

فصلنامه پژوهش‌های اقتصادی ایران / شماره ۱۲ / پاییز ۱۳۸۱

## کاربرد شبکه‌های عصبی در پیش‌بینی سری‌های زمانی

محمد رضا اصغری اسکویی\*

تاریخ ارسال: ۸۱/۷/۸ تاریخ پذیرش: ۸۱/۱۰/۱۸

### چکیده

استفاده از روش‌های غیر کلاسیک در شناسایی مدل و پیش‌بینی رفتار سیستم‌های پیچیده، مدت‌هاست در محافل علمی و حتی حرفه‌ای متداول و معمول شده است. در بسیاری از سیستم‌های پیچیده و خصوصاً غیر خطی که مدل‌سازی و به دنبال آن پیش‌بینی و کنترل آنها از طریق روش‌های کلاسیک و تحلیلی امری بسیار دشوار و حتی بعضاً غیر ممکن می‌نماید، از روش‌های غیر کلاسیک که از ویژگی‌هایی همچون هوشمندی، مبتنی بر معرفت و خبرگی برخوردار هستند، استفاده می‌شود. شبکه‌های عصبی، یکی از این روش‌های بدیع و در حال تحول است که در موضوعات متنوعی از قبیل الگوسازی، شناخت الگو، خوشه‌بندی و پیش‌بینی به کار رفته و نتایج مفیدی داشته است. در این مقاله، از شبکه‌های عصبی در پیش‌بینی سری‌های زمانی داده‌های اقتصادی استفاده کرده ایم. در این رابطه عوامل مختلف ساختاری، روش‌های مختلف یادگیری شبکه‌های عصبی و انتخاب و کاربرد مناسب داده‌ها در فرایند پیش‌بینی، مورد ارزیابی و بررسی قرار گرفته است. در این پژوهش، از ابزارهای محاسباتی نرم افزار *MATLAB* و داده‌های اقتصادی کشور استفاده شده است.

واژه‌های کلیدی: شبکه‌های عصبی، سری زمانی، نرم افزار *MATLAB*

\* عضو هیئت علمی گروه آمار، ریاضی و کامپیوتر دانشگاه علامه طباطبائی

Email: [oskoei@atu-economics.ac.ir](mailto:oskoei@atu-economics.ac.ir)

### ۱. شبکه‌های عصبی

گرچه هنوز بیش از ۵۰ سال از تولد روش‌های محاسباتی مبتنی بر شبکه‌های عصبی مصنوعی نمی‌گذرد، لکن این شبکه‌ها به دلیل ویژگی‌هایی همچون پردازش موازی، هوشمندی و انعطاف‌پذیری جایگاه چشم‌گیری در مسائل پیچیده از قبیل شناخت الگو، خوشه‌بندی، مدل‌سازی، تخمین و شناسایی و پیش‌بینی برای خود باز کرده است.

شبکه عصبی در کاربرد از جهاتی همانند یک «ذهن زنده» عمل می‌کند. به این معنا که از مشاهدات انتزاعی خود به قضاوت می‌پردازد. لذا، شبکه عصبی مدتی را صرف آموزش کرده و سپس به صورت عملیاتی به کار گرفته می‌شود.

در آموزش شبکه عصبی، هر چه مشاهدات کامل‌تر باشد آنچه انتزاع شده است، صحیح‌تر خواهد بود. البته، این احتمال وجود دارد که برخی مشاهدات گمراه‌کننده بوده و با روش کلی مشاهدات هماهنگ نباشد. بنابراین، آنچه که به عنوان نمونه‌های آموزشی در اختیار شبکه عصبی قرار می‌گیرد، تا حد امکان باید پالایش شده و همسان باشند. شبکه عصبی، آنچه را مشاهده می‌کند در قالب پارامترهای درونی خود به خاطر می‌سپارد. در واقع، تکرار هر یک از مشاهدات موجب تغییر پارامترهای درونی شبکه در جهت حفظ روابط حاکم بر مشاهدات است. آنچه در ذهن شبکه عصبی نگهداری می‌شود، نه تک تک مشاهدات بلکه روش و برداشت کلی از مشاهدات است. به همین دلیل است که بعضاً شبکه عصبی در رویارویی مجدد با نمونه‌های آموزشی، چه بسا همراه با خطای قابل اغماضی عکس‌العمل نشان دهد، اما، این استواری و ثبات در عمل را دارد که در برخورد با عموم نمونه‌های مشابه، عملکردی مناسب و همراه با خطای قابل اغماض داشته باشد.

مشاهدات یا همان نمونه‌های آموزشی شبکه عصبی، می‌تواند همراه با یک پیش‌قضاوت اولیه و یا بدون قضاوت اولیه باشد. به عبارتی، آموزش شبکه عصبی می‌تواند همراه با معلم و یا بدون

1. Artificial Neural Nets-ANN
2. Pattern Recognition
3. Clustering
4. Modeling
5. Estimation & Identification
6. Forecasting
7. Supervised Learning

معلم باشد. در آموزش همراه با معلم آنچه به شبکه به عنوان مجموعه آموزشی، آموزش داده می‌شود، همراه با قضاوتی است که معلم انتظار دارد، لذا، نمونه‌ها همراه با قضاوت از پیش تعیین شده آموزش داده می‌شود تا در آینده شبکه در صورت برخورد با نمونه‌های جدید با توجه به روالی که آموزشی دیده است، عمل کند. در مواردی نیز، نمونه‌ها بدون قضاوت اولیه در اختیار شبکه عصبی قرار می‌گیرد تا خود با مشاهدات پیاپی به دسته بندی و نهایتاً انتزاع کلی از آنها اقدام کند.

شبکه عصبی، از خصوصیات درونی برخوردار است که این خصوصیات نشان دهنده استعداد و گنجایش آن است. همان‌گونه که یک ذهن زنده در برخی موارد بسیار مستعد، پیشرو و موفق و در برخی موارد، بسیار کند و ناموفق عمل می‌کند، در ابعاد تنزل یافته، شبکه عصبی نیز بسته به ساختار درونیش در برخورد با مسائل مختلف، عملکرد متفاوتی دارد. یک شبکه عصبی چه بسا در حل یک مسئله بسیار خوب و در حل مسئله ای دیگر بسیار بد عمل کند. لذا، انتخاب ساختار شبکه، متناسب با موضوع مسئله از اهمیت بسیار بالایی برخوردار است.

