، محاسبه می‌شود. این اطلاعات کلی از تمام الگوها، از تغییر وزن‌های بیهوده ناشی از الگوهای متفاوت بصورت جداگانه جلوگیری می‌کند. در بعضی موارد با کمی تغییر و تنظیم الگوریتم برخط، با تعداد کمتری داده، به جواب نهایی خواهیم رسید و سرعت یادگیری در چنین مواردی اصلاً قابل مقایسه با روش‌های خارج خط نیست. از جمله این تنظیمات می‌توان به انتخاب اندازه نرخ یادگیری، اندازه نرخ مومنتم، برنامه کاهش نرخ یادگیری (نرخ یادگیری تطبیقی)، تغییر وزن بر اساس زیر مجموعه‌ای از الگوهای مرتبط، آموزش با دسته دادگان آموزشی کوچک بصورت خارج خط، اصلاحات انتخابی در صورتی که خطا از حد آستانه‌ای بزرگتر است، دنباله الگوهای ورودی نامرتب تصادفی و . . . . اشاره کرد.

روش BP سنتی شامل بعضی کاستی‌ها و معایب از جمله سرعت همگرایی پایین، حساسیت به مقادیر اولیه، افتادن در دام مینیمم‌های محلی و ناپایداری با نرخ آموزش بزرگ می‌باشد. دسته وسیعی از تحقیقات در این زمینه برای برطرف کردن این مشکلات وجود دارد. از جمله روش‌های اصلاح‌شده‌ای که بر اساس گرادیان مرتبه اول هستند می‌توان به استفاده از مومنتم، مومنتم جداشده[۱]، پس انتشار خطا با نرخ آموزش متغیر، دلتا بار دلتا، RPROP[2]، پس انتشار خطای وفقی[۳]، پس انتشار خطای گسترده[۴] و . . . . اشاره کرد.

۲-۲- پس انتشار با مومنتم

در این روش برای پیشگیری از افتادن در دام مینیمم محلی و تغییر وزن‌های ناگهانی، ضریبی از مقدار تغییر وزن در مرحله قبل به مرحله فعلی آموزش اضافه می‌شود. رابطه تغییر وزن در این روش به فرم رابطه (۴) است[۱-۴].

$$w_{k+1} = w_k - \varepsilon \nabla E_p(w_k) + \mu(w_k - w_{k-1}) \qquad (۴)$$

در رابطه (۴)، μ نرخ مومنتم است.

۲-۳- نرخ آموزش ثابت یا متغیر

در فرمول‌های اولیه، نرخ آموزش، ε، ثابت در نظر گرفته می‌شد. متاسفانه اگر نرخ آموزش یک ثابت دلخواه در نظر گرفته شود، هیچ ضمانتی برای همگرایی شبکه به نقطه‌ای مناسب وجود ندارد. در مقاله [۵] مسئله نرخ آموزش ثابت برای مسایل برخط بررسی شده است؛ نتایج این مقاله نشان می‌دهد که زمانی همگرایی LMS[۵] ضمانت می‌شود که $\varepsilon < 1/(N\sqrt{\lambda_{\max}})$ باشد؛ در این رابطه، N تعداد پارامترهای شبکه و $\lambda_{\max}$ بزرگترین مقدار ویژه ماتریس کواریانس ورودی‌ها (ماتریس هسین[۶]) می‌باشد. تاثیر خودهمبستگی ورودی‌ها در فرایند یادگیری در [۶] بررسی شده است؛ در این مقاله بهترین مقدار نرخ آموزش ثابت برای گرادیان نزولی $1/\lambda_{\max}$ تعریف شده است. این نتایج را نمی‌توان به یک شبکه چند لایه با توابع انتقال غیر خطی گسترش داد؛ این نتایج برای شبکه‌های عصبی چندلایه فقط می‌توانند نقطه شروعی برای تجربیات باشد.

همگرایی LMS با نرخ آموزش متغیر وفقی در [۷] بررسی شده است. نتایج اصلی این مقاله این است که نرخ یادگیری ε_n برای n امین تکرار یادگیری (یک الگو در هر تکرار)، زمانی به نقطه بهینه می‌رسد که در روابط (۵) صدق کند.

$$\sum_{n=1}^{\infty} \varepsilon_n = \infty \qquad \sum_{n=1}^{\infty} \varepsilon_n^2 < \infty \qquad (۵)$$

۲-۴- روش RPROP

الگوریتم RPROP [۸]، یا پس انتشار محکم و انعطاف‌پذیر، روشی بر اساس گرادیان مرتبه اول برای تغییر وزن‌های یک شبکه چندلایه پیشرو است. در این روش، تغییر وزن‌ها به اندازه مشتقات خطا بطور مستقیم وابسته نیست؛ بلکه با تغییر علامت مشتق جزئی خطا تغییر

می کند. این الگوریتم با یک مقدار اولیه $\Delta_{ij}^{(0)}$ای بازای هر وزن در شبکه، شروع می‌شود. رابطه تغییر وزن‌ها بصورت رابطه (۶) است.

