آموزش بدون نظارت شبکه عصبی RBF به وسیله الگوریتم ژنتیک

محمدصادق محمدی

دانشکده فنی – دانشگاه گیلان

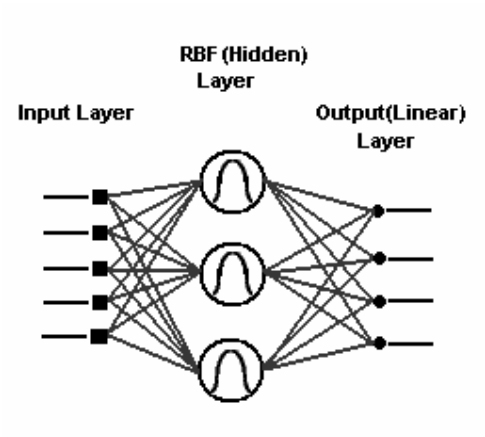
Email: m.s.mohammadi@gmail.com

چکیده – در این مقاله روشی کار آمد برای آموزش شبکه های عصبی *RBF* به کمک الگوریتم ژنتیک مطرح شده است. کارآمد ترین نوع آموزش این شبکه ها استفاده از روش های ترکیبی است. در کلیه این روش ها، یک بخش بدون نظارت وجود دارد که از آن برای تنظیم پارامتر های طبقه *RBF* این شبکه ها استفاده می شود. در مواردی که حجم داده های آموزش زیاد است، معمولاً از روش های خوشه بندی برای کاهش داده هایی که شبکه با آن ها مواجه است استفاده شود. در این مقاله می خواهیم از الگوریتم ژنتیک برای انتخاب بهینه مراکز توابع *RBF* در لایه پنهان شبکه، به کمک یک رهیافت معکوس استفاده نماییم. با توجه به این نکته که هم خوشه بندی و هم الگوریتم های تکاملی بر مبنای تکرار و صرف وقت هستند، لذا استفاده از الگوریتم ژنتیک در مورد زمان آموزش، تفاوت زیادی نمی کند؛ اما اگر بتوان خطای تست شبکه را از این طریق کاهش داد، کارایی آن بارز می شود. در پایان، نتایج با روش های دیگر خوشه بندی همچون روش *K-Means* و خوشه بندی سلسله مراتبی بر مبنای فاصله اقلیدسی مقایسه شده است. کاهش چشمگیر خطای تست شبکه حاکی از عملکرد مناسب این شیوه آموزش است.

کلید واژه- آموزش بدون نظارت، الگوریتم ژنتیک ، خوشه بندی، شبکه عصبی RBF

1- مقدمه

یکی از قدرتمند ترین شبکه های عصبی مورد استفاده در مسائل تخمین تابع، شبکه عصبی RBF است. این نوع شبکه نسبت به شبکه های عصبی پرسپترون پیشخوردی، مزایای استراتژیکی دارد [1],[12]. بر خلاف شبکه های MLP که دارای لایه های متوالی متعددی هستند، شبکه RBF از سه لایه ثابت تشکیل شده است. لایه ورودی که محل تزریق سیگنال های ورودی به شبکه است؛ لایه میانی یا طبقه RBF که شامل توابع RBF می شود، و لایه خروجی که ترکیبی خطی از کلیه خروجی های طبقه RBF را می سازد. در اکثر موارد از توابع گوسی در لایه RBF استفاده می شود که این توابع با دو پارامتر مرکز گوسی و وواریانس یا میزان گستردگی گوسی شناسایی می شوند. ساختمان یک شبکه RBF در شکل 1 مشاهده می شود. عملیاتی که در این شبکه انجام می شود در فرم ماتریسی به صورت (1) و (2) است.



شکل 1 : ساختار شبکه عصبی RBF

$$d_i(P) = \sum_{j=1}^{N} \varphi_j(P).w_{ij} \qquad (1)$$

$$\varphi_j = Exp(\frac{-1}{2\sigma_j^2}\left\|P - C_j\right\|^2) \qquad (2)$$

که در آن $d_i(P)$ خروجی i ام شبکه و φ_j خروجی گره RBF j ام به ازای بردار ورودی P است. w_{ij} نیز اتصال سیناپسی میان آن دو است. تعداد کل نورون های لایه پنهان نیز N است. بردار C_j را مرکز گوسی می نامیم و σ_j نیز مشخص کننده میزان گستردگی گوسی است.