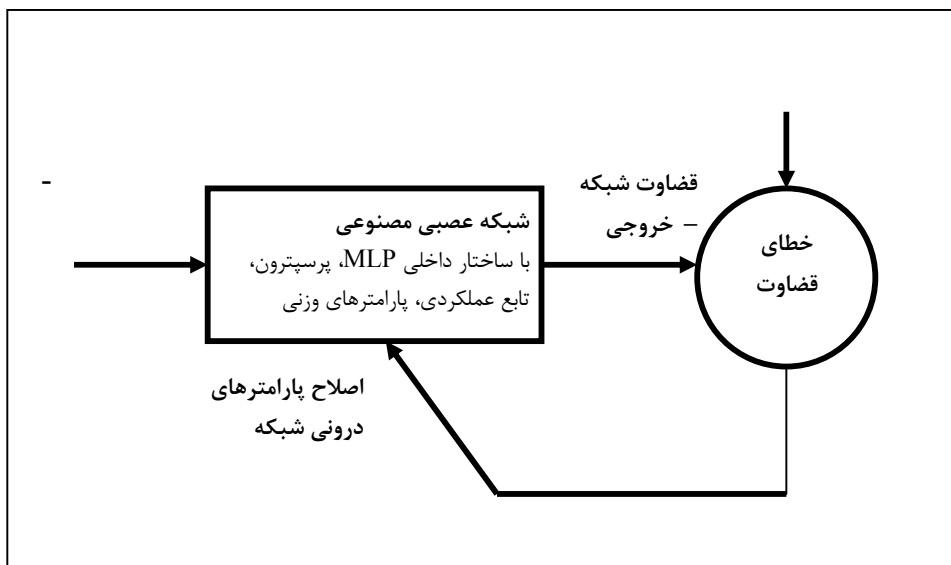
همان‌گونه که شرایط و موقعیت نیز در نتیجه آموزش تأثیرگذار است، میزان یادگیری شبکه عصبی نیز به شدت از شرایط اولیه آغاز آموزش شبکه تأثیر می‌پذیرد. انتخاب مناسب مقادیر اولیه پارامترهای شبکه در نتیجه آموزش آن، بسیار مؤثر خواهد بود.

یک شبکه عصبی پس از مدتی کاربرد، نیاز به آموزش‌های مجدد برای تعمیق و گسترش دانایی خود دارد. آموزش چنانچه شامل نمونه‌های جدید و یا نمونه‌هایی که قبلاً خوب آموزش داده نشده است، باشد، بسیار مؤثر خواهد بود.

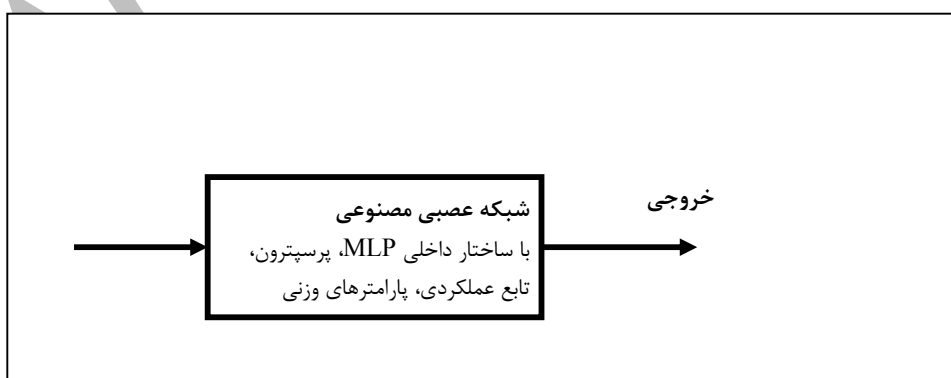
---

## 1. Unsupervised Learning

شکل - ۱. شبکه عصبی در حال یادگیری (Training Mode)



شکل - ۲. شبکه عصبی در حال به کارگیری (Operating Mode)



شبکه‌های عصبی از جنبه‌های توپولوژی، ساختاری و روش‌های یادگیری به انواع مختلفی تقسیم می‌شوند و هر یک در کاربردهای خاصی عملکرد مناسبی از خود نشان می‌دهند. شبکه عصبی چند لایه پرسپترون با روش یادگیری انتشار وارون یکی از متداول‌ترین شبکه‌های کاربردی است، که در این مقاله، صرفاً بررسی و مطالعه این شبکه مد نظر است. در مباحث نظری اثبات شده که شبکه MLP در صورت انتخاب صحیح ساختار مناسب داخلی، قادر است هرگونه سیستم غیر خطی را مدل کرده و شبیه‌سازی کند.

ساختار شبکه عصبی شامل تعدادی پرسپترون با تابع عملکردی مشخص است که در لایه‌های مجزا قرار دارند. هر پرسپترون، به واسطه ضرایب وزنی خود، خروجی تمامی پرسپترون‌های لایه قبلی را تجمیع کرده و از طریق تابع عملکردی به لایه بعدی ارسال می‌کند. شبکه عصبی دارای یک لایه ورودی، یک لایه خروجی و حداقل یک لایه پنهانی است، تعداد پرسپترون در هر لایه متفاوت و بستگی به ساختار شبکه و مسئله دارد.

الگوریتم‌های محاسباتی یادگیری انتشار وارون (Back Propagation)، بسیار متنوع و با نتایج و عملکردهای متفاوت هستند. در ساده‌ترین الگوریتم، ضرایب وزنی شبکه در جهت حداقل کردن تابع هدف شبکه (که گرادیان خطای قضاوت خروجی شبکه است) تغییر می‌کنند. لذا، در هر مرحله آموزش ضرایب وزنی به قرار زیر تغییر می‌یابد.

$$w_{k+1} = w_k - a_k g_k$$

$w_k$  بردار ضرایب وزنی شبکه،  $g_k$  گرادیان خطای خروجی شبکه و  $a_k$  ضریب یادگیری شبکه است. این روش، که به اصطلاح الگوریتم کاهش گرادیان نامیده می‌شود، به دو صورت افزایشی و یا بسته‌ای قابل

1. Multi Layer Perceptron (MLP)
2. Back Propagation

۳. رجوع کنید به منابع ۴، ۵ و ۷

4. Activity Function
5. Weight
6. Input Layer
7. Output Layer
8. Hidden Layer
9. Gradient Descent Algorithm
10. Incremental Mode
11. Batch Mode

پیاده‌سازی است. در حالت افزایشی، گرادیان خطا پس از هر بار مشاهده یکی از نمونه‌های آموزشی محاسبه و ضرایب وزنی بهبود داده می‌شود. در حالت بسته‌ای، این کار پس از مشاهده یک دوره کامل نمونه‌های آموزشی و محاسبه گرادیان کل انجام می‌گیرد. در اکثر موارد، روش افزایشی عملکرد بهتری نسبت به روش بسته‌ای دارد.

در کنار دو روش یادشده، روش‌های محاسباتی نوینی با دو هدف افزایش سرعت محاسبات و کاهش حجم محاسبات نیز ارائه شده است. این روش‌ها با عنوان روش‌های یادگیری سریع معرفی می‌شوند. روش‌های سریع خود به دو گروه روش‌های مبتنی بر تکنیک‌های انتزاعی و یا مبتنی بر تکنیک‌های بهینه‌سازی عددی تقسیم می‌شوند. در روش‌های مبتنی بر تکنیک‌های انتزاعی، ضریب یادگیری در طی مراحل آموزش متغیر است، این ضریب، در طی گام‌های آموزش کوچک و کوچکتر می‌شود تا حداقل گرادیان خطا در زمان کمتری به دست آید. در روش‌های مبتنی بر تکنیک‌های بهینه‌سازی عددی، با بهره‌گیری از شیوه‌های محاسباتی همچون *Quasi Newton*، *Conjugate Gradient* و یا روش نوین *Levenberg-Marquardt* سرعت محاسبات رسیدن به حداقل گرادیان خطا افزایش و درعین حال حجم محاسبات و حافظه مورد نیاز در محاسبات کاهش می‌یابد.