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t)}, & if \quad \dfrac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ +\Delta_{ij}^{(t)}, & if \quad \dfrac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ 0, & else \end{cases} \qquad (۶)$$

بعد از هر تغییر وزن مقادیر $\Delta_{ij}^{(t)}$ برای مرحله بعدی آموزش بصورت رابطه (۷) تغییر می‌کند.

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^{+}.\Delta_{ij}^{(t-1)}, & if \quad \dfrac{\partial E^{(t-1)}}{\partial w_{ij}}.\dfrac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ \eta^{-}.\Delta_{ij}^{(t-1)}, & if \quad \dfrac{\partial E^{(t-1)}}{\partial w_{ij}}.\dfrac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ \Delta_{ij}^{(t-1)}, & else \end{cases} \qquad (۷)$$

در رابطه (۷)، نرخ آموزش افزایشی(η^{+}) و کاهشی(η^{-}) معمولاً مقادیر ثابت ۱/۲ و ۰/۵ هستند.

۳- روش‌های مبتنی بر گرادیان مرتبه دوم

برای سرعت دادن به روش‌های گرادیان مرتبه اول و برطرف کردن معایب آن‌ها، روش‌هایی ارائه شده است که در آن از مشتقات جزئی مرتبه دو تابع خطا یا خروجی شبکه استفاده می‌شود. رابطه (۸) روش *نیوتن کلاسیک* را که پایه بسیاری از روش‌های نیوتنی و گرادیان مرتبه دو است، نشان می‌دهد.

$$w_{k+1} = w_k - \varepsilon H^{-1} g \qquad (۸)$$

در رابطه (۸)، H ماتریس هسین، g بردار گرادیان تابع خطای شبکه و ε نرخ آموزش است. در این روش تغییر وزن‌ها، با محاسبهٔ مستقیم ماتریس هسین انجام می‌شود. نکته کلیدی و تفاوت روش‌های مبتنی بر گرادیان مرتبه دو در محاسبه معکوس ماتریس هسین می‌باشد.

۳-۱- ماتریس هسین

ماتریس هسین [۹] گرادیان مرتبه دوم تابع خطا نسبت به وزن‌های شبکه است و با H نشان داده می‌شود. در الگوریتم LMS، ماتریس H، همان ماتریس خودهمبستگی ورودی‌ها، R_x، است[۱]. ماتریس هسین نقش مهمی را در مطالعه شبکه‌های عصبی بازی می‌کند، به عنوان مثال:

۱- مقادیر ویژه ماتریس H تاثیر عمیقی بر روی پویایی یادگیری پس انتشار خطا دارند.

۲- از معکوس ماتریس H برای هرس وزن‌های نامناسب از یک پرسپترون چندلایه استفاده می‌شود.

۳- ماتریس H، پایه فرمولهای روش‌های گرادیان مرتبه دوم است. این روش‌ها معمولاً بهتر از BP عمل می‌کنند و برای ارتقاء و افزایش سرعت روش‌های مرتبه اول معرفی شده‌اند.

۳-۲- روش کوشی- نیوتن

هدف روش کوشی- نیوتن[۴]، محاسبه ماتریس M بصورت تکراری به گونه‌ای است که حد آن به معکوس ماتریس هسین میل کند. پس M حاصل باید در رابطه (۸) به جای معکوس ماتریس هسین قرار بگیرد، تا زمانی که شبکه به یک نقطه مینیمم برسد (رابطه (۹)).

$$\begin{aligned} w_{k+1} &= w_k + M_{k+1}.\Delta g_k \\ \Delta g_k &= g_{k+1} - g_k \end{aligned} \qquad (۹)$$

در این روش، از یکی از فرمول‌های محاسبات تکراری DFP[7] یا BFGS[8]، به ترتیب روابط (۱۰) و (۱۱)، برای محاسبه M استفاده می‌شود.

$$M_{k+1} = M_k + \frac{\Delta w_k.\Delta w_k^T}{\Delta w_k^T.\Delta g_k} - \frac{M_k \Delta g_k \Delta g_k^T M_k}{\Delta g_k^T M_k \Delta g_k} \qquad (۱۰)$$

$$\begin{aligned} M_{k+1} = {} & (I - \frac{\Delta w_k \Delta g_k^T}{\Delta w_k^T \Delta g_k}) M_k (I - \frac{\Delta g_k \Delta w_k^T}{\Delta w_k^T \Delta g_k}) \\ & + \frac{\Delta w_k \Delta w_k^T}{\Delta w_k^T \Delta g_k} \end{aligned} \qquad (۱۱)$$

این روش نیز مانند بسیاری از روش‌های مرتبه دوم نیاز به فضا و محاسبات تکراری زیادی دارند.

