بهبودی بر شبکه های عصبی چند لایه انتشار برگشتی با به کارگیری نرخ یادگیری متغیر و تئوری اتوماتان وتعیین نرخ یادگیری بهینه

محمدرضا جعفریان

عضو هیئت علمی دانشگاه آزاد اسلامی واحد ایلام
Avishan_sazeh@Yahoo.com

علیرضا عباس زاده

عضو هیات علمی دانشگاه آزاد اسلامی واحد ایلام
Hiva_Reza@Yahoo.Com

چکیده:

شبکه های عصبی چند لایه پیش خور از دیر باز به طور وسیعی مورد توجه محققان بوده است. این شبکه ها علی رغم موفقیت چشم گیر در برقراری ارتباط بین ورودی وخروجی، دارای چندین نقطه ضعف بوده اند. به عنوان مثال زمان آموزش این شبکه ها نسبتاً طولانی است وگاهی ممکن است این شبکه ها آموزش نبینند. دلیل طولانی بودن زمان آموزش را می توان به انتخاب نامناسب پارامترهای شبکه نسبت داد. روش به دست آوردن پارامترهای وزن وبایاس شبکه، استفاده ازگرادیان تابع انرژی شبکه می باشد. همان طور که می دانیم تابع خطای شبکه دارای سطح ناهمواری بوده لذا شبکه درنقاط بهینه محلی متوقف شده وآموزش نمی بیند. برای جبران اشکال های وارد به الگوریتم بازگشتی، جهت بالا بردن سرعت آموزش از نرخ یادگیری متغیر تطبیقی و برای جلوگیری از به دام افتادن شبکه در نقاط بهینه محلی از روش الگوریتم اتوماتان استفاده می نماییم. با استفاده از این روش ها می توان نرخ یادگیری بهینه برای شبکه های مختلف به دست آورد.

کلمات کلیدی:

شبکه های عصبی چند لایه،الگوریتم انتشار برگشتی، نرخ یادگیری متغیر، تئوری اتوماتان

۱- مقدمه:

شبکه های عصبی از عناصر ساده عملیاتی تشکیل شده اند، این عناصر در پردازش اطلاعات در شبکه ذاتاً به صورت موازی عمل می کنند. روش کار این عناصر بر گرفته از سلول های عصبی بیولوژیکی می باشد. بطور طبیعی ارتباط های بین عناصر، وظیفه شبکه را تعیین می کنند.

با تنظیم ارتباط های بین عناصر (وزن های ارتباطی و ترم های بایاس) می توان شبکه را در اجرای یک وظیفه خاص آموزش داد. بنابراین شبکه آموزش دیده در مقابل یک ورودی خاص، یک خروجی خاص را پاسخ می دهند. شبکه با مقایسه مقدار خروجی تولید شده (بر مبنای ورودی اعمال شده به آن) و مقدار خروجی واقعی، اقدام به تنظیم کردن پارامترهای خود می کند به طوریکه در هر بار تنظیم پاسخ شبکه به مقدار واقعی نزدیک شود. البته این فرآیند در قالب یک روش یادگیری مناسب انجام می گیرد.

ویلیام جیمز در سال ۱۸۹۰ ایده های بسیار مهمی در باره روانشناسی و سازمان مغز آدمی، همچون فعالیت نرونی،

وزن های اتصال و پردازش موازی ارائه داد.

در سال ۱۹۵۸،فرانک روزن بلات نخستین شبکه عصبی را شبیه سازی کرد وآن را پرسپترون نامید. انتشار برگشتی تاریخچه جالبی دارد. رامل هارت، هینتون و ویلیامز در سال۱۹۸۶شرح مختصر واضحی از الگوریتم برگشتی را بیان داشتند. بعد از انتشار نتایج آنها معلوم شد که پارکر در سال ۱۹۸۲ کارهای رامل هارت را قبلاً پیش بینی کرده بود.کمی بعد از این معلوم شد که ورباس در سال ۱۹۷۴ این روش را بیان نموده است. در سال۲۰۰۱ میبدی و بیگی الگوریتم مومنتم تطبیقی را ارائه نمودند

همانند سایر رشته های علوم و مهندسی، از شبکه عصبی در پروژه های مختلف مهندسی برق و عمران نیز بهره گیری شده است. به عنوان مثال: در مهندسی برق: پردازش سیگنالها[3]، شناسایی دینامیک و مدل سازی و شناسایی عیب در بویلرها و ژنراتور پیش بینی میزان برق مصرفی و مورد نیاز کوتاه و متوسط و دراز مدت، برنامه ریزی، بهره برداری و نگهداری نیروگاهها. در مهندسی عمران: مدل سازی بتن، خاک، مواد پلاستیک و کنترل سازه ها در برابر بارهای زلزله و باد، بهینه سازی شناخت سیستم و پارامترهای ساختمان، تشخیص منابع آب، تهیه هیدروگراف واحد برای کنترل سیلابها و بسیاری موارد دیگر.

۲- شبکه عصبی چند لایه ای پیش خور:[4]

شبکه های عصبی چند لایه ای پیش خور که به اختصار *MLFN* خوانده می شوند، از معمول ترین انواع شبکه های عصبی می باشند. همان طور که در شکل(۱) نمایش داده شده، یک *MLFN* از تعدادی واحدهای پردازنده اطلاعات تشکیل یافته که به هم متصل هستند اما نحوه اتصال آنها به گونه ای است که واحدها در لایه هایی قرار داده می شوند و در ضمن هیچ واحدی اطلاعات را به لایه های پیشین یا به لایه خود نمی فرستد. این ساختار ساده، کاربرد *MLFN* ها را برای مهندسین آسان تر ساخته است.