ضمناً، برخی روش‌های محاسباتی آموزش انتشار وارون نیز با هدف رسیدن به حداکثر میزان یادگیری با توجه ویژه به تعمیم‌پذیری شبکه در مواجهه با نمونه‌های جدید ارائه شده است که روش‌های بهبود یافته تعمیم‌پذیری شبکه نامیده می‌شوند. در این روش‌ها از آنجا که هدف دستیابی به حداکثر قدرت بسط و تعمیم حوزه یادگیری شبکه در برخورد با نمونه‌های جدید است، شبکه عصبی مجموعه آموزشی را به دو زیر مجموعه آموزش و تست آموزش تفکیک می‌کند. شبکه در پایان هر گام آموزش، بلافاصله عملکرد شبکه در مورد زیر مجموعه تست آموزش را اندازه‌گیری کرده و آموزش تا رسیدن به یک عملکرد مناسب در مورد نمونه‌های تست ادامه می‌یابد.

از آنجا که معرفی انواع روش‌های آموزش شبکه و ویژگی‌های هر یک خارج از موضوع این نوشتار است، علاقه‌مندان را به پیوست این مقاله و برای آشنایی بیشتر، به منبع شماره (۳) ارجاع می‌دهیم.

1. Faster Training
2. Heuristic Techniques
3. Numerical Optimization Techniques

## ۲. سری زمانی

روزانه سؤالات بسیاری از رفتارهای آتی پدیده‌های پیرامون ما مطرح می‌شود، که پاسخ به آنها بستگی به دانایی و آگاهی کافی از سازوکار رویداد آن پدیده‌ها دارد. پیش‌بینی وضع هوا، پیش‌بینی نرخ سهام، پیش‌بینی قیمت نفت جهانی و دهها مورد مشابه از جمله سؤالات مورد علاقه ما است. در یک نگاه کلاسیک علمی، برای پاسخ دادن به هر یک از موارد یادشده، لازم است از سازوکار رخ داد هر یک از پدیده‌ها و تأثیر هر عامل در رخ داد آن به صورت تحلیلی اطلاع کافی داشته باشیم. پر واضح است به دست آوردن این سازوکارها در بسیاری از موارد اگر غیر ممکن نباشد، بسیار دشوار و دست‌نیافتنی است.

در یک نگاه متفاوت، چنانچه، هر یک از رخ داده‌ها به صورت یک دنباله در قالب اعداد و کمیت، مورد بررسی قرار گیرد، می‌تواند راهکاری برای پیش‌بینی مقادیر آتی آن باشد. این دنباله‌ها، صرفنظر از اینکه مربوط به چه پدیده و برخاسته از چه سازوکار و متأثر از چه عواملی باشند، به نام «سری زمانی» تحلیل و آنالیز می‌شوند. البته، ناگفته‌نماند که بسیاری از مفروضات و اطلاعات موجود در مورد یک پدیده، می‌تواند در آنالیز سری زمانی مربوط مورد استفاده قرار گیرد. در یک سری زمانی متشکل از  $n$  نمونه

$$X_1, X_2, X_3, \dots, X_i, \dots, X_n$$

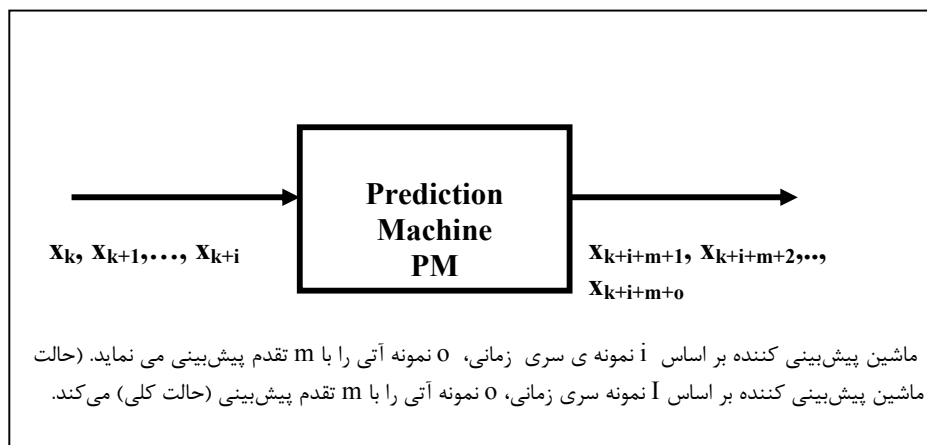
مقادیر آتی، تابعی از مقادیر قبلی خود است.

$$X_k = f(X_{k-1}, X_{k-2}, \dots, X_{k-p})$$

در مدل‌های خطی سری زمانی همچون  $AR$ ،  $MA$ ،  $ARMA$  و  $ARMAX$  روش‌های متنوع کلاسیک «تخمین و شناسایی» وجود دارد که می‌توان مقادیر آتی سری زمانی را بر اساس هر یک از مدل‌های گفته شده مبتنی بر مقادیر قبلی محاسبه کرد. لکن، پیش‌بینی سری‌های زمانی که از مدل غیر خطی پیروی می‌کنند، نیازمند ابزارهای هوشمند و پیشرفته‌ای مانند شبکه‌های عصبی است. به طور خلاصه، هدف، یافتن یک «ماشین پیش‌بینی کننده» است که می‌تواند با مشاهده مقادیری از یک سری زمانی، مقادیر آتی آن را پیش‌بینی کند.

## 1. Prediction Machine-PM

شکل ۳- ماشین پیش‌بینی کننده



در این مقاله، پیش‌بینی سری زمانی با تقدم صفر ( $m=0$ ) به وسیله شبکه عصبی MLP با الگوریتم یادگیری انتشار وارون از چهار روش مختلف تحلیل نمونه‌های آموزشی مورد بررسی قرار گرفته است. در هر یک از این روشها، یک ماشین پیش‌بینی (شبکه عصبی) با  $I$  ورودی و  $O$  خروجی استفاده شده است. این ماشین، قادر است پس از طی مراحل آموزش، با مشاهده  $I$  نمونه از سری زمانی،  $O$  نمونه بلافاصل آنها را پیش‌بینی کند برای آموزش ماشین (یا شبکه عصبی) یک مجموعه آموزشی با  $p$  عضو (هر عضو متشکل از  $I$  نمونه) و برای ارزیابی و آزمایش آن یک مجموعه آزمون با  $q$  عضو استفاده شده است. در هر مرحله آموزش، خروجی ماشین پیش‌بینی کننده (یا شبکه عصبی) با مقادیر واقعی نمونه‌ها مقایسه شده و ضرایب درونی شبکه بر اساس الگوریتم آموزشی مورد نظر اصلاح می‌شود. پس از پایان دوره آموزش، مجموعه آزمون به ماشین پیش‌بینی کننده داده شده و خروجی آن با مقادیر اصلی مقایسه می‌شود. نتایج مقایسه مقادیر پیش‌بینی شده به وسیله ماشین و مقادیر اصلی، مواد لازم را جهت محاسبه معیار و در نهایت، ارزیابی پیش‌بینی را فراهم می‌کند.