۳-۳- روش گرادیان توام[9]

در روش گرادیان توام، جهت حرکت به گونه‌ای انتخاب می‌شود که همواره g_k بر Δw_{k-1} عمود باشد. از طرفی این روش در فضای n بُعدی، n بردار Δw_k را به گونه‌ای تولید می‌کند که مستقل خطی هستند و بنابراین در این روش ادعا می‌شود که حداکثر با n گام، الگوریتم همگرا می‌گردد. رابطه (۱۲) معادلات تغییر وزن، نرخ آموزش و جهت حرکت در این روش را نشان می‌دهد. [۱و۱۰].

$$\begin{aligned} w_{k+1} &= w_k + \eta_k d_k, \\ \eta_k &= \frac{g_k^T.g_k}{d_k^T.H.d_k}, \\ d_k &= -g_k + \beta_k d_{k-1}, \quad d_0 = -g_0 \end{aligned} \qquad (۱۲)$$

در رابطه ۱۲، η_k نرخ آموزش و d_k جهت حرکت است. همچنین β_k ضریبی است که از یکی از سه رابطه (۱۳)، (۱۴) و (۱۵) قابل محاسبه است.

(۱۳) رابطه‌ی فلچر- ریو:

$$\beta_k = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}}$$

(۱۴) رابطه‌ی پولاک-ریبیر:

$$\beta_k = \frac{g_k^T (g_k - g_{k-1})}{g_{k-1}^T g_{k-1}}$$

(۱۵) رابطه‌ی بیل- سورنسون:

$$\beta_k = \frac{g_k^T (g_k - g_{k-1})}{d_{k-1}^T (g_k - g_{k-1})}$$

می‌توان نشان داد، روابط ۱۴ و ۱۵ همان رابطه ۱۳ هستند. در روش گرادیان توام اگر بعد از n تکرار مسئله همگرا نشد، d_k مجدداً به مقدار منفی گرادیان مقداردهی می‌شود. این روش برای مسائلی با مقیاس بزرگ مناسب است.

گرادیان توام در مقایسه با *BP* به همراه مومنتم

استفاده از مومنتم در آموزش را می‌توان تقریبی از گرادیان توام در نظر گرفت. در هر دو روش گرادیان توام و مومنتم، جهت گرادیان با یک عبارت جدید تغییر می‌کند. تفاوت اصلی این دو روش در این است که پارامتر β در گرادیان توام بصورت اتوماتیک تولید می‌شود؛ در صورتی که ضریب مومنتم با آزمون و خطا بدست می‌آید.

۳-۴- روش لونمارکوت (LM)

روش لونمارکوت برای مسائلی که در آن‌ها ماتریس هسین مثبت تعریف شده یا معکوس‌پذیر نباشد مناسب است؛ چرا که هدف از این روش، اضافه‌کردن یک ماتریس مثبت به ماتریس هسین به گونه‌ای است که ماتریس حاصل معکوس‌پذیر باشد. رابطه‌ی ۱۶، معادله تغییر وزن‌ها در این روش را نشان می‌دهد[۱۱و۱۲].

$$\Delta w = [J^T(w)J(w) + \mu.I]^{-1} J^T(w)e(w) \qquad (۱۶)$$

$$J(w) = \begin{bmatrix} \frac{\partial e_1}{\partial w_1} & \cdots & \frac{\partial e_1}{\partial w_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial e_N}{\partial w_1} & \cdots & \frac{\partial e_N}{\partial w_n} \end{bmatrix}$$

در رابطه‌ی ۱۶، J ماتریس ژاکوبین و e بردار خطا است. همچنین μ ضریب مثبتی است که معمولاً کمی بزرگتر از اندازهٔ منفی‌ترین مقدار ویژه ماتریس هسین در نظر گرفته می‌شود. توجه کنید که هر گاه μ بزرگ باشد، این الگوریتم با گام 1/ μ همان روش شیب‌دارترین نزول خواهد بود؛ در صورتی که با μ کوچک، الگوریتم گاوس- نیوتن را خواهیم داشت. روش LM تغییر یافتهٔ روش گاوس- نیوتن می‌باشد.