با توجه به این ساختمان، آموزش شبکه RBF عبارت است از یافتن مقادیر مناسبی برای مراکز و گستردگی گوسی ها و همچنین وزن های لایه خطی آخر. تعداد گره های RBF در لایه میانی نیز باید مشخص شود. علاوه بر سادگی ساختار، شبکه عصبی RBF می تواند هر تابعی را با در نظر گرفتن شرایط محدودی تقریب بزند، که اهمیت آن را در مسائل تخمین تابع دو چندان می کند[12],[1]. در کاربرد های عملی این شبکه ها، حالت های متعددی پیش می آید که ما با تعداد بسیار زیادی از بردار های آموزش دهنده مواجه می شویم و مجبور هستیم به گونه ای حجم داده هایی را که برای آموزش شبکه به آن ها نیاز داریم را کاهش دهیم[4]. در اکثر مواقع از روش های ترکیبی برای آموزش این شبکه ها استفاده می شود که شامل دو بخش بدون نظارت و تحت نظارت می شود [7],[3],[2]. از بخش بدون نظارت برای تنظیم پارامتر های RBF و از بخش تحت نظارت برای تنظیم وزن های اتصالات طبقه خطی شبکه استفاده می شود. یکی از پر اسفاده ترین عملیاتی که در بخش بدون نظارت این روش ها برای فشرده سازی اطلاعات آموزش دهنده استفاده می شود، خوشه بندی الگو ها است که به کمک آن به جای استفاده از تک تک الگو های آموزش دهنده، آن ها را درون تعداد معینی خوشه در نظر می گیریم و از مختصات مرکز خوشه که برآیندی از کلیه الگوهای موجود در خوشه است استفاده می کنیم. در این صورت تعداد الگو های آموزش که ما با آن ها مواجهیم از تعداد الگوی موجود به تعداد خوشه ها کاهش می یابد. با توجه به حساسیت شبکه RBF به پارامتر هایRBF اهمیت این بخش از آموزش شبکه نیز بیشتر مورد توجه قرار می گیرد. ما ابتدا به معرفی الگوریتم ژنتیک مورد استفاده برای آموزش بدون نظارت شبکه RBF می پردازیم و سپس جزئیات روش پیشنهادی را بررسی می کنیم و در نهایت نتایج عملکرد شبکه را برای حل یک مساله تخمین تابع در دو حالت متفاوت مقایسه می کنیم. حالت اول استفاده از روش خوشه بندی و روش دوم استفاده از الگوریتم ژنتیک برای آموزش شبکه است.

2- صورت مساله

اگر از روش های ترکیبی برای آموزش شبکه RBF استفاده شود، که در اکثر مواقع چنین است[11]، در بخش بدون نظارت آموزش شبکه باید پارامتر های مربوط به لایه RBF تنظیم شوند. این پارامتر ها عبارتند از تعداد واحد های RBF در این لایه، میزان گستردگی هر یک از توابع گوسی و موقعیت مرکز هر گوسی. در روش آموزش بدون نظارت، ما با یک سری *مشاهده*، به صورت نقاطی منفرد در فضای داده ها مواجه هستیم که باید پارامتر های مطلوب را با توجه به ویژگی ها و تفاوت های اساسی در بین این مشاهدات بدست آوریم. در این مرحله، برای تنظیم پارامتر های RBF می توانیم سه استراتژی متفاوت را اتخاذ کنیم که ممکن است تنها یکی از آن ها موجب حل درست مساله گردد[1]. استراتژی اول، انتخاب تصادفی تعدادی از مشاهدات در فضای داده ها و در نظر گرفتن مختصات هر الگو به عنوان مرکز یک گوسی است. همچنین می توان از یک مقدار ثابت برای گستردگی تمام گوسی ها استفاده کرد. در روش دوم از یک عملیات تحت نظارت تصحیح خطا که معمولا از نوع نقصان گرادیان و حالت جامع تری از الگوریتم LMS است، استفاده می شود و به کمک آن مینیمم کردن مجموع مربعات تفاضل ورودی ها و خروجی ها صورت می گیرد. در روش سوم، از عملیات خوشه بندی استفاده می شود. شبکه های RBF با وجود این که در کل، عملیات تخمین تابع را به صورت جامع انجام می دهند، ولی از واحد هایی تشکیل شده اند که به صورت موضعی و محلی عمل می کنند. میزان پوشش هر واحد روی فضای داده ها توسط پارامتر گستردگی یا واریانس گوسی مورد استفاده در RBF تعیین می شود و موقعیت تحط پوشش آن، همان مختصات مرکز گوسی است. پس از انجام خوشه بندی نیز داده ها به نواحی خاصی از فضای داده تقسیم میشوند لذا به راحتی می توان پارامتر های RBF را پس از خوشه بندی، از روی موقعیت مرکز و میزان گستردگی خوشه ها به دست آورد. پس از به دست آمدن پارامتر های لایه RBF یعنی تعداد، موقعیت مرکز و میزان گستردگی هر یک از توابع گوسی، می توان وزن های لایه خطی خروجی شبکه RBF را با یک روش تحت نظارت همچون قاعده دلتا مربوط به آموزش SLP و یا