1. Training Set
2. Target



$X_i$  نمونه  $i$  ام سری زمانی  
 $\underline{X}_i$  نمونه  $i$  ام سری زمانی پیش‌بینی شده  
 $n$  تعداد کل نمونه‌های سری زمانی  
 $I$  تعداد ورودی‌های شبکه عصبی  
 $O$  تعداد خروجی‌های شبکه عصبی  
 $p$  تعداد عناصر مجموعه آموزشی  
 $Q$  تعداد عناصر مجموعه آزمون  
 $y_i$  نمونه  $i$  ام از سری زمانی پردازش شده  
 $y_i = h(x_i, x_{i-1})$

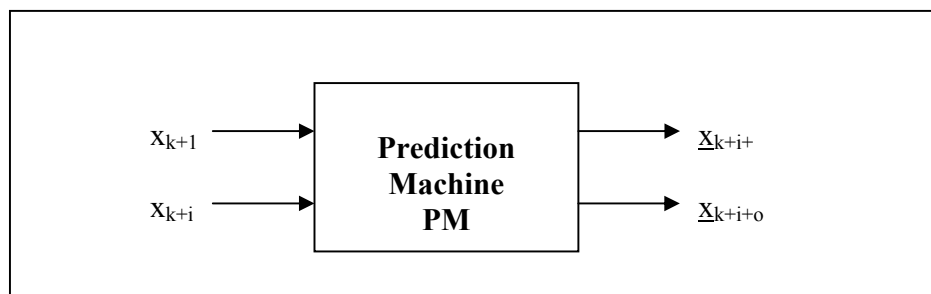
### روش اول

در این روش، مجموعه آموزشی شبکه شامل  $p$  عضو  $i$  تایی از نمونه‌های سری زمانی است. هر  $i$  نمونه تعداد  $o$  نمونه آتی را پیش‌بینی می‌کند.

Training Set نمونه‌های آموزشی شبکه	Output پیش‌بینی شبکه	Target مقادیر واقعی
$X_1, X_2, \dots, X_i$	$\underline{X}_{i+1}, \underline{X}_{i+2}, \dots, \underline{X}_{i+o}$	$X_{i+1}, X_{i+2}, \dots, X_{i+o}$
$X_2, X_3, \dots, X_{i+1}$	$\underline{X}_{i+2}, \underline{X}_{i+3}, \dots, \underline{X}_{i+o+1}$	$X_{i+2}, X_{i+3}, \dots, X_{i+o+1}$
$X_3, X_4, \dots, X_{i+2}$	$\underline{X}_{i+3}, \underline{X}_{i+4}, \dots, \underline{X}_{i+o+2}$	$X_{i+3}, X_{i+4}, \dots, X_{i+o+2}$
.....	.....	.....
$X_p, X_{p+1}, \dots, X_{p+i-1}$	$\underline{X}_{p+i}, \underline{X}_{p+i+1}, \dots, \underline{X}_{p+i+o-1}$	$X_{p+i}, X_{p+i+1}, \dots, X_{p+i+o-1}$

Test Set نمونه‌های ارزیابی شبکه	Output پیش‌بینی شبکه	Error خطا
$X_{p+1}, X_{p+2}, \dots, X_{p+i}$	$\underline{X}_{p+i+1}, \underline{X}_{p+i+2}, \dots, \underline{X}_{p+i+o}$	
$X_{p+2}, X_{p+3}, \dots, X_{p+i+1}$	$\underline{X}_{p+i+2}, \underline{X}_{p+i+3}, \dots, \underline{X}_{p+i+o+1}$	
.....	.....	
$X_{p+q}, X_{p+q+1}, \dots, X_{p+q+i-1}$	$\underline{X}_{p+q+i}, \underline{X}_{p+q+i+1}, \dots, \underline{X}_{p+q+i+o-1}$	

شکل - ۴. روش اول پیش‌بینی



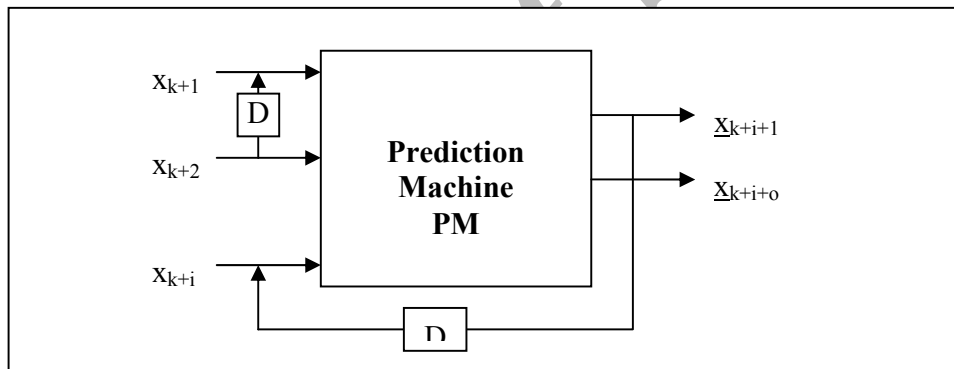
## روش دوم

در این روش، مجموعه آموزشی شبکه، همچون روش اول شامل  $p$  نمونه است. اولین عضو این مجموعه،  $i$  نمونه اول سری زمانی است. در عضو دوم، یک نمونه جدید از مجموعه پیش‌بینی شده مرحله قبلی اضافه می‌شود. این روند ادامه می‌یابد، یعنی در هر عضو جدید، یک نمونه از نمونه‌های پیش‌بینی شده قبلی اضافه می‌شود. مقادیر واقعی در آموزش شبکه، مقادیر اصلی سری زمانی است. پر واضح است در چنین روشی، خطا انباشته شده و پیش‌بینی به اندازه روش اول دقیق نخواهد بود. در واقع در این روش، آنچه رخ می‌دهد نه فقط یک پیش‌بینی، بلکه یک «تخیل» است.

Training Set نمونه‌های آموزشی شبکه	Output پیش‌بینی شبکه	Target مقادیر واقعی
$X_1, X_2, \dots, X_i$	$\underline{X}_{i+1}, \underline{X}_{i+2}, \dots, \underline{X}_{i+o}$	$X_{i+1}, X_{i+2}, \dots, X_{i+o}$
$X_2, X_3, \dots, X_i, \underline{X}_{i+1}$	$\underline{X}_{i+2}, \underline{X}_{i+3}, \dots, \underline{X}_{i+o+1}$	$X_{i+2}, X_{i+3}, \dots, X_{i+o+1}$
$X_3, X_4, \dots, X_i, \underline{X}_{i+1}, \underline{X}_{i+2}$	$\underline{X}_{i+3}, \underline{X}_{i+4}, \dots, \underline{X}_{i+o+2}$	$X_{i+3}, X_{i+4}, \dots, X_{i+o+2}$
.....	.....	.....
$\underline{X}_p, \underline{X}_{p+1}, \dots, \underline{X}_{p+i-1}$	$\underline{X}_{p+i}, \underline{X}_{p+i+1}, \dots, \underline{X}_{p+i+o-1}$	$X_{p+i}, X_{p+i+1}, \dots, X_{p+i+o-1}$

Test Set نمونه‌های ارزیابی شبکه	Output پیش‌بینی شبکه	Error خطا
$X_{p+1}, X_{p+2}, \dots, X_{p+i}$	$\underline{X}_{p+i+1}, \underline{X}_{p+i+2}, \dots, \underline{X}_{p+i+o}$	
$X_{p+2}, X_{p+3}, \dots, X_{p+i+1}, \underline{X}_{p+i+1}$	$\underline{X}_{p+i+2}, \underline{X}_{p+i+3}, \dots, \underline{X}_{p+i+o+1}$	
.....	.....	
$\underline{X}_{p+q}, \underline{X}_{p+q+1}, \dots, \underline{X}_{p+q+i-1}$	$\underline{X}_{p+q+i}, \underline{X}_{p+q+i+1}, \dots, \underline{X}_{p+q+i+o-1}$	