۳-۵- روش گاوس- نیوتن:

این روش مشابه روش *LM* می‌باشد با این تفاوت که رابطه تغییر وزن‌ها به صورت رابطه (۱۷) است.

$$\Delta w = [J^T(w)J(w)]^{-1} J^T(w)e(w) \qquad (۱۷)$$

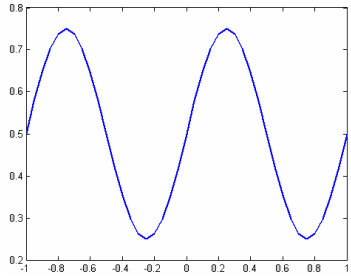
۴- شبیه‌سازی و نتایج

الگوریتم‌های گرادیان ذکر شده در بخش‌های قبل، بر روی چهار مسئله تخمین تابع متفاوت تست شده‌اند. برای مقایسه این روش‌ها از الگوریتم‌های پس انتشار خطای استاندارد، پس انتشار با مومنتم[۱۰]، پس انتشار با نرخ آموزش متفاوت[۱۱] و *RPROP* به عنوان روش‌های گرادیانی مرتبه اول و از الگوریتم‌های گرادیان توام (*CG*)، کوشی- نیوتن(*QN*) و لونبرگ- مارکوارد (*LM*) به عنوان روش‌های گرادیانی مرتبه دوم استفاده شده است. شبکه‌ها در هر پیاده‌سازی ساختار لایه‌ای یکسان دارند و برای آموزش شبکه‌ها از وزن‌های اولیه یکسان استفاده شده است. این وزن‌های اولیه بصورت تصادفی هستند ولی توسط روش *Nguyen-Widrow* نرمال شده‌اند. در این پیاده‌سازی آموزش بر روی سه دسته وزن‌های اولیه متفاوت انجام شد. مقدار اولیه نرخ آموزش در تمام این پیاده‌سازی‌ها، ۰/۰۵ در نظر گرفته شد.

روش پس انتشار با نرخ آموزش متفاوتی(*VLR*) که در این پیاده‌سازی استفاده شد، روشی ساده است؛ به این صورت که با مقدار فعلی نرخ آموزش، مقادیر خطا و وزن‌های جدید محاسبه می‌شوند؛ اگر مقدار خطا نسبت به مرحله قبل کاهش داشته باشد، نرخ آموزش در ضریب *lr_inc* ضرب می‌شود. در صورتی که خطا نسبت به مرحله قبل افزایش یابد، نرخ آموزش طی ضریب *lr_dec* کاهش می‌یابد و وزن‌ها به مقادیر قبلی خود برمی‌گردند. مقادیر *lr_inc* و *lr_dec* به ترتیب ۱/۰۵ و ۰/۷ در نظر گرفته شد. رابطه تغییر وزن‌ها در این روش همان گرادیان نزولی استاندارد (پس انتشار استاندارد) می‌باشد. در این پیاده‌سازی، همچنین، مقدار ضریب مومنتم در پس‌انتشار با مومنتم ۰/۹ و در الگوریتم گرادیان توام روش *Fletcher-Reeves* استفاده شد.

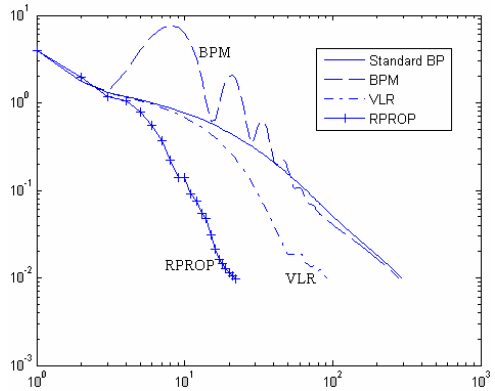
مسئله ۱: تخمین تابع سینوسی

به عنوان اولین مسئله تخمین تابع، شبکه‌ای ۱-۱۵-۱ با تابع انتقال *tansig* در لایه مخفی و خطی در لایه آخر، برای تخمین تابع $y = 1/2 + 1/4\sin(2\pi.x)$ ، شکل(۱) با ورودی [۱ ۱-] در نظر گرفته شد. این مسئله شامل ۴۰ جفت ورودی/خروجی داده آموزشی است.



شکل ۱- نمودار تابع سینوسی

شکل (۲) همگرایی شبکه‌های *Standard BP*، *BPM*، *VLR* و *RPROP* را برای رسیدن به مقدار خطای ۰/۰۱ در این مسئله نشان می‌دهد.