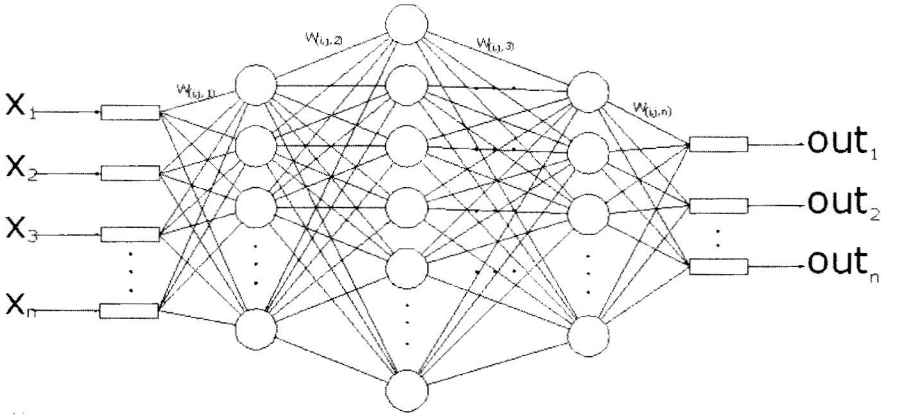
ساختار این شبکه دارای آرایش استاتیکی بوده و نیز می توان از آن برای کنترل سیستمهای دینامیک استفاده کرد. برای این منظور لازم است تمام اطلاعاتی که دینامیک سیستم را شامل می شوند در هر لحظه به کنترل های عصبی با ساختار *MLFN* داده شود[۱]

۳- مدل نرونی مک کالو - پیتس

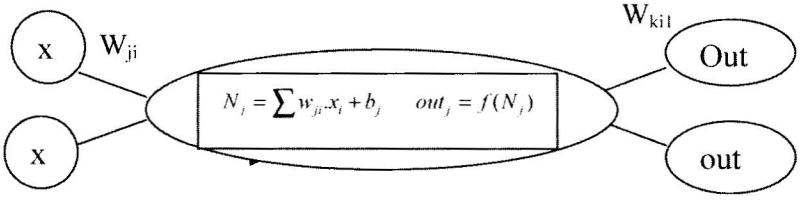
مدل مک کالو و پیتس برای توضیح رفتارسلول های عصبی طبیعی(نرون) ارائه شد. این مدل درسال ۱۹۴۳ پیشنهاد شد و دربرگیرنده اطلاعات اساسی و مهمی برای ساخت شبکه های عصبی بود. درشکل(۲) یک نرون پیشنهادی دیده می شود.

3 -Signal Prossesing

4 -Multi-Layer Feed-forward Neural Network



شکل(۱):شبکه عصبی پیش خور $MLFN$



شکل ۲: مدل نرونی مک کالووپیتس

هـر نـرون بوسـیله اطلاعـات وزن داری که ابتدا به صورت رندم تعیـن مـی شـوند، بـه نـرونهای دیگـر متصـل اسـت و از طریق این اتصـالات سـیگنالهایی را از آنهـا بـه عنوان ورودی دریافت می دارد و پـس از انجام یکسری محاسبات ریاضی بر روی آنها نتیجه را بصورت خروجی به نرونهای بعدی منتقل می سازد.

همچـنان کـه در شـکل(۲) مشـاهده می شود، برای نرون شماره j ام خواهیم داشت:

(۱)

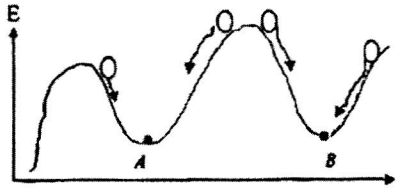
$$N_j = \sum_i^n W_{ji} X_i + b_j$$

(۲)

$$out_j = f(N_j) = \begin{cases} ۱ & if \quad N_j \geq ۰ \\ ۰ & if \quad N_j < ۰ \end{cases}$$

کـه در آن ورودیهای X_i در وزن اتصال مربوط به خود W_{ji} ضرب شـده و بـا هم جمع می شوند و سپس یک مقدار انحراف (بایاس) که مقـدار آن نـیز تعیـن می شود، به آن افزوده می شود. نتیجه این عمل کـه N_j نامـیده مـی شـود به تابع $f(N_j)$ که به نام تابع آستانه تحـریک دوگانـه۵ خوانـده مـی شـود، وارد شـده و مقدار out_j یعنی خروجیـنرون شماره‌jام محاسبه می شود. این خروجی خود ممکن است ورودی به نرونهای دیگری باشد.

بـا بررسـی تابع انرژی شبکه ها، در می یابیم که این تابع دارای حفـره هایی است که می توانند به عنوان مکان ذخیره سازی الگوهای شـبکه، مـورد اسـتفاده قـرار گـیرند. ایـن حفره ها در واقع نقاط جذب هسـتند. درشـکل (۲)نقـاطA,B جـذب بـه بـیان ریاضـی نقاط پایدار مجانبـی۶ هستند و در شبکه ها نقاطی هستند که شبکه به آن همگرا می شود [۲].



شکل۳: تابع انرژی شبکه با یک نرون که در آن $A.B$ نقاط جذبند

توصـیف هندسـی مسـاله همگرایـی را می توان مطابق شکل(۳) بدیـن صـورت بـیان نمـود: نقطه ورودی که وضعیت اولیه بردار حالت شـبکه را مشـخص مـی کند، به سمت کوچکترین ناحیه ای که شامل یکـی از حفـره های جذب (نقطه پایدار مجانبی) می باشد، نزدیک می شـود و هنگامـی که داخل محوطه قرار گرفت از هر طرف که حرکت کـند نقطـه جذب آن رابه سمت خود می کشاند و نهایتاً نقطه ورودی جـذب حفـره مـی شود و در همان جا قرار خواهد گرفت، چراکه در هر جـای دیگر در حول و حوش آن نقطه دارای سطح شیب مثبت (جهت گیری به سمت بالا) خواهد بود.

5 -Binary Threshold

6 -Asymptotically Stable Points

در ایـن قسـمت روش آمـوزش پـس انتشار خطای استاندارد ارائه مـی شود. برای این منظور ساختار شبکه n لایه در شکل(۱) را در نظر می گیریم:

ایـن سـاختار دارای n- ۱لایه پنهان[7] با S_1 نرون در لایه پنهان اول و S_1 نـرون در لایـه مخفـی دوم و S_{n-1} نـرون در لایه یکی به آخر و R ورودی و S_n خروجـی مـی باشـد. از نگـارش زیر برای توضیح عملکرد شبکه استفاده شده است:

L: تعداد لایه های شبکه است (بدون در نظر گرفتن لایه ورودی)

S_i:تعداد نرون های لایه i ام

net_j^l: کل اطلاعات وارد شده به j امین نرون، از لایه i ام

a_j^l: خروجی j امین نرون از لایه i ام

W_{ji}^l: وزن ارتباطـی از j امیـن نرون لایه (L-۱) ام به i امین نرون لایه

b_j^l: مقدار بایاس j امین نرون از لایه l ام

$f^L(*)$[8]: تـابع تحـریک نـرون هـای لایـه L که درآن i=۱,۲,..,L می باشد.