شکل ۵- روش دوم پیش‌بینی

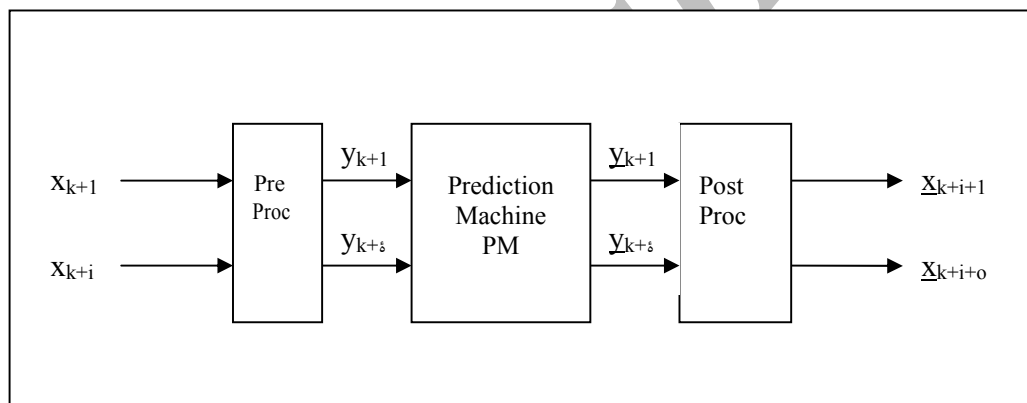


در این روش، با توجه به اینکه نمونه‌های آموزشی نتیجه پیش‌بینی مرحله قبل است، در هر مرحله آموزش، مجموعه آموزشی، مقادیر جدیدی خواهد بود، در حالی که مجموعه مقادیر واقعی همواره ثابت و برابر نمونه‌های اصلی سری زمانی است. لذا، نتایج پیش‌بینی به مقادیر اولیه خود به شدت حساس است. چنانچه، از طرح شماتیک این روش نمایان است. این روش پیش‌بینی، یک سیستم بسته است که صرفاً از یک نمونه آزمون اولیه آغاز شده و تا بی‌نهایت قادر به پیش‌بینی سلسله وار است (یعنی برای هر پیش‌بینی جدید هر بار نیاز به ورودی جدید ندارد).

## روش سوم

در این روش، از یک ماشین پیش‌بینی (یا یک شبکه عصبی) با  $i$  ورودی و  $o$  خروجی برای پیش‌بینی «پراکندگی سری زمانی» استفاده شده است. در واقع، قبل از ماشین پیش‌بینی، یک مرحله پیش‌پردازش و پس از آن یک مرحله پس‌پردازش وجود دارد.

شکل - ۶. روش سوم پیش‌بینی



همان گونه که تأکید شد، در این روش، پراکندگی سری زمانی از متوسط آن مطابق با فرمول زیر پیش‌بینی می‌شود.

$$e(x) = \sum x_k / n$$

$$y_k = x_k - e(x)$$

$$\hat{x}_k = \hat{y}_k - e(x)$$

1. Preprocessing
2. Postprocessing

## روش چهارم

در این روش، از یک ماشین پیش‌بینی (یا شبکه عصبی) با  $\hat{a}$  ورودی و  $0$  خروجی برای پیش‌بینی «رشد مقادیر سری زمانی» استفاده شده است. در واقع، همانند روش سوم قبل از ماشین پیش‌بینی، یک مرحله پیش‌پردازش و پس از آن، یک مرحله پس‌پردازش وجود دارد. در این روش، رشد مقادیر سری زمانی در هر مرحله نسبت به مرحله قبل مطابق با فرمول زیر پیش‌بینی می‌شود.

$$\begin{aligned}y_1 &= x_1 \\y_k &= x_k - x_{k-1} \\ \underline{y}_1 &= \underline{y}_1 \\ \underline{x}_k &= \underline{y}_k - \underline{y}_{k-1}\end{aligned}$$

## ۳. ارزیابی عملکرد ماشین پیش‌بینی

## ۳-۱. معیار ارزیابی

از آنجا که هدف اصلی این مقاله، بررسی عملکرد شبکه‌های عصبی با الگوریتم‌های مختلف در پیش‌بینی یک سری زمانی با مدل غیر خطی است، بسیار ضروری است تا معیار محکم و مطمئنی برای ارزیابی انتخاب شود. دو معیار که در این نوشتار مورد استفاده قرار گرفته عبارتند از «نسبت پراکندگی نمونه‌های پیش‌بینی شده به پراکندگی نمونه‌های سری زمانی» و «متوسط قدر مطلق خطای پیش‌بینی نمونه‌های تست» که در مراجع مختلف علمی قبلاً مورد استفاده قرار گرفته است که ذیلاً به معرفی آنها می‌پردازیم:

$R$ ، نسبت پراکندگی نمونه‌های پیش‌بینی شده به پراکندگی نمونه‌های سری زمانی

$$\begin{aligned}e(x) &= \sum x_k / n \\v(x) &= \sum (x_k - e(x))^2 / n \\v(\underline{x}) &= \sum (x_k - \underline{x}_k)^2 / n \\R &= v(\underline{x}) / v(x)\end{aligned}$$

$P$ ، متوسط قدر مطلق خطای پیش‌بینی نمونه‌های آزمون

$$p_k = |(x_k - \underline{x}_k) / x_k| * 100$$

$$P = e(p) = \sum p_k / n$$

$$v(p) = \sum (p_k - e(p))^2 / n$$

در این نوشتار، معیار اصلی ارزیابی، مقدار نسبت پراکندگی نمونه‌های پیش‌بینی شده به پراکندگی نمونه‌های سری زمانی (R) است و مقدار متوسط قدر مطلق خطای پیش‌بینی نمونه‌های آزمون (P) نیز در کنار آن مورد توجه قرار گرفته است.

### ۲-۳. متدولوژی ارزیابی عملکرد ماشین پیش‌بینی کننده

برای ارزیابی ماشین پیش‌بینی کننده از چندین مجموعه سری زمانی استفاده شده است. اکثر این نمونه‌ها مربوط به شاخص‌های اقتصادی است که از نرم افزارهای کاربردی و یا از طریق اینترنت به دست آمده‌اند. ضمناً، در مواردی از شاخص‌های اقتصادی کشور نیز استفاده شده است. لکن، چنانچه تأکید شد، هدف اصلی، ارزیابی روش‌های پیش‌بینی است، لذا، صرفاً از این مجموعه‌های سری زمانی به عنوان ابزار ارزیابی استفاده شده و توجهی به خود شاخص و مشخصه‌های درونی آن نشده است. مسلماً، در صورت پرداختن به مشخصه‌های درونی آن سری زمانی شاید نتایج بسیار بهتری در پیش‌بینی حاصل شود.