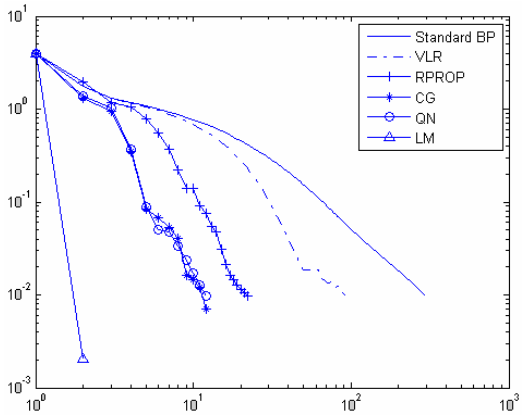


شکل ۲- همگرایی شبکه‌های گرادیان مرتبه اول برای مسئله تخمین تابع سینوسی (مجموع مربعات خطا بازای هر *epoch*)

در شکل (۲)، پس‌انتشار استاندارد، کندترین همگرایی را با کمترین شیب بصورت یکنواخت دارد. روش *BPM*، در ابتدا جستجوی گسترده‌ای را قبل از یافتن مسیر رسیدن به مینیمم خطا انجام می‌دهد؛ ولی با همان سرعت پس انتشار استاندارد، تنها با تفاوت *epoch* کمتر از ۱۰، همگرا می‌شود. در روش *VLR*، به علت بزرگ بودن نرخ همگرایی در ابتدای آموزش سریعتر همگرا می‌شود؛ شیب همگرایی این روش در انتهای کار برای دقت محاسبات بیشتر کمتر از حتی پس‌انتشار استاندارد است. روش *RPROP* با سرعت قابل توجهی به نسبت دیگر روش‌های گرادیان مرتبه اول همگرا می‌گردد.

روش‌های مرتبه اول در مقایسه با مرتبه دوم

شکل (۳) همگرایی روش‌های گرادیانی مرتبه اول را در مقایسه با روش‌های گرادیانی مرتبه دوم نشان می‌دهد.



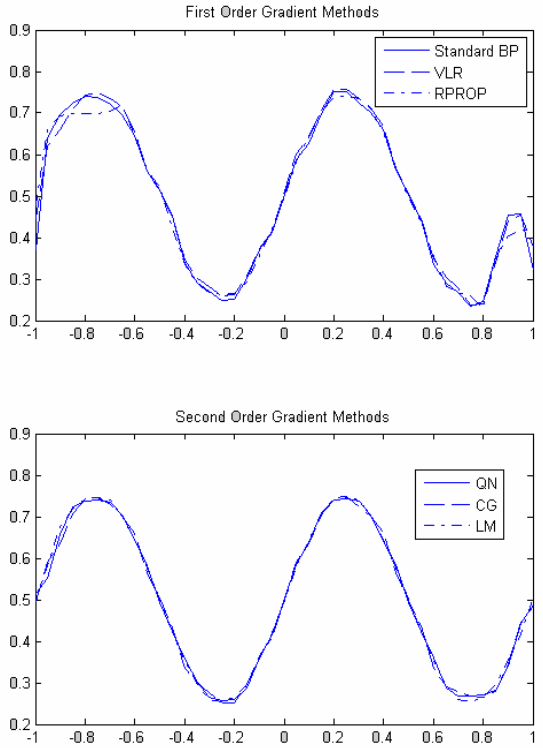
شکل ۳- همگرایی شبکه‌های گرادیان مرتبه اول در مقایسه با مرتبه دوم برای مسئله تخمین تابع سینوسی

بطور کلی روش‌های مرتبه دوم بسیار سریعتر از مرتبه اول همگرا می‌شوند. روش‌های *CG* و *QN* بسیار شبیه به هم عمل می‌کنند و *LM* بطور قابل ملاحظه‌ای سریع است. در روش‌های مرتبه دوم، علاوه بر افزایش سرعت همگرایی، آموزش نیز به بهترین شکل انجام می‌شود، وزن‌ها به خوبی تغییر می‌کنند و نتایج شبیه‌سازی بسیار بهتر از روش‌های مرتبه اول است. شکل (۴) شاهدی بر این ادعاست.

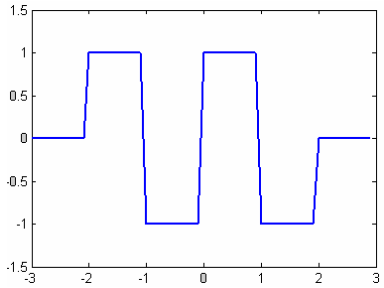
مسئله ۲: تخمین تابع پله

برای تخمین تابع پلهٔ نشان‌داده شده در شکل (۵)، همان شبکه ۱-۱۵-۱ با همان توابع انتقال *tansig* در لایه مخفی و خطی در لایه آخر با ۶۰ جفت ورودی- خروجی داده‌های آموزشی استفاده شد.