استفاده از روش شبه معکوس به دست آورد.(مرجع [1] را ببینید) ما در این مقاله نتایج آموزش به وسیله الگوریتم ژنتیک را با آموزش بدون نظارت به کمک دو روش خوشه بندی یعنی خوشه بندی K – Means و خوشه بندی سلسله مراتبی بر مبنای فاصله اقلیدسی، مقایسه کرده ایم. در هر سه رهیافت، برای آموزش وزن های لایه آخر از روش شبه معکوس استفاده شده است[2],[1] .

3- داده های مساله

برای تست نمودن روشی که ارائه خواهد شد، یک مثال تخمین تابع را در نظر می گیریم. در این مساله 100 جفت الگوی ورودی – خروجی داریم که در آن ها هر خروجی تابعی نامعلوم از بردار دو بعدی ورودی است. یعنی ورودی ها به صورت یک ماتریس 2×100 و خروجی ها به صورت یک ماتریس 1×100 است. این 100 الگو را به منظور آموزش شبکه در نظر می گیریم. به طور مشابه 100 الگوی دیگر نیز داریم که از آن ها به منظور تست کارایی شبکه آموزش دیده استفاده می کنیم.

4- خوشه بندی بهینه

فرض کنید $X_i= [x_{i1} , x_{i2} , x_{i3} , \dots ,x_{in}]^T$ یک بردار باشد و A مجموعه ای m عضوی از این بردار ها باشد. پارامتر c_j را مرکز جرم یا به اختصار مرکز خوشه S_j می نامیم وآن را به صورت زیر تعریف می کنیم:

$$c_j = ((\sum_{k=1}^{K} x_{k1}) / K,\dots,(\sum_{k=1}^{K} x_{kn}) / K) \qquad (3)$$

که در آن K تعداد بردار های موجود در خوشه S_j است. همچنین پارامتر d_{uv} را به صورت زیر در نظر می گیریم:

$$d_{uv} = D(X_u, X_v) = \sum_{k=1}^{n} (x_{uk} - x_{vk})^2 \qquad (4)$$

که مشخص کننده فاصله اقلیدسی دوبر دار X_u و X_v است. ممکن است چگالی داده ها در همه فضای داده یکسان نباشد و در بعضی نواحی بیشتر یا کمتر باشد. یکی از حالات نامساعد در انتخاب موقعیت مراکز گوسی ها در لایه RBF، وضعیتی است که مراکز گوسی روی هم یا نزدیک هم قرار می گیرند. در این صورت با وجود در نظر گرفتن دو گوسی متفاوت در این لایه، عملا از وجود یکی از آن ها (گوسی بزرگتر) استفاده می شود. در بعضی مواقع هم همپوشانی گوسی ها بیش از حد مطلوب است که در این مورد هم کارایی آن ها از بین می رود. در نتیجه، موقعیت مراکز باید به گونه ای تنظیم شود که تاحد ممکن به تمام الگو ها نزدیک باشد و در عین حال بیشترین فاصله اقلیدسی را تا مرکز خوشه های مجاور داشته باشد. در ضمن برای افزایش خاصیت تعمیم دهی شبکه عصبی، لازم است خوشه ها با هم به میزان منطقی همپوشانی نیز داشته باشند.