در ارزیابی، یک سری زمانی به دو بخش تقسیم می‌شود. بخش اول (که عموماً بیش از ۶۰ درصد سری زمانی است)، مجموعه آموزشی ماشین پیش‌بینی کننده را تشکیل می‌دهد. از نمونه‌های بخش اول، به صورت ترکیب‌های  $i$ -تایی تعداد  $p$  عنصر آموزش شبکه عصبی ساخته می‌شود. بخش دوم (که عموماً کمتر از ۴۰ درصد سری زمانی است)، برای آزمون و ارزیابی ماشین پیش‌بینی کننده به کار می‌رود. از نمونه‌های بخش دوم، به صورت ترکیب‌های  $i$ -تایی تعداد  $q$  عنصر آزمون شبکه عصبی ساخته می‌شود. البته، معیار نهایی ارزیابی عملکرد شبکه عصبی، برای محاسبه نسبت پراکندگی نمونه‌های پیش‌بینی شده به پراکندگی نمونه‌های سری زمانی، شامل تمام نتیجه پیش‌بینی شبکه اعم از مجموعه آزمون و آموزش است. اما، در محاسبه متوسط قدر مطلق خطای پیش‌بینی نمونه‌های آزمون، چنانچه از عنوان آن نیز مشخص است صرفاً نمونه‌های مجموعه آزمون مد نظر است.

از آنجا که عملکرد شبکه عصبی تا حدودی وابسته به انتخاب مقادیر اولیه پارامترهای وزنی شبکه است، سعی شده هر آزمون چند بار انجام گیرد تا نتایج به اندازه کافی قابل اطمینان باشد. عوامل زیر در ارزیابی عملکرد پیش‌بینی تاثیر قابل توجهی دارد که در این مقاله، به تناسب اهمیت هر یک به آنها پرداخته شده است.

- روش پیش‌بینی
- الگوریتم آموزش شبکه عصبی
- ساختار شبکه عصبی (تعداد گره‌های لایه‌ها)
- پنجره نمونه‌های ورودی (تعداد ورودی شبکه عصبی)
- پنجره نمونه‌های خروجی (تعداد خروجی شبکه عصبی)
- ضریب یادگیری الگوریتم آموزش شبکه
- تعداد عناصر مجموعه آموزش نسبت به مجموعه ترمون
- نوع سری زمانی

در این راستا ابتدا، الگوریتم‌های مختلف به ازای ساختارهای مختلف شبکه عصبی مورد بررسی و ارزیابی قرار گرفته و سه الگوریتم موفق تر انتخاب می‌شود. سپس، به ازای این سه الگوریتم، ساختارهای متنوع تری از شبکه با پنجره نمونه‌های متنوع و نیز ضرایب یادگیری مختلف مورد بررسی و ارزیابی قرار می‌گیرد. در این مقطع، الگوریتم‌های موفق همراه با ساختار مناسب به صورت آماری شناسایی و ماشین‌های پیش‌بینی کننده با عملکرد مطلوب انتخاب می‌شوند. در مرحله بعد، رفتار ماشین پیش‌بینی کننده به ازای تعداد عناصر مختلف آموزشی و تستی و نیز به ازای سری‌های متنوع مورد بررسی و ارزیابی قرار می‌گیرد. عوامل زیر در عملکرد ماشین پیش‌بینی کننده مورد ارزیابی قرار گرفته است.

## فهرست مؤلفه‌های تأثیرگذار در ارزیابی عملکرد پیش‌بینی سری‌های زمانی

الگوریتم‌های آموزش شبکه	ساختار شبکه و پنجره‌های نمونه‌ها	ضریب یادگیری	تعداد عناصر آموزشی و تستی	روش‌های پیش‌بینی
<b>Incremental Training</b> Without Momentum	2-4-1	0.1	300 Train	روش اول
	4-4-1	0.2	200 Test	
With Momentum	6-4-1	0.6	350 Train	روش دوم
	6-6-1	0.8		
<b>Batch Training</b> Without Momentum	8-8-1	10-10-1	400 Train	روش سوم
	10-15-1			
With Momentum	15-15-1	15-20-1	100 Test	روش چهارم
	20-30-1			
<b>Faster Training</b> <u>Heuristic Techniques</u> Variable Learning Rate Resilient Backpropagation (Rprop)	4-4-4	8-10-4		
	4-8-4			
	10-15-10			
	20-30-10			
<u>Numerical Optimization Techniques</u> Conjugate Gradient Quasi-Newton Levenberg-Marquardt				
<b>Improving Generalization</b> Regularization Automated Regularization Early Stopping				

نکات:

- برخی از الگوریتم‌های آموزش فوق، خود شامل چند نوع مختلف هستند که برای اختصار در اینجا فقط نام الگوریتم اصلی آورده شده است. لکن، در ارزیابی‌ها هر یک از انواع الگوریتم‌ها جداگانه بررسی شده است.
- ساختارها، ضرایب و تعداد نمونه‌های پیشنهادی در جدول بالا، از نتایج ارزیابی‌های بسیار حجیم اولیه که به صورت ترکیبی انجام گرفته، به دست آمده است.



## ۴. نتایج ارزیابی

مجموعه ارزیابی‌ها شامل ۲۴۴۲ مورد ارزیابی است که در آنها تمامی موارد ارزیابی یادشده، در نظر گرفته شده است.

جدول ۱. بهترین نتایج به دست آمده از ۲۴۴۲ مورد ارزیابی - روش اول

ردیف	الگوریتم آموزش	تعداد عناصر آموزشی	ساختار شبکه	ضریب یادگیری	درصد خطای پیش‌بینی P (%)	در صد پراکندگی R (%)
1	Conjugate Gradient – 10	350	4-4-1	0.8	0.60	5.57
2	Conjugate Gradient – 10	350	5-8-1	0.1	0.60	5.75
3	Conjugate Gradient – 11	400	4-4-1	0.1	0.70	5.75
4	Conjugate Gradient – 11	350	4-4-1	0.1	0.60	5.75
5	Conjugate Gradient – 10	350	10-15-1	0.1	0.60	5.94
6	Conjugate Gradient – 10	350	10-15-1	0.1	0.60	5.94
7	Conjugate Gradient – 10	350	5-8-1	0.8	0.60	5.94
8	Conjugate Gradient – 11	350	4-4-1	0.1	0.70	5.94
9	Conjugate Gradient – 11	350	2-4-1	0.1	0.70	6.12
10	Conjugate Gradient – 11	350	2-4-1	0.1	0.70	6.12
11	Conjugate Gradient – 11	350	2-4-1	0.1	0.70	6.12
12	Conjugate Gradient – 11	350	2-4-1	0.1	0.70	6.12
13	Conjugate Gradient – 9	350	10-15-1	0.1	0.70	6.31
14	Conjugate Gradient – 9	350	10-15-1	0.1	0.70	6.31
15	Conjugate Gradient – 8	350	10-12-1	0.1	0.70	6.31
16	Conjugate Gradient – 8	350	10-15-1	0.1	0.70	6.31
17	Conjugate Gradient – 10	350	4-4-1	0.1	0.70	6.31
18	Conjugate Gradient – 9	350	4-4-1	0.1	0.70	6.31
19	Conjugate Gradient – 9	350	8-10-1	0.8	0.70	6.49
20	Conjugate Gradient – 9	350	4-4-1	0.1	0.70	6.49
21	Conjugate Gradient – 8	350	8-10-1	0.1	0.80	6.68
22	Conjugate Gradient – 11	350	5-7-1	0.1	0.70	6.68
23	Conjugate Gradient – 8	350	5-5-1	0.1	0.70	6.68
24	Conjugate Gradient – 11	350	5-7-1	0.1	0.70	6.68
25	Conjugate Gradient – 8	350	8-10-1	0.1	0.80	6.68
26	Conjugate Gradient – 8	350	5-8-1	0.1	0.80	6.68
27	Conjugate Gradient – 11	350	2-4-1	0.1	0.70	6.86
28	Quasi Newton – 13	350	2-4-1	0.1	0.70	6.86
29	Quasi Newton – 13	350	2-4-1	0.1	0.70	6.86
30	Conjugate Gradient – 8	350	8-10-1	0.1	0.80	6.86