با توجه به شرح بالا برای پارامترها،کل اطلاعات وارد شده به نرون j از لایه i برابر است با:

$$net_j^l = \sum_{i=1}^{S_l} W_{ij}^l \times a_i^{l-1} + b_j^l \qquad (۳)$$

خروجی نرون ورودی همان بردار ورودی می باشد و خروجی نرون j ام از لایه i ام برابر است با:

$$a_i^0 = P_j \quad و \quad a_j^l = f^l(net_j^l) \qquad (۴)$$

تـابع معـیار یـا انـرژی شـبکه در الگوریتم پس انتشار خطای استاندارد توسط رابطه زیردرنظر گرفته می شود:

$$E = \sum_{i=1}^{L} \sum_{j=1}^{S_L} (t_j^d - a_j^l)^2 \qquad (۵)$$

که در آن t_j^d پاسخ مورد نظر(واقعی) شبکه می باشد.

(۶)

$$\delta_j^l = \frac{\partial E}{\partial a_j^l} = \frac{\partial (\sum (t_j^d - a_j^l)^2)}{\partial a_j^l} = ۲ \times (t_j^d - a_j^l), \qquad j = ۱,۲,...,S_L$$

$$\overline{\delta}^L = \overline{T} - \overline{A}$$

برای لایه مخفی داریم:

$$\delta_j^l = \frac{\partial E}{\partial a_j^l} = \sum_{k=1}^{S_{l+1}} \frac{\partial E}{\partial a_j^{l+1}} \times \frac{\partial a_k^{l+1}}{\partial a_j^l} \qquad (۷)$$

$$\delta_j^l = \sum_{k=1}^{S_{l+1}} \delta_j^{l+1} \cdot \frac{\partial a_k^{l+1}}{\partial a_j^l} \quad , \quad \delta_j^{l+1} = \frac{\partial E}{\partial a_k^{l+1}}$$

$$\frac{\partial a_k^{l+1}}{\partial a_j^l} = \frac{\partial f_k^{l+1}(net_k)}{\partial a_j^l}$$

$$\frac{\partial f_k^{l+1}}{\partial net_j^{l+1}} \times \frac{\partial net_j^{l+1}}{\partial a_j^l} \qquad a_k^l = f_k^l$$

که در آن

$$net_k^{l+1} = \sum_{i=1}^{S_l} W_{ki}^{l+1} \times a_i^l + b_k^{l+1} \qquad l = ۱,۲,۳,.....,l-۱$$

همـانطور کـه قـبلاً گفـته شد تابع f(x)یک تابع غیر خطی پیوسته و مشـتق پذیـر اسـت و در روابـط مسـیر برگشت، مشتق تابع مورد نیاز اسـت. اگـر از توابـع زیگمویـید بـه عـنوان تابع تحریک استفاده شود، می توان نوشت:

(۸)

$$f(x) = c.\frac{۱}{۱+exp(-k.x)}$$

$$\frac{\partial f(x)}{\partial x} = c.k.f(x).(۱-f(x))$$

$$f(x) = c.tanh(kx) = c.\frac{۱-exp(-۲k.x)}{۱+exp(-۲k.x)}$$

در صورتی که از تابع تانژانت هیپربولیک به عنوان تابع تحریک استفاده شود، در این صورت داریم:

(۹)

$$\frac{\partial f(x)}{\partial x} = c.k(۱-f^۲(x)) \qquad c=۱ \quad \& \quad k=۱$$

با توجه به رابطه (۷) می توان نوشت:

(۱۰)

$$\frac{\partial\, net_k^{l+1}}{\partial\, a_j^l} = W_{ij}^{l+1}$$

در نتیجه رابطه به صورت زیر در می آید:

(۱۱)

$$\delta_i^l = \sum \delta_i^{l+1} . f_k^{l+1}(۱-f_k^{l+1}) . W_{ki}^{l+1} \qquad l = ۱,۲,.....,L$$

حـال بـا مشـخص شـدن δ_i^l تمام نرونها، می توان حساسیت انرژی شبکه را نسبت به تغییر پارامترهای W_{ij}^l و b_i^l به دست آورد.

(۱۲)

$$\begin{cases} \Delta W_{ij}^l = \dfrac{\partial E}{\partial W_{ij}^l} = \dfrac{\partial E}{\partial a_i^l} \cdot \dfrac{\partial a_i^l}{\partial W_{ij}^l} \\ \Delta b_i^l = \dfrac{\partial E}{\partial b_i^l} = \dfrac{\partial E}{\partial a_i^l} \cdot \dfrac{\partial a_i^l}{\partial b_i^l} \end{cases}$$

(۱۳)

$$\begin{cases} \Delta W_{ij}^l = \dfrac{\partial E}{\partial f_i^l} \cdot \dfrac{\partial f_i^l}{\partial W_{ij}^l} . \delta_i^l \cdot \dfrac{\partial f_i^l}{\partial net_i^l} \cdot \dfrac{\partial net_i^l}{\partial W_{ij}^l} \\ \Delta b_i^l = \dfrac{\partial E}{\partial f_i^l} \cdot \dfrac{\partial f_i^l}{\partial b_i^l} . \delta_i^l \cdot \dfrac{\partial f_i^l}{\partial net_i^l} \cdot \dfrac{\partial net_i^l}{\partial b_i^l} \end{cases}$$

7 -Hidden Layer

8 -Active Function

(۱۴)

$$f_i^l(net_i^l) = a_i^l \qquad \& \qquad \frac{\partial E}{\partial f_i^l} = \delta_i^l$$

(۱۵)

$$\begin{cases} \dfrac{\partial a_i^l}{\partial W_{ij}^l} = f_i^{'l}(net_i^l).a_i^{l-1} = a_i^l(1-a_i^l).a_i^{l-1} \\ \dfrac{\partial a_i^l}{\partial b_i^l} = f_i^{'l}(net_i^l) = a_i^l(1-a_i^l) \end{cases}$$

بنابراین برای لایه آخر در حالت تابع زیگموید داریم:

(۱۶)

$$\begin{cases} \Delta W_{ij}^l = \delta_i^l.a_i^l(1-a_i^l).a_i^{l-1} \\ \Delta b_i^l = \delta_i^l.a_i^l(1-a_i^l) \\ i = 1,2,\ldots\ldots,S_l \\ j = 1,2,\ldots\ldots,S_{l-1} \end{cases}$$