5- مروری بر الگوریتم ژنتیک

اساس کار الگوریتم ژنتیک، جستجوی تصادفی است. با این وجود الگوریتم ژنتیک می تواند در اکثر موارد به نزدیک جواب بهینه همگرا شود. الگوریتم ژنتیک نسبت به روش های کلاسیک بهینه سازی تفاوت هایی دارد، ازجمله این که به جای شروع از یک نقطه در مجموعه جواب ها، از چندین نقطه به طور همزمان به جستجوی پاسخ می پردازد که این باعث جلوگیری از افتادن الگوریتم در مینیمم های محلی می شود. همچنین با خود تابعی که باید مینیمم شود روبرو می شود و نیازی به مشتقات آن ندارد، لذا شامل پیچیدگی های محاسباتی، یا محاسبات اضافی نمی شود. و در نهایت این که از قواعد آماری به جای قواعد صریح استفاده می کند که این خصوصیات بر قدرت آن می افزاید.

6- تحقق الگوریتم ژنتیک

برای استفاده از الگوریتم ژنتیک باید مراحل زیر انجام شود:

- فرموله سازی مساله
- کد بندی کروموزوم ها
- تولید جمعیت اولیه
- تعریف تابع هدف
- عملیات ادغام (تقاطع) دو کروموزوم
- عملیات جهش
- انتخاب والدین برای تولید کروموزوم جدید

برای اطلاعات بیشتر به مراجع[9] و [11] مراجعه کنید. در ادامه به توضیح هر یک از موارد فوق می پردازیم.

1-6- فرموله نمودن مساله و کدگذاری

برای انجام خوشه بندی دو شیوه متفاوت را می توانیم مد نظر قرار دهیم. می توانیم مختصات خوشه ها را از روی داده ها محاسبه کرده، نقاط جدیدی را پیدا کنیم یا این که مراکز خوشه ها را از میان داده ها انتخاب کنیم. این دو روش در مرجع [8] مقایسه شده اند و ما از روش دوم استفاده می کنیم. فرض کنید مجموعه A از داده های آموزش وجود دارد که شامل m بردار می شود. هدف، انتخاب J بردار از این بردار ها است که بهترین شرایط را داشته باشند. می توانیم از کدبندی باینری[11] یا صحیح استفاده کنیم. کروموزومی به طول J در نظر می گیریم که هر ژن در آن عددی طبیعی بین 1 و m است. این عدد بیان کننده اندیس برداری است که در مجموعه A به عنوان مرکز خوشه در نظر گرفته می شود.(شکل2)

1	5	8	6	3	4	7	2	9	6

شکل 2 : کدبندی کروموزوم ها با استفاده از مقادیر صحیح

بدین ترتیب طول کروموزوم با تعداد خوشه ها یا تعداد واحد های RBF در لایه پنهان برابر است.

2-6- تابع هدف

تابع هدف، تابعی است از کروموزومی که به عنوان آرگومان به آن داده می شود. و می خواهیم مقدار آن را مینیمم کنیم. پارامتری که ما به دنبال مینیمم کردن آن هستیم، خطای تست شبکه است و استفاده مستقیم از آن در اینجا مفید است[10]. در واقع در اینجا می خواهیم به کمک تابع هدف و استفاده از روش تکاملی، با یک رهیافت معکوس، مراکز توابع گوسی را به طور خودکار و بدون خوشه بندی به سوی نقاط بهینه هدایت نماییم. با استفاده از اندیس هایی که در کروموزوم مشخص شده، مختصات مراکز گوسی ها در هر واحد RBF مشخص می شود. سپس به روش نزدیک ترین همسایه مرتبه دوم (مرجع [3])، میزان گستردگی هر یک از گوسی ها را بدست می آوریم. در نهایت وزن های طبقه آخر را به شیوه شبه معکوس پیدا می کنیم. در این مرحله شبکه آموزش دیده را با 100 الگویی که برای تست در نظر گرفته بودیم آزمایش می کنیم. مقدار تابع هدف برای این کروموزوم از رابطه زیر بدست می اید:

$$F = Error = MSE(T - Output) \qquad (5)$$

در رابطه فوق Output، خروجی شبکه به ازای ورودی های تست و T میزان اصلی این مقادیر است. تابع MSE میانگین مربع خطای این دو بردار را می دهد.