از جدول بالا می‌توان موارد زیر را استنباط کرد

- الگوریتم Conjugate Gradient Back Propagation در بین انواع الگوریتم‌ها مناسب‌ترین است.
  - ساختار شبکه چندان تأثیر گذار نیست و صرفاً تعداد خروجی 1 اهمیت دارد (بدین معنا که در هر مرحله پیش‌بینی، صرفاً یک نمونه آتی پیش‌بینی می‌شود).
  - میانگین خطای پیش‌بینی کمتر از 1% است (شرایط خوب).
- در نمونه‌های پیش‌بینی شده پراکندگی حدود 6% نسبت به پراکندگی سری زمانی افزایش می‌یابد (شرایط خوب).

جدول ۲- بهترین نتایج به دست آمده از ۲۴۴۲ مورد ارزیابی - روش دوم

ردیف	الگوریتم آموزش	تعداد عناصر آموزشی	ساختار شبکه	ضریب یادگیری	درصد خطای پیش‌بینی (P %)	درصد پراکندگی (R %)
1	Conjugate Gradient - 8	350	10-12-1	0.1	0.7	22.62
2	Conjugate Gradient - 9	400	10-15-1	0.1	0.5	22.62
3	Quasi Newton - 13	400	10-15-1	0.1	0.5	22.62
4	Quasi Newton - 13	350	5-8-1	0.1	0.8	22.62
5	Quasi Newton - 13	350	5-8-1	0.1	0.8	22.62
6	Conjugate Gradient - 8	400	10-12-1	0.1	0.5	23.81
7	Conjugate Gradient - 8	400	10-12-1	0.1	0.5	23.81
8	Conjugate Gradient - 9	400	10-12-1	0.1	0.5	23.81
9	Conjugate Gradient - 9	400	10-12-1	0.1	0.5	23.81
10	Conjugate Gradient - 10	400	10-12-1	0.1	0.5	23.81
11	Conjugate Gradient - 11	400	10-12-1	0.1	0.5	23.81
12	Conjugate Gradient - 12	400	10-12-1	0.1	0.5	23.81
13	Quasi Newton - 13	400	10-12-1	0.1	0.5	23.81
14	Resilient - 7	400	10-12-1	0.1	0.5	23.81
15	Conjugate Gradient - 8	400	10-15-1	0.1	0.5	23.81
16	Conjugate Gradient - 10	400	10-15-1	0.1	0.5	23.81
17	Conjugate Gradient - 11	400	10-15-1	0.1	0.5	23.81
18	Quasi Newton - 12	400	10-15-1	0.1	0.5	23.81
19	Conjugate Gradient - 9	350	10-15-1	0.1	0.8	23.81
20	Resilient - 7	400	10-15-1	0.1	0.5	23.81
21	Conjugate Gradient - 8	400	2-4-1	0.1	0.6	23.81
22	Conjugate Gradient - 9	400	2-4-1	0.1	0.5	23.81
23	Conjugate Gradient - 10	400	2-4-1	0.1	0.6	23.81
24	Conjugate Gradient - 11	400	2-4-1	0.1	0.6	23.81
25	Quasi Newton - 12	400	2-4-1	0.1	0.6	23.81
26	Quasi Newton - 13	400	2-4-1	0.1	0.5	23.81
27	Levenberg Marquardt - 14	400	2-4-1	0.1	0.5	23.81
28	Levenberg Marquardt - 14	400	2-4-1	0.1	0.6	23.81
29	Conjugate Gradient - 10	350	2-4-1	0.1	0.8	23.81
30	Resilient - 7	400	2-4-1	0.1	0.6	23.81

از جدول (۲) موارد زیر استنباط می‌شود.

- الگوریتم Conjugate Gradient Back Propagation در بین انواع الگوریتم‌ها مناسب‌ترین است.
- پنجره نمونه‌های ورودی بیش از ۱۰ و پنجره نمونه‌های خروجی ۱ باشد (به عبارتی پس از مشاهده ۱۰ نمونه قبلی یک نمونه آتی پیش‌بینی می‌شود).
- میانگین خطای پیش‌بینی کمتر از یک درصد است (شرایط خوب).
- در نمونه‌های پیش‌بینی شده پراکندگی حدود ۲۲ درصد نسبت به پراکندگی سری زمانی افزایش می‌یابد (شرایط خوب).

جدول-۳. بهترین نتایج به دست آمده از ۳۵ مورد ارزیابی - روش سوم

ردیف	الگوریتم آموزش	تعداد عناصر آموزشی	ساختار شبکه	ضریب یادگیری	درصد خطای پیش‌بینی P (%)	درصد پراکندگی R (%)
1	Conjugate Gradient – 11	350	4-4-1	0.1	0.60	5.94
2	Conjugate Gradient – 10	350	4-4-1	0.1	0.80	7.42

از جدول (۳) می‌توان دریافت که:

- تمایز چندانی در کاربرد روش اول و سوم مشاهده نمی‌شود.
- الگوریتم Conjugate Gradient Back Propagation در بین انواع الگوریتم‌ها مناسب‌ترین است.
- میانگین خطای پیش‌بینی کمتر از یک درصد است (شرایط خوب).
- در نمونه‌های پیش‌بینی شده پراکندگی حدود ۷ درصد نسبت به پراکندگی سری زمانی افزایش می‌یابد (شرایط خوب).

جدول - ۴. فراوانی پیش‌بینی با نسبت پراکندگی کمتر از ۱۰٪

الگوریتم آموزش	ساختار شبکه	میانگین در صد پراکندگی R (%)	فراوانی
Incremental - 1			0
Incremental with Momentum-2			0
Batch - 3	2-4-1	9.6	2
Batch with Momentum - 4	2-4-1	9.6	2
Variable Learning Rate - 5			0
Variable Learning Rate - 6			0
Resilient - 7	5-8-1, 8-10-1	9	15
Conjugate Gradient - 8	10-12-1, 10-15-1	8	34
Conjugate Gradient - 9	10-15-1, 4-4-1	8	30
Conjugate Gradient - 10	4-4-1, 5-8-1	7	35
Conjugate Gradient - 11	4-4-1, 2-4-1	7	48
Quasi Newton - 12	5-8-1, 10-12-1	8	20
Quasi Newton - 13	2-4-1, 8-10-1	8	28
Levenberg Marquardt - 14	2-4-1	8	2
Regularization - 15			0
Auto Regularization - 16	2-4-1	8	44
Early Stopping - 17	8-10-1, 10-15-1	8	15

آنچه از جدول بالا می‌توان دریافت به قرار زیر است:

- الگوریتم **Conjugate Gradient Back Propagation** در بین انواع الگوریتم‌ها مناسب‌ترین است.
- پس از آن الگوریتم **Automated Regularization Back Propagation** الگوریتم مناسبی است.
- به طور کلی پنجره نمونه‌های ورودی کوچک مناسب است (به عبارتی با ۴ نمونه قبلی یک نمونه آتی را می‌توان پیش‌بینی کرد).
- ساختار (۲-۴-۱) و (۴-۴-۱) مناسب است.