حال باتغییر زیر:

(۱۷)

$$e = (t_i^l - d_i^l) \& \quad \delta^{'l} = -ef^{'l}(net^l)$$

خواهیم داشت:

(۱۸)

$$\begin{cases} \Delta W_{ji}^{l} = \delta_i^{'L}.a_i^l \\ \Delta b_i^l = \delta_i^{'L} \end{cases}$$

حال برای لایه های بعدی بااستفاده ازقانون مشتق زنجیری خواهیم داشت:

(۱۹)

$$\frac{\partial E}{\partial W_{ij}^{L-1}} = -2.(t_d^L - a_i^L).(-1).\frac{\partial a_i^L}{\partial W_{ij}^{L-1}}$$

$$= -e.f_i^{'L-1}(net_i^{L-1}).\frac{\partial(a_i^l \times W_{ij}^L + b_i^l)}{\partial W_{ij}^{L-1}}$$

$$= -e.f_i^{'L}(a_i^l \times W_{ij}^L + b_i^l).W_{ij}^L.\frac{\partial a_i^L}{\partial W_{ij}^{L-1}}$$

$$= -e.f_i^{'L}(a_i^l \times W_{ij}^L + b_i^l).W_{ij}^L.f_i^{'L-1}(a_i^{L-1} \times W_{ij}^{L-1} + b_i^{L-1}).a_i^L$$

اگرتغییرمتغیررا اعمال کنیم داریم:

(۲۰)

$$\begin{cases} \delta_i^{'L-1} = \delta_i^{'L} W_{ij}^L.f_i^{'L}(net_i^L) \\ \Delta W_{ij}^{L-1} = \delta_i^{L-1}.a_i^L \qquad i = 1,2,\ldots\ldots,L-1 \\ \Delta b_{ij}^{L-1} = \delta_i^{L-1} \end{cases}$$

حال می توان پارامترهای شبکه را به طریق زیر تصحیح نمود:

(۲۰)

$$\begin{cases} W_{ij}^{new} = W_{ij}^{old} - \alpha.\Delta W_{ij} \\ b_i^{new} = b_i^{old} - \alpha\Delta b_i \end{cases}$$

۴- قانون عام دلتا[9]:

پس از یافتن مشتق خطا نسبت به وزنهای شبکه عصبی، از روش تندترین کاهش شیب[10] جهت تغییر وزنها برای کاهش خطای تخمین شبکه عصبی به فرم زیر استفاده می شود:

(۲۱)

$$\Delta W_{ij} = -\alpha\frac{\partial E}{\partial W_{ij}}$$

$$W_{ij}^{new} = W_{ij}^{old} + \Delta W_{ij} \qquad 0<\alpha<1 \qquad (۲۲)$$

و α عدد ثابتی است که نرخ یادگیری[11] نامیده می شود. این تغییر در W_{ij} توسط رامل هارت و همکارانش ارائه شده است [۳] با توجه به کندی همگرایی روش، اگر نرخ یادگیری کوچک باشد همگرایی خیلی کند ولی اگر نرخ یادگیری بزرگ انتخاب شود شبکه حالت نوسانی خواهد داشت. جهت بهبود این روش یک مقدار اینرسی یا اندازه حرکت[12] به شبکه افزوده می شود تا شبکه در جهت مناسب حرکت کند، دراین صورت نموها به قرار زیر تصحیح می شوند.

(۲۴)

$$\Delta W_{ij}^l(K) = \beta.\Delta W_{ij}^l(K-1) + \alpha.\delta_i^{'L}(k).a_i^l$$

$$\Delta b_i^l(K) = \beta.\Delta b_i^l(K-1) + \alpha.\delta_i^{'L}(k)$$

که در آن β مومنتم نامیده می شود و رابطه بین مومنتم و نرخ یادگیری به قرار زیر است:

(۲۵)

$$\beta = \alpha(1-\alpha)$$

۵- نرخ یادگیری متغیر (VLR)[13]:

درالگوریتم .B.P استاندارد، نرخ یادگیری درطول آموزش ثابت نگه داشته می شود. عملکرد الگوریتم به انتخاب مناسب نرخ یادگیری خیلی حساس می باشد. اگرنرخ یادگیری خیلی بزرگ باشد سرعت یادگیری را بالا برده ولی دراکثرموارد سبب نوسانی شدن (ناپایدار) شدن شبکه می گردد. اگرنرخ یادگیری کوچک انتخاب شود زمان زیادی طول می کشد تا الگوریتم همگرا شود. انتخاب نرخ یادگیری بهینه قبل ازیادگیری عملی نبوده ودرحقیقت نرخ یادگیری بهینه به هنگام پروسه آموزش همچنان که الگوریتم بر روی سطح خطا حرکت می کند دایماً تغییرمی کند [۲].

اگربه طریقی بتوان نرخ یادگیری را درهنگام پروسه آموزش متغیردرنظرگرفت عملکرد الگوریتم .B.P استاندارد را می توان بهبود

9 - General Delta Rule

10 -Steepest Descent

11- Learnig Rate

12 -Momentum

13-Variable Learning Rate

بخشید. نرخ یادگیری تطبیقی سعی می کندکه نرخ یادگیری را تاآنجایی که ممکن است وسیستم ناپایدارنشده است، افزایش دهد.

نرخ یادگیری تطبیقی نیاز به تغییراتی درالگوریتم $B.P.$ استاندارد دارد. ابتدا خروجی وخطای شبکه محاسبه می شود. سپس خروجیها وخطای شبکه جدید محاسبه می شود. اگرخطای جدید ازخطای قبلی ازیک نسبت از قبل تعیین شده[14] $max-perf-inc$ بیشترشد، ازوزنها وبایاس های جدید صرفنظر می شود. علاوه براین، نرخ یادگیری کاهش داده می شود.(معمولاً با ضرب کردن درضریب $lr-dec$)[15]. درغیراین صورت وزنهای جدید ثابت نگه داشته می شوند. اگرخطای جدید ازخطای قبلی کمترباشد، نرخ یادگیری افزایش داده می شود.(باضرب در $lr-inc$)[16] [۳].