3-6- تولید جمعیت اولیه

در مرحله اول یک جمعیت P عضوی اولیه از این افراد به صورت تصادفی تولید می کنیم. تعداد اعضای جمعیت متناسب با طول کروموزوم ها است. اگر طول کروموزوم یا تعداد متغیر های مجهول مساله بیشتر شود باید جمعیت اولیه نیز بزرگتر باشد. هر یک از اعضای جمعیت یک جواب مساله است.

4-6- اوپراتور ادغام

اوپراتور ادغام را به صورت پخشی در نظر می گیریم، به این صورت که ابتدا یک ماسک باینری به طول کروموزوم به صورت تصادفی تولید می کنیم که به ترتیب مشخص می کند که فرزند کدام ژن را از کدامیک از والدین به ارث می برد. عدد صفر مشخص کننده کروموزوم اول است و عدد یک برای کروموزوم دوم استفاده می شود. برای مثال در شکل 3 ، وقتی ماسک maskX را برای والدین Parent1 و Parent2 در نظر بگیریم، حاصل OffspringX خواهد بود.

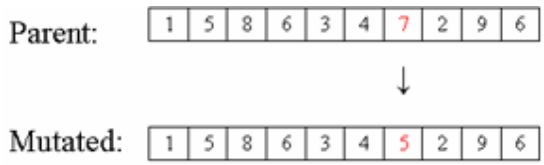
MaskX:	0	1	1	0	1	0	1	1	0	1
Parent1:	a	d	s	v	n	y	g	f	c	b
Parent2:	1	5	8	6	3	4	7	2	9	6
OffspringX:	a	5	8	v	3	y	7	2	c	6

شکل3) نحوه انجام اوپراتور ادغام(تقاطع) پخشی

5-6- اوپراتور جهش

برای پرهیز از مینیمم های محلی و همچنین گسترش

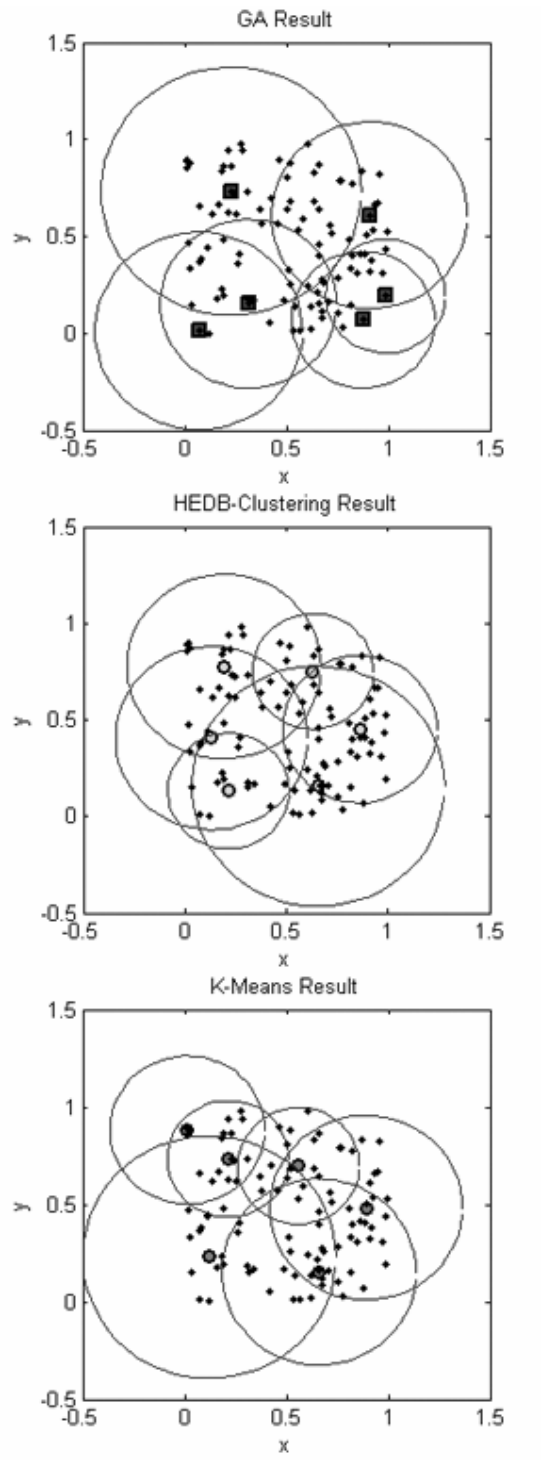
فضای جستجو در الگوریتم از اوپراتور جهش استفاده می شود. نرخ جهش نباید خیلی زیاد باشد چون موجب کندی یا واگرایی الگوریتم می شود. هر چند این مقدار کم باشد، ولی نباید آن را صفر در نظر گرفت. در اینجا اوپراتور جهش به این صورت است که در کروموزومی که برای جهش انتخاب شده، یک موقعیت به صورت تصادفی انتخاب می کنیم، سپس یک عدد تصادفی بین 1 و کل تعداد داده های موجود در مجموعه A یعنی m تولید می کنیم و ژن مورد نظر را به این مقدار جدید تغییر می دهیم.(شکل 4)