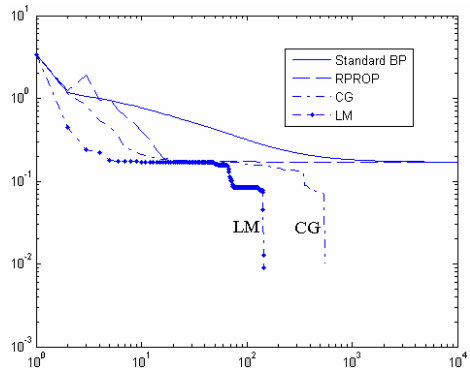
شکل (۶) همگرایی دو شبکه‌ی مرتبه اول پس انتشار استاندارد و *RPROP* و دو شبکه‌ی مرتبه دوم *CG* و *LM* را برای رسیدن به خطای ۰/۰۱ در تخمین تابع پله نشان می‌دهد. با وجود تعداد تکرارهای زیاد، روش‌های مرتبه اول، (حتی *RPROP* که سرعت همگرایی بهتری دارد) نیز برای مسئله پله به همگرایی نرسیدند؛ در مقابل روش‌های مرتبه دوم با تکرارهای بسیار کمتر و به راحتی همگرا شدند.



شکل ۴- نتایج شبیه‌سازی مسئله تخمین تابع سینوسی برای شبکه‌های مرتبه اول در مقایسه با مرتبه دوم

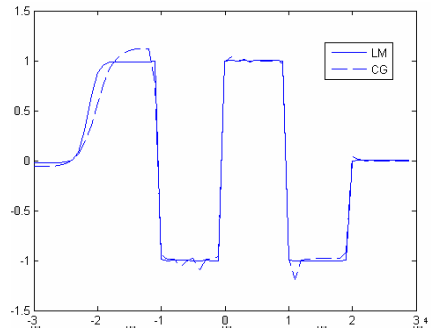


شکل ۵- نمودار تابع پله



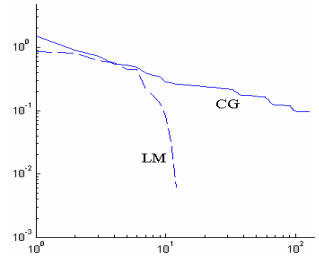
شکل ۶- همگرایی شبکه‌ها برای مسئله پله
(مجموع مربعات خطا بازای هر *epoch*)

شکل پله‌ای روش‌های CG و LM به این خاطر است که در این روش‌ها، بعد از گذشت n مرحله از آموزش، بردار گرادیان به مقدار اولیه خود *reset* می‌شود. در این آموزش روش LM علاوه بر همگرایی سریعتر، نتایج شبیه‌سازی بهتری نسبت به CG نیز نشان داد(شکل ۷).



شکل ۷- نتایج شبیه‌سازی شبکه‌های مرتبه دوم برای تخمین تابع پله

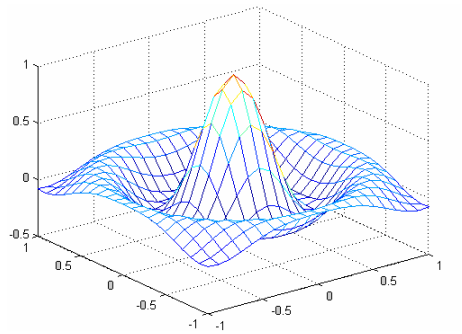
روشLM علاوه بر همگرایی سریعتر و شبیه‌سازی بهتر نسبت به CG، امتیاز دیگری نیز دارد. در مسائلی که روش‌های گرادیان توام و کوشی- نیوتن اصلاً به جواب نمی‌رسند و همگرا نمی‌شوند، شبکه LM با حداقل تعداد نرون و حداقل تعداد تکرار به راحتی همگرا می‌شود. به عنوان مثال تخمین تابع پله با شبکه‌ای با ساختار ۱-۵-۱، با روش‌های CG وQN غیر ممکن است. (الگوریتم به حداقل مقدار گام[۱۲] می‌رسد، در صورتی که خطا هنوز به مقدار $goal=0.01$ نرسیده است.) روش LM با ۵ نرون در لایه میانی طبق شکل (۸) به سرعت همگرا می‌گردد.



شکل ۸- همگرایی *LM* در تخمین پله با ۵ نرون در لایه میانی در مقایسه با همگرا نشدن *CG* در همین شرایط
(مجموع مربعات خطا بازای هر *epoch*)

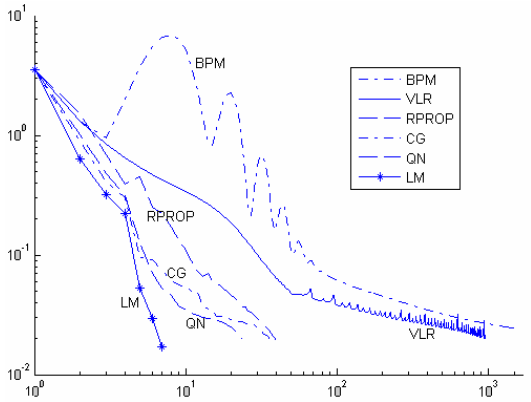
مسئله ۳- تخمین تابع sinc سه بُعدی

شبکه‌ای ۱-۱۵-۲ با تابع انتقال *tansig* در لایه مخفی و خطی در لایه آخر، برای تخمین تابع *sinc* (شکل ۹) با دو ورودی در بازه [۱ ۱-] در نظر گرفته شد. این مسئله شامل ۴۴۰ جفت ورودی/خروجی داده آموزشی است.