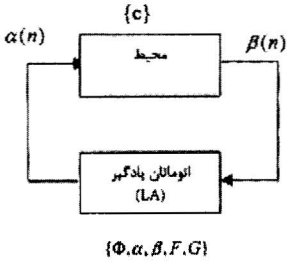
با این روش می توان نرخ یادگیری بهینه را به دست آورد. البته با این روش و با استفاده از تئوری اتوماتان که بعداً توضیح داده خواهد شد می توان نرخ یادگیری وضرایب ذکرشده فوق الذکر را بهینه کرد.

۶- تئوری اتوماتان[17] درشبکه عصبی:

هرسیستم فیزیکی یا بیولوژیکی یا هردستگاهی که بتواند یک الگوریتم کنترلی تطبیقی را محقق کند، یک سیستم اتوماتان می باشد. این بدین معناست که سیستم بتواند ورودی های کنترلی را درهرلحظه زمانی با استفاده از اطلاعات موجود واستراتژی تطبیق خاص تولید نماید. اتومااان یادگیردرمحیطی تصادفی کار می کند و درچنین محیطی تعریف می شود. این محیط با سه تایی $<\alpha,p,\beta>$مدل می شودکه با اتوماتان یادگیرتبادل اطلاعات می نماید [۴].

محیط $= E \equiv \{\alpha, p, \beta\}$

که درآن c_i, p, β, α به ترتیب مجموعه ورودی ها، مجموعه خروجی ها، مجموعه احتمالات جریمه و احتمال جریمه برای ورودی خواهد بود.



شکل۴: اتصال فیدبک اتوماتان با محیط

(۲۶)

$$\alpha = \{\alpha_1, \alpha_2, \ldots\ldots, \alpha_n\}$$
$$\beta = \{\beta_1, \beta_2, \ldots\ldots, \beta_n\}$$

$$p = \{c_1, c_2, \ldots\ldots, c_n\}$$
$$c_i = prob(\beta(k) = 1 \mid \quad \alpha(k) = \alpha_i)$$

واین یعنی اینکهc_i متناظربا α_i است ومحیط دارای پاسخ تصادفی است. مقادیرc_i نامعلوم هستند وعموماً در طول زمان یادگیری تعیین می شوند وبه همین طریق می توان احتمال پاداش را تعریف کرد که یک اقدام پاسخی مطلوب ازمحیط را به همراه دارد و $\beta = 0$ را نمایندگی می کند. دراین صورت توابع جریمه ثابت هستند و محیط را پایستارگویند. درغیراینصورت $\beta = 1$ محیط را غیرپایستارگویند.

اتوماتانهای تصادفی را می توان به ساختارثابت[18] و ساختارمتغیر[19] تقسیم بندی کرد. دراتوماتان با ساختارثابت $(FSLA)$ توابع احتمال با زمان تغییرنمی یابند. لیکن دراتوماتان با ساختارمتغیر $(VSLA)$توابع توزیع احتمالات درهرمرحله تنظیم می گردند.(بازمان تغییرمی یابند).

(۲۷)

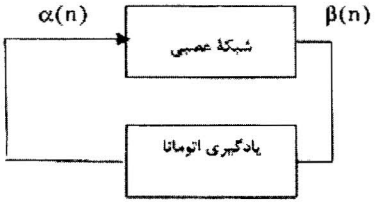
$$P(k) = \{P_1(k), \ldots\ldots, P_n(k)\}$$
$$P_i(k) = prob([\alpha(k) = \alpha_i])$$
$$\sum_{i=1}^{n} P_i(k) = 1$$

قابل توجه که احتمال انتخاب اقدامها در ابتدا مساوی است.

۷- نقش اتوماتانهای یادگیر با ساختار متغیردر آموزش شبکه عصبی

دراین قسمت روشهای مبتنی بر اتوماتونهای یادگیر برای تطبیق پارامترهای شبکه عصبی شرح داده می شود. دراین روشها شبکه عصبی بعنوان محیط عمل می کند. نحوه عمل واتصال شبکه عصبی واتوماتان درشکل (۵) نشان داده شده ست.

این الگوریتم را باختصار به عنوان $VSLA-\alpha(r)$ نشان می دهند. عبارت $VSLA$ نشان می دهدکه از اتوماتان با ساختارمتغیراستفاده شده است.



شکل۵: نحوه اتصال اتوماتان یادگیربا شبکه عصبی

αنرخ یادگیری و r تعداد اقدامها می باشد. ابتدا وزنهای شبکه عصبی و پارامترهای اتوماتان یادگیرعدد دهی اولیه می شوند. احتمال

14-Maximum Performance Increase

15-Learning Rate Decrease

16- Learning Rate Increase

17-Automata

18-Fixed Structure

19-Variable Structure

اقدامهای مختلف رادرلحظه شروع مساوی می گیریم. به طورتصادفی وبه طریقی که بعداً توضیح داده می شود، یک اقدام از مجموعه اقدامها انتخاب می شود. لازم به ذکراست بین مجموعه اقدامها ومجموعه نرخ های یادگیری تناظر یک به یک وجود دارد. شبکه عصبی به ازای اقدام انتخاب شده به اندازه m بارالگوریتم انتشاربرگشتی را اجرا می کند. حداقل مقدارخطا در m تکرارحساب شده با حداقل تعداد خطا در m بارقبلی مقایسه می شود. اگرنسبت حداقل خطا در m تکرارفعلی به حداقل خطا در m تکرارقبلی ازیک ضریبی به نام نسبت خطای ماکزیمم[20] بیشترشد، شبکه عصبی یک پاسخ نامطلوب ($\beta_{(n)}=1$) برای اتوماتان می فرستد. درغیراینصورت پاسخ شبکه عصبی به اتوماتان یک پاسخ مطلوب ($\beta_{(n)}=0$)خواهدبود. مقداراین ضریب ها درساختارشبکه متغیراست وحدوداً مقدار ۱/۰۵ انتخاب می شوند. درصورتی که پاسخ شبکه عصبی به اتوماتان نامطلوب باشد، وزنها تغییرداده نمی شود. ولی اگرپاسخ مطلوب باشد، وزنها تغییرداده می شود. اتوماتان پس ازدریافت پاسخ ازمحیط براساس الگوریتم یادگیری احتمالهای جدید اقدامها را حساب می کند. درتکراربعدی براساس احتمالهای جدید اقدام دیگری انتخاب کرده وبه شبکه عصبی اعمال می شود. همین روند به دفعات مشخص ویا تا زمانی که خطا ازیک مقدارآستانه کمترشود ادامه می یابد [۵].