شکل 4 : اوپراتور جهش برای مقادیر صحیح

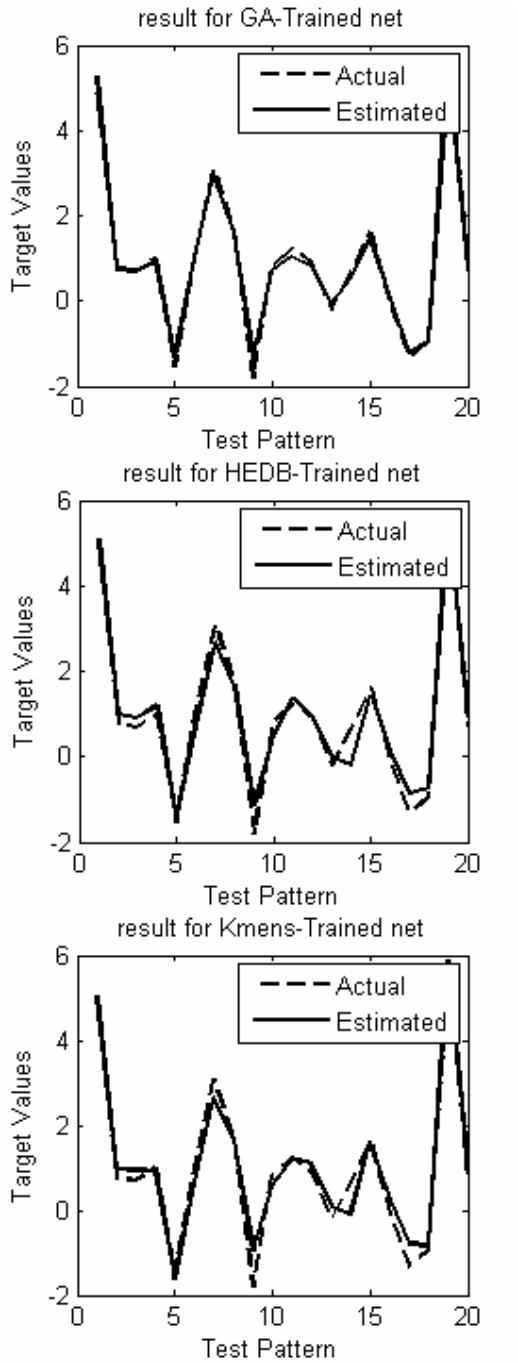
7- نتایج شبیه سازی

الگوریتم فوق به همراه دو روش خوشه بندی K-Means [1] و خوشه بندی سلسله مراتبی بر مبنای فاصله اقلیدسی(Heuristic Euclidean Distance Based Clustering)[5],[7] روی 100 الگوی ورودی – خروجی که برای آموزش شبکه در نظر گرفته بودیم پیاده سازی شد. در هر سه شبکه از الگوریتم نزدیک ترین همسایه مرتبه دوم برای محاسبه میزان گستردگی (واریانس) هر گوسی و از روش ماتریس شبه معکوس برای به دست آوردن وزن های لایه آخر استفاده شده است. سپس هر سه شبکه RBF به وسیله 100 الگوی تست که قبلا در نظر گرفته بودیم، آزمایش شدند. طول کروموزوم ها یا تعداد واحد های RBF را 6 در نظر می گیریم. در دو الگوریتم دیگر نیز با تنظیم پارامتر دقت، تعداد خوشه ها را 6 در نظر می گیریم. ما به کمک دو روش خوشه بندی که ذکر شد، موقعیت مناسب مراکز RBF را محاسبه کردیم و مختصاتی که برای مراکز به دست آوردیم، نقاط جدیدی بودند که لزوماً در بین داده های ورودی وجود نداشتند، اما در روش الگوریتم ژنتیک تعدادی از نقاط موجود در داده های ورودی را انتخاب نمودیم. مختصات مراکز خوشه ها و میزان گستردگی هر یک از آن ها، که در هر روش به دست آمده در دیاگرام 5 مشاهده می شود.