شکل ۹- نمودار تابع *sinc*

نمودارهای همگرایی شبکه‌های مرتبه اول و دوم در تخمین تابع این مسئله تا رسیدن به خطای ۰/۰۲ در شکل (۱۰) نشان داده شده است. توجه کنید که سرعت همگرایی LM بسیار چشمگیر است. در این آزمایش، زمان همگرایی روش‌های BPM و VLR بیش از ۱۰۰ برابر روش LM دیده شده است.



شکل ۱۰- همگرایی شبکه‌ها در تخمین تابع *sinc*

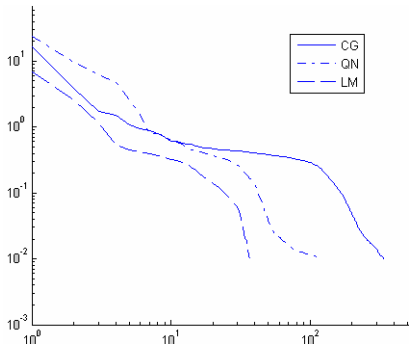
مسئله ۴- تابعی ۴ ورودی

شبکه‌ای با ساختار ۱-۵۰-۴ با تابع انتقال *tansig* در لایه مخفی و خطی در لایه آخر برای تخمین تابع

$$y = \sin(2\pi . x_1) x_2^2 x_3^3 x_4^4 \exp(-(x_1 + x_2 + x_3 + x_4))$$

با چهار ورودی در بازه [۱ ۱-] در نظر گرفته شد. این مسئله شامل ۶۲۵ جفت ورودی/خروجی داده آموزشی است. همگرایی روش‌های مرتبه دوم در تخمین این تابع تا رسیدن به خطای ۰/۰۱ در ادامه نشان داده شده است. از روش‌های مرتبه اول به علت زمان بسیار طولانی همگرایی در این مسئله استفاده نشد. شکل (۱۱) نمودار

همگرایی سه شبکه‌ی گرادیانی مرتبه دوم در تخمین این تابع را نشان می‌دهد.



شکل ۱۱- همگرایی شبکه‌های مرتبه دوم برای تخمین تابعی ۴ ورودی

روش LM همچنان سریعترین همگرایی را دارد. با مقایسه اشکال ۱۰ و ۱۱، درمی‌یابیم که پیچیدگی مسئله، زمان همگرایی را در LM تقریباً ۵ برابر ولی در CG تقریباً ۱۰ برابر افزایش داده است.

۵- جمع‌بندی و نتیجه‌گیری

مقایسه کلی دو دسته روش‌های گرادیان مرتبه اول و دوم در جدول ۱ نشان داده شده است. در بین روش‌های گرادیان مرتبه اول بکارگرفته شده در این مقاله، همگرایی سریعتر در روش RPROP و در روش‌های مرتبه دوم در الگوریتم LM دیده شد. نتایج این پیاده‌سازی‌ها نشان داد که الگوریتم LM علاوه بر اینکه در بین تمام الگوریتم‌های ذکر شده، سرعت همگرایی فوق‌العاده بالایی دارد، در نتایج شبیه‌سازی نیز بهتر عمل می‌کند؛ ضمن اینکه در بسیاری از مسائل که روش‌های مرتبه دوم گرادیان توام و کوشی- نیوتن جواب نمی‌دهند، LM با حداقل تعداد نرون‌ها و با سرعت بالا به راحتی همگرا می‌گردد. جدول۲ نیز ویژگی‌های بعضی از روش‌های گرادیان مرتبه اول و دوم را نشان می‌دهد.