۸- نحوه انتخاب یک اقدام:

درابتدای هرتکراربایستی ازمجموعه اقدامها باتوجه به توزیع احتمال آنها یک اقدام انتخاب می شود. دراینجا نحوه انجام این عمل توضیح داده می شود. برای سادگی فرض می شود اتوماتون دارای n اقدام بوده واحتمال آنها به ترتیب $p_1, p_2, \ldots\ldots, p_n$ می باشد. ابتدا ازروی توابع توزیع احتمال *(PDF)*، توابع انباشتگی[21] *(CDF)* را حساب می کنیم [۶]:

(۲۸)

$$CDF_i = \sum_{n=1}^{i} PDF(n) = \sum_{n=1}^{i} P_n$$

$$CDF_i = \begin{cases} 0 & 0 \le i < 1 \\ P_1 & 1 \le i < 2 \\ P_1 + P_2 & 2 \le i < 3 \\ P_1 + P_2 + P_3 & 3 \le i < 4 \\ . & . \\ . & . \\ P_1 + P_2 + P_3 + \ldots + P_n & i \ge n \end{cases}$$

حال برای یک اقدام از یک مولد اعداد تصادفی[22] با توزیع یکنواخت که خروجی آن بین ۰ و۱ است، استفاده می کنیم. نحوه انتخاب مطابق شکل وبه صورت زیراست.

α_1 انتخاب شده

$0 \le RN < P_1$

α_2 انتخاب شده

$P_1 \le RN < P_1 + P_2$

α_3 انتخاب شده

$P_1 + P_2 \le RN < P_1 + P_2 + P_3$

.

α_n انتخاب شده

$P_1 + P_2 + \ldots\ldots P_{n-1} \le RN < P_1 + P_2 + \ldots\ldots P_n$

نکته: احتمال انتخاب شدن اقدامهای با احتمال بیشتر، بیشتر است.

۹- تنظیم اقدام های شبکه

درشبکه های با ساختار اتوماتان متغیر، بعد از دریافت پاسخ از محیط باید با توجه پاسخ(خوب یا بد) شبکه احتمال انتخاب اقدامهای بعدی تنظیم شوند. اگر پاسخ محیط به اتوماتان خوب باشد به اقدام مورد نظر جایزه داده وگرنه اقدام مربوطه جریمه می شود. این جریمه و تشویق اتوماتان با کاهش و افزایش احتمال اقدام مورد بررسی صورت می گیرد. نحوه انجام این کار برای پاسخ مطلوب به طریقه زیر است:

(۲۹)

$$p'_j(k) = p_j(k)$$

$$p_j(k) = p_j(k) * reward$$

$$RR = \frac{1 - p'_j(k)}{1 - p_j(k)}$$

$$p_j(i) = p_j(i) * RR \qquad i \ne k$$

که در آن k اقدام مورد بررسی است. در صورت پاسخ نامناسب شبکه عصبی به اتوماتان احتمال انتخاب اقدامها به قرار زیر تنظیم می شوند:

(۳۰)

$$p'_j(k) = p_j(k)$$

$$p_j(k) = p_j(k) * reward$$

$$RR = \frac{1 - p'_j(k)}{1 - p_j(k)}$$

$$p_j(i) = p_j(i) * RR \qquad i \ne k$$

به این ترتیب احتمال های اقدام های مربوطه تنظیم می شوند وبهترین اقدام و نرخ یادگیری بهینه به دست می آید.

20-Maximum Error Ratio

21-Commulative Distribution Function

22-Random Number Generator

۱۰- پارامترهای شبکه عصبی *B.P.*

انتخاب پارامترهای شبکه عصبی در آموزش شبکه بسیار مهم است. مرحله آموزش شبکه پر زحمت ترین مرحله نگاشت شبکه می باشد. چه بسا انتخاب نامناسب پارامترها سبب همگرا نشدن شبکه می شود. از مهم ترین پارامترهای تاثیرگذار می توان به تعداد لایه و تعداد نرون های شبکه اشاره کرد. با توجه به تجربیاتی که حل مسایل به دست آمده می توان اظهار داشت که رابطه تعداد الگوها ونرونهای شبکه رابطه زیر برقرار است:

$$S_1 \times S_2 + S_2 S_3 + \ldots S_{L-1} \times S_L + S_1 + \ldots + S_L \le Q$$

or

$$\sum_{i=1}^{L-1} (S_i \times S_{i+1} + S_i) \le Q$$

که در آن Q تعداد نمونه ها، S_i ها تعداد نرونهای لایه i ام می باشند. این فرمول از قانون تعداد مجهولات کمتر از معادلات پیروی می کند. همچنین در مورد نرخ یادگیری تطبیقی می توان گفت هر چه شبکه گسترده تر باشد، نرخ یادگیری تطبیقی مقادیر کمتری تطبیق داده می شود. دوره اتوماتان را برای شبکه های پیچده بایستی مقادیر بزرگتر نسبت به شبکه های با ساختار ساده تعریف کرد. مقدار مومنتم برای مسایل مختلف عموماً عدد ۰/۹ جواب خوبی می دهد.

۱۱- مثال های عددی:

حال به بررسی مثال های حل شده توسط برنامه می پردازیم:

۱۱-۱ - مساله XOR

مرسوم است که قدرت شبکه های عصبی را توسط مثال معرفی بنامXOR با یکدیگر مقایسه می کنند. بنابراین قدرت نرم افزار نوشته شده با شبکه های دیگر مقایسه می کنیم.

در واقع هدف تخمین تابعی است که بصورت مقادیر زیر تعریف می شود. هر سطر از مجموعه فوق در واقع یک بردار آموزشی مساله است. در این مثال بردارهای آموزشی بصورت گروهی انتخاب شده اند.

جدول(۶-۷): نمایش ورودی-خروجی مثال۴

ورودی اول (x_1)	ورودی دوم (x_2)	خروجی (y)
۰	۰	۱
۰	۱	۰
۱	۰	۰
۱	۱	۱

عبارات E_p ، E_r ، F_n ، R_e ، M ، β ، α به ترتیب بیانگرتعداد تکرار مورد نظر برای شبکه، خطای مطلوب شبکه،ضریب جریمه شبکه، ضریب جایزه، تعداد تکرارها برای اتوماتان، ضریب مومنتم و نرخ یادگیری اولیه می باشند.