شکل 5 : الگوی پوشش روی فضای ورودی که به وسیله سه روش متفاوت به دست آمده است. در هر دیاگرام، مراکز RBF انتخاب شده به همراه میزان گستردگی آن ها نما یش داده شده است.

در شکل 6 نمودار های مربوط به تست سه شبکه دیده می شود. در هر نمودار، مقادیر اصلی و تخمینی به وسیله شبکه RBF مربوطه که با 20 الگو تست شده مشاهده می شود.



شکل 6 : مقایسه خروجی شبکه آموزش دیده با مقادیر اصلی به ازای 20 الگوی تست برای سه شبکه عصبی.

میزان خطای MSE برای 100 الگوی تست برای روش ژنتیک 0.04 ، روش خوشه بندی بر مبنای فاصله اقلیدسی 0.16 و برای روش خوشه بندی K-Means، 0.18 به دست آمد که نشان دهنده برتری روش اول می باشد.

مراجع

[1] S. Haykin, *"Neural networks: A Comprehensive Foundation"*, 2nd ed., Upper Saddle River, NJ: Prentice-Hall, 1999.

[2] علی کرمی و محمدصادق محمدی، *"پخش بار بلادرنگ با استفاده از شبکه عصبی RBF"*، بیستمین کنفرانس بین المللی برق(PSC2005) ، تهران، آبان 1384.

[3] Nicolaos B Karayiannis, Mi Glenn Weiqun, *"Growing Radial Basis Neural Networks: Merging Supervised and Unsupervised Learning with Network Growth Techniques"* IEEE Trans. on Neural Networks, Vol. 8, No. 6, 1997, pp. 1492-1506

[4] Jau-Jia Guo and Peter B. Luh, *"Selecting Input Factors for Clusters of Gaussian Radial Basis Function Networks to Improve Market Clearing Price Prediction"*, IEEE Trans. on Power systems Part B: Cybernetics, Vol. 18, No. 2, 2003, pp. 665-672.

[5] علی کرمی و محمدصادق محمدی، *"پخش بار بلادرنگ با استفاده از شبکه عصبی RBF با درنظر گرفتن ساختار متغیر سیستم قدرت"*، چهاردهمین کنفرانس مهندسی برق ایران، تهران، اردیبهشت 1385.

[6] Qin Ding and Jim Gasvoda, *"A Genetic Algorithm for Clustering on Image Data"*, International Journal of Computational Intelligence Vol. 1 No. 1 2004

[7] Y.-H. Pao, Sobajic, *"Combined Use of Unsupervised and Supervised Learning for Dynamic Security Assessment"*, IEEE Trans. on Power Systems, Vol. 7, No. 2, 1992, pp. 878-884.

[8] udmila I. Kuncheva, J. C. Bezdek *"Nearest Prototype Classification: Clustering, Genetic Algorithms, or Random Search?"* IEEE Trans. on Systems, MAN, and Cybernetics - Part C.Vol. 28, No. 1,1998.

[9] D.E. Goldberg , *"Genetic algorithms in search, optimization and machine learning"*, Addison-Wesley, Reading, MA, 1989.

[10] B.A. Whitehead, T.D. Choate, *"Evolving Space-Filling Curves to Distribute Radial Basis Functions Over an Input Space"*. IEEE Trans. on Neural Networks Vol. 5, No. 1, 1994, pp.15–23.

[11] X. Yao, "Evolving artificial neural networks," Proc. IEEE, Vol. 87, No.9, pp. 1423–1447, Sept. 1999.

[12] C. Christodoulou, M. Georgiopoulos, *"Applications of Neural networks in Electromagnetics"*, Artech House Publishers, 2001.