جدول ۱- مقایسه دو دسته روش‌های گرادیان مرتبه اول و دوم

	حسن	عیب
مرتبه اول	سادگی، حجم محاسبات و فضای کم، مناسب برای مسائلی که کانتورهای تابع هدف دایره‌ایست[۱]،	نیاز به تکرار زیاد الگوریتم یادگیری، خطر افتادن در مینیمم محلی بیشتر از مرتبه دوم، نامناسب برای مسائل بزرگ
مرتبه دوم	افزایش سرعت همگرایی، مناسب حتی در مسائلی که کانتورهای تابع هدف غیر دایره‌ای یا بیضوی است[۱]	نیاز به حجم محاسبات و فضای زیاد، مناسب برای مسائل بزرگ

جدول ۲- ویژگی‌های روش‌های گرادیان مرتبه اول و دوم

	نام روش	حسن	عیب
مرتبه اول	شیب‌دارترین نزول	به علت سادگی، در بسیاری از مسائل غیر خطی استفاده می‌شود.	تمام معایب روش‌های مبتنی بر گرادیان مرتبه اول
	با مومنتم	احتمال افتادن در دام مینیمم محلی و تغییر وزن‌های ناگهانی کمتر است.	دغدغه تنظیم پارامتر مومنتم
	RPROP	سرعت همگرایی بیشتر از دو روش قبل، سادگی محاسبات، حذف تاثیرات بد اندازه گرادیان روی اندازه گام	دغدغهٔ تنظیم پارامترهای $\eta^{+},\eta^{-},\Delta_0$ و وابستگی سرعت همگرایی به تنظیم این پارامترها
مرتبه دوم	نیوتن کلاسیک	در صورت وجود ماتریس هسین مثبت تعریف شده و معکوس‌پذیر با یک‌گام همگرایی انجام می‌شود. (همگرایی بسیار سریع)	همیشه ماتریس هسین به راحتی قابل محاسبه و معکوس‌پذیر نیست.
	کوشی- نیوتن	مناسب برای مسائلی که معکوس ماتریس هسین به راحتی قابل محاسبه نیست.	نیاز به فضا و محاسبات تکراری زیاد
	گرادیان توام	به فضای کمتری نسبت به کوشی- نیوتن نیاز دارد، انتخاب جهت حرکت همزمان با اندازه گام به گونه‌ای که دقیقا به هدف می‌زند، همگرایی در همه مسائل	نیاز به محاسبات تکراری زیاد نامناسب برای مسائل کوچک
	لونمارکوت	همگرایی سریع	حساسیت همگرایی مسئله به تنظیم پارامترها

مراجع

[1] Gupta M., Jin L. and Homma N., *Static and Dynamic Neural Networks: From Fundamental to Advanced Theory*, John Wiley & Sons Inc., 2003.

[2] Haykin S., *Neural Networks*: *A Comprehensive Foundation*, Second Edition, Prentice Hall, 2005.

[3] Hagan M., Demuth H. and Beale M., *Neural Network Design*, China Machine Press, 2002.

[4] Battiti, R., "First and second order methods for learning: Between steepest descent and Newton's method," *Neural Computation*, Vol. 4, No. 2, 1992, pp. 141-166.

[5] Bingham, *The Theory and Practice of Modem Design,* Wiley, NewYork, 1988.

[6] Le Cun, Kanter and Solla, "Second Order Properties of Error Surface: Learning Time and Generalization", *Neural Information Processing Systems,* NIPS 3, pp.918-924, Moragn Kaufmann and Can Mateo, CA, 1991.

[7] Lou, "On The Convergence of The LMS Algorithm with Adaptive Learning Rate for Linear Feed Forward Networks", *Neural Computation 3*, pp.227-245. 1991.

[8] Riedmiller M. and Braun H., "RPROP- A Fast Adaptive Learning Algorithm", *Proc. of ISCIS VII.*

[9] Bishop Ch., "Exact Calculation of the Hessian Matrix for the Multi-layer Perceptron", *Neural Computation* 4 No. 4,pp. 494 – 501, 1992.

[10] Charalambous, C., "Conjugate gradient algorithm for efficient training of artificial neural networks," *IEE Proceedings-G.*, Vol. 139, No. 3, 1992, pp. 301-310.

[11] Hagan, M.T., and M. Menhaj, "Training feed-forward networks with the Marquardt algorithm," *IEEE Trans. on Neural Networks*, Vol. 5, No. 6, 1999, pp. 989-993, 1994

[12] Suratgar A., Tavakoli M., and Hoseinabadi A., "Modified Levenberg-Marquardt Method for Neural Networks Training", *Proc. Of World Academy Of Science, Engineering And Technology*, Volume 6 June 2005.

[1] Decoupled momentum
[2] **R**esilient back **PROP**agation
[3] Adaptive Back Propagation
[4] Extended Back Propagation
[5] Least Mean Square
[6] Hessian Matrix
[7] Davidson-Fletcher-Powell
[8] Broyden-Fletcher-Goldfarb-Shannon
[9] Conjugate Gradient
[10] BPM (BackPropagation with Momentum)
[11] VLR (Variable Learning Rate)
[12] Minimum Step Size