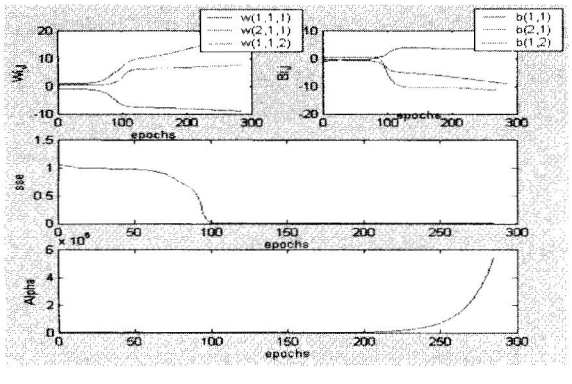
جدول(۶-۸): مقادیر اولیه پارامترهای شبکه

E_p	E_r	F_n	R_e	M	β	α
۱۰۰۰	$1e^{-8}$	۰/۹	۱/۰۴	۴۰	۰/۷۹	۰/۳۵

بابررسی شبکه های S_{2-3-1} S_{2-2-1} S_{2-4-1} و مورد مطالعه قرار دادن دوره آموزش آن متوجه می شویم که شبکه S_{2-4-1} بهترین حالت همگرایی داشته است. این شبکه دارای توابع تحریک سیگموید در لایه های اول وتابع تحریک خطی در لایه خروجی می باشد.

در ادامه بررسی شبکه به بررسی نمودار شکل (۶) می پردازیم. این شکل نتایج خروجی از شبکه و تغییرات وزنها و با یاس های شبکه را نشان می دهد. انتخاب وزنها و بایاس های انتخابی بوده و برای شفافیت شکل از چند درآیه انتخابی وزنها و بایاس ها استفاده شده است.

ملاحظه می شود که در این شکل مقادیر وزن وبایاس بعد از تعداد تکرارهای لازم به همگرایی خود دست یافته اند. چون این مثال نسبتاً برای شبکه ساده بود و همچنین دوره اتوماتان این شبکه ۴۰ فرض شده بودو در مقایسه با دوره آموزش عدد بزرگی است بدین دلیل نرخ یادگیری همواره در حال افزایش بوده و هرگز جریمه نشده است پس دیاگرام نرخ یادگیری همواره افزاینده است و شبکه نرخ یادگیری بهینه را پیدا نکرده است.



شکل۶: دیاگرام تغییرات وزن، بایاس،خطاونرخ آموزش

در ادامه بررسی این مثال، نتایج برنامه نوشته شده در این تحقیق با برنامه های دیگرکه در مرجع[۴۰] نوشته شده، در جدول زیر آمده است.

با مقایسه نتایج فهمیده می شود که برنامه الگوریتم تطبیقی با اتوماتان دارای قدرت بیشتری می باشد.

۱۱-۲- تشخیص نقاط داخل و خارج دایره ای به شعاع واحد توسط شبکه عصبی

در این مثال هدف بررسی توانایی نرم افزار نوشته شده برای برازش داده هایی با خاصیت کاملاً غیرخطی است. برای این منظور نقاط داخل دایره به شعاع واحد را با کد۱ و نقاط خارج با کد۰،کد گذاری شده است. نحوه ایجاد نمونه ها از طریق فرمول زیر بود.

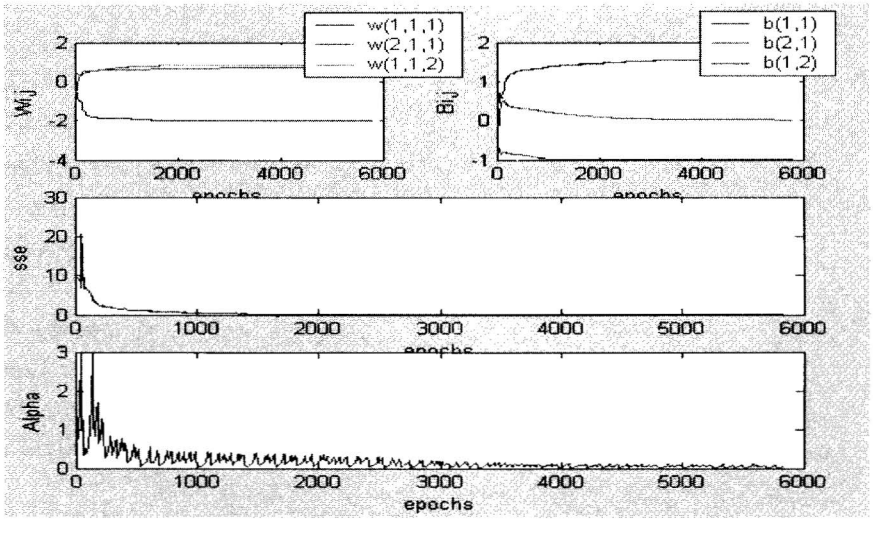
(۳۱)

$$if \begin{cases} X_2 + Y_2 \le 1 & out = 1 \\ X_2 + Y_2 > 1 & out = 0 \end{cases}$$

حال برای این مساله ۴۱ نمونه آموزشی که بردار ورودی آن دارای دو مولفه y،x و بردار خروجی آن بردار واحد است ایجاد شده است. بقیه اطلاعات شبکه مطابق با جدول نشان داده شده می باشد.

جدول(۶-۹): مقایسه نتایج خروجی الگوریتم های مختلف مشابه

ورودی اول (x_1)	ورودی دوم (x_2)	خروجی مطلوب	خروجی NN معمولی [6]	خروجی WNN [6]	خروجی الگوریتم تطبیقی با اتوماتان
۰	۱	۰	۰/۰۲	-۰/۰۴۱	۰/۰۰۰۱۴
۰	۰	۱	۰/۹۱۱	۰/۹۶۹	۰/۹۹۹۹۸
۱	۱	۱	۰/۹۹	۰/۹۴۵	۰/۹۹۹۸۷
۱	۰	۰	۰/۰۱۰	-۰/۰۰۶	۰/۰۰۰۲۱۹
			تعداد دوره آموزش ۷۰۳۷۵	تعداد دوره آموزش ۳۳۴۶۸	تعداد دوره آموزش ۳۶۵



شکل۷: دیاگرام تغییرات وزن، بایاس،خطاونرخ آموزش شبکه ۶ لایه

جدول(۶-۸): مقادیر اولیه پارامترهای شبکه برای مثال۴

E_p	E_r	F_n	R_c	M	β	α
۱۰۰۰	۰/۰۱	۰/۹	۱/۰۴	۴۰	۰/۹	۰/۴۵

بـرای حل این مساله تعداد لایه ها را بعنوان متغییر در نظر گرفته شـده و نتایج بدست آمده با هم مقایسه شده تا تأثیر تعداد لایه ها در شـبکه عصـبی بـرای ایـن مثال بررسی شود. با بررسی شبکه هـای۴،۵،۶ لایـه درمی یابیم که شبکه شش لایه بهترین برازش بین ورودی و خروجی انجام می دهد.در این شبکه از۶ لایه پنهان و یـک لایـه ورودی اسـتفاده شـده است. در لایه های ورودی تا خروجی شبکه از۱،۴،۵،۸،۳،۲ نرون استفاده شده و با $S_{۲-۳-۸-۵-۴-۱}$ نمایش می دهیم.