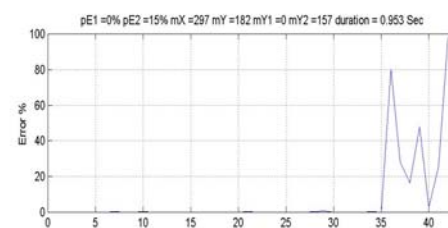
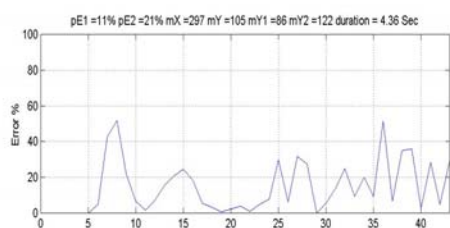
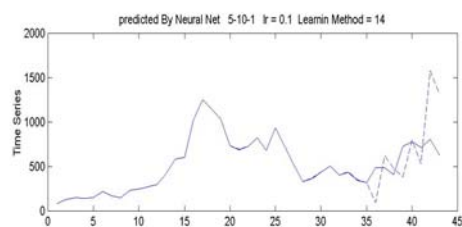
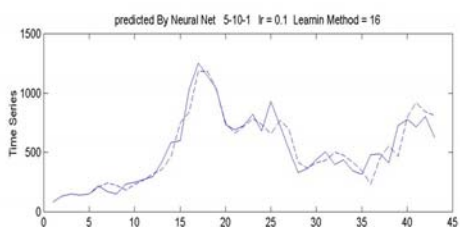
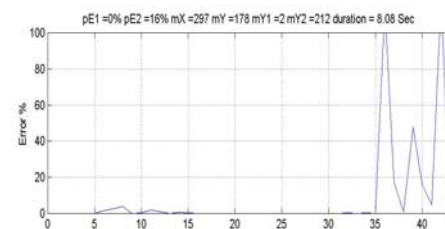
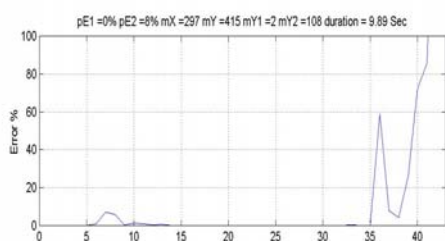
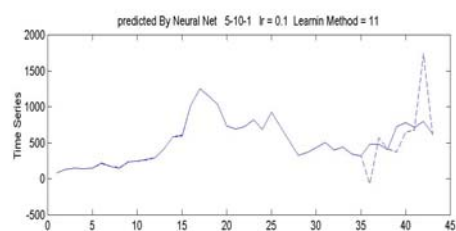
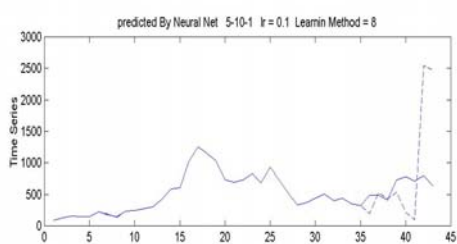
## ۵. چند نمونه پیش‌بینی

در آزمون‌های مختلف برای پیش‌بینی سری‌های زمانی از مجموع ۴۳ نمونه سری زمانی مربوط به مشخصه‌های اقتصادی سال‌های ۱۳۳۸ - ۱۳۸۰، سی نمونه اول سری زمانی به عنوان نمونه‌های مجموعه آموزشی و ۱۳ نمونه باقی مانده به عنوان نمونه‌های مجموعه آزمون مورد استفاده قرار گرفته است. شبکه عصبی به کار رفته یک شبکه MLP با ساختار ۵ ورودی، ۱۰ پرسپترون در لایه پنهان و یک خروجی و ضریب یادگیری ۰/۱ و حداکثر دفعات آموزش ۱۰۰۰ بار و گرادینت خطای هدف ۰/۰۰۰۰۱ با الگوریتم‌های آموزشی مختلف است. در مورد روش‌های آموزش شبکه عصبی انواع مختلف الگوریتم Conjugate Gradient (8-11) و الگوریتم Levenberg-Marquardt (14) و الگوریتم Automated Regularization (16) با هم مقایسه می‌شوند. در هر آزمون شماره کد روش آموزش به کار گرفته شده در قسمت بالای نمودار اول نشان داده شده است (رجوع شود به پیوست).

نتایج هر آزمون در دو نمودار نشان داده شده است. در نمودار اول، سری زمانی (خط ممتد) و سری زمانی پیش‌بینی شده (خط چین) نمایش داده شده است. در این نمودار، مقادیر به میلیارد ریال است. در نمودار دوم، خطای پیش‌بینی به درصد نمایش داده شده است. پارامترهای نمایش داده شده در هر نمودار به قرار زیر است:

pE2، متوسط خطای پیش‌بینی در مجموعه آزمون	pM1، متوسط خطای پیش‌بینی در مجموعه آموزشی
mY، پراکندگی سری زمانی پیش‌بینی شده به سری زمانی اولیه	mX، پراکندگی سری زمانی
mY2، پراکندگی سری زمانی پیش‌بینی شده به سری زمانی اولیه (محدود به مجموعه آزمون)	mY1، پراکندگی سری زمانی پیش‌بینی شده به سری زمانی اولیه (محدود به مجموعه آموزشی)
	Duration، مدت زمان آموزش شبکه عصبی به ثانیه

نمودار ۱- پیش‌بینی سری زمانی "تشکیل سرمایه ثابت ناخالص کل کشاورزی به قیمت ثابت ۱۳۶۹"

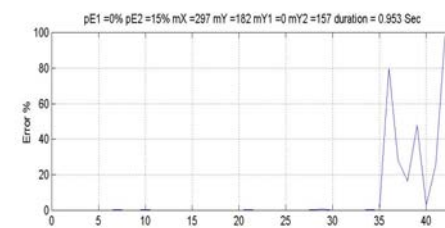
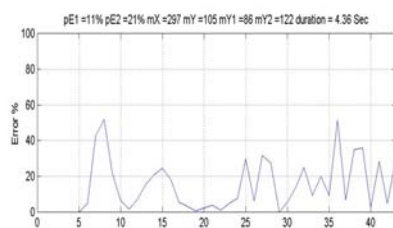
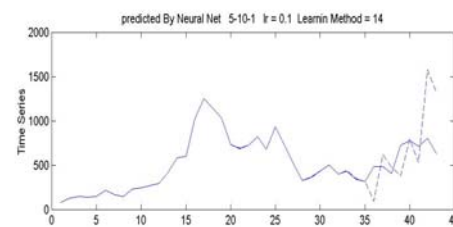
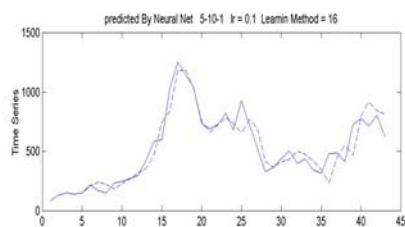