این شبکه، در کلیه لایه ها، به جز لایه خروجی از توابع سیگموید و لایه خروجی از تابع تحریک خطی به کار رفته شده است.

ایـن شبکه نسبت به شبکه های ذکرشده دارای دوره آموزش کمترمـی باشـد. همچنیـن با بررسی بیشتر این شبکه ها می توان گفت که نرخ یادگیری شبکه های با تعداد نرون ولایه های کمتر نسـبت بـه دیگر شبکه ها دارای نوسانات بیشتر و مقادیر بیشتری بـه آن تطبـیق داده مـی شـد. در حالیکه برای شبکه های با لایه بیشتر، عکس این مطلب صادق است.

در ادامـه بررسی شبکه ها برای مثال حاضر، شبکه بهینه را برای آموزش مجدد به کار گرفته شده و در عوض خطای مطلوب شبکه را کمـتر انـتخاب شـد. هـدف از ایـن بررسـی تغیـیرات وزنها و بایاس هاسـت.همـانطور که در شکل (۷) دیده می شود، نمودار وزن، بایاس، خطا و نرخ یادگیری در مقابل دوره آموزش رسم شده است.

بـا توجه به شکل (۷) ملاحظه می شود که وزنها و بایاس ها مقـدار مطلـوب خـود را بعـد از ۴۰۰۰ تکرار پیدا کرده و بعد از آن دیگرتغیـیرات محسوسـی در آنها دیده نمی شود. همچنین با دقت بیشـتر بـه خطـای شـبکه، ملاحظه می شود که عمل آموزش به خوبـی انجـام شـده اسـت و قسـمت آخـر شکل بیانگر تغییرات و تطبـیق یادگـیری در طول و دوره آموزش می باشد. ملاحظه می شـود کـه در طـول دوره آمـوزش هـم نـرخ یادگیری و هم اقدام مـربوط به اتوماتان شبکه بهینه شده اند. اقدام بهینه مربوطه (۹۸/ ۰، ۱/۰۲، ۱/۰۴) می باشد.

بعـد از آموزش شبکه مرحله بعدی امتحان شبکه می باشد. یعنی ببینـیم که شبکه برای هر ورودی، خروجی مطلوب را به ما می دهدیا نـه. نـتایج آزمـون شبکه در جدول (۶-۹) آورده شده است. با بررسی جدول ملاحظه می گردد که شبکه به خوبی تربیت شده است.

جدول(۶-۹): مقایسه نتایج خروجی الگوریتم های با جواب حقیقی

ورودی اول (x)	ورودی دوم (y)	خروجی مطلوب	خروجی شبکه ۵لایه
۱	۰/۵	۰	۰/۰۲۵
۱	۱	۰	۰/۰۰۰۷۸
۰/۶۷۷	۰/۶۷۷	۱	۰/۹۰۸۶
۰/۵۲	۰/۵۲	۱	۰/۹۹۹۶

مراجع:

[۱] کریمـی، هجـیر،" مطالعه رابطه مشخصه های کف فلوتاسیون مس با عملکـرد فلوتاسـیون به کمک شبکه عصبی"- پایان نامه دکتری مهندسی شیمی- دانشگاه سیستان و بلوچستان-۱۳۸۲

[۲] مـنهاج، محمـد باقـر." مبانـی شـبکه هـای عصـبی مصـنوعی ". مرکزنشردانشگاه صنعتی امیر کبیر،۱۳۸۱

[3].Rumelhart,D.E,MC Clelland,J.L."Parallel Distributed Processing, Vol.1; Foundation. MIT Press. Combridge,MA

[4].Mashoofi,M.B.Menhaj,S.A.Motamedi and M.R. Meibodi,"Adaptie VLRBP for MLP Networks Using LA" Submitted to Amir Kabir journal, Tehran-Iran(2000)

[5].Beigy, and M.R. Meibodi, M.B.Menhaj,"Adaptive of Learning Rate in Back Propgation Algorithm Using Fixed Structure Learning Automata" pro. Of ICEE-98

[۶] قربانی،امیـن"بهیـنه سـازی قابهـای صفحه ای وفضایی با محدودیت فـرکانس با استفاده از شبکه عصبی موجکها"-پایان نامه کارشناسی ارشد- دانشگاه سیستان و بلوچستان-۱۳۸۲

[7] Beigy, and M.R. Meibodi, "A note on learning automata-based schemes for adapting of BP parameters"Neurocomputing

Improving multilayer back propagation neural networks by using variable learning rate and automata theory and determining optimum learning rate

Alireza abbaszadeh
A faculty member of Islamic Azad university Ilam branch
hiva_reza@yahoo.com

Mohammad Reza jaefarian
A faculty member of Islamic Azad university Ilam branch
Avishan_sazeh@yahoo.com

Abstract:

Multilayer Bach propagation neural networks have been considered by researchers. Despite their outstanding success in managing contact between input and output , they have had several drawbacks. For example the time needed for the training of these neural networks is long, and some times not to be teachable. The reason for this long time of teaching is due to the selection unsuitable network parameters. The method for obtaining the network parameters of bias and weight is using is using gradient network energy function. As we know, network error function is of a not flat level ,so the network is stopped at some optimum local points and we have no instruction at this points . To compensate for returned algorithm drawbacks, we use adaptive variable learning rate to enhance learning rate and in order to avoiding our network trapping in local points we used automata algorithm . By using this method it is possible to obtain improved learning rate for different net works .

Key words:

Multilayer neural network, back propagation algorithm, Variable learning rate , Automata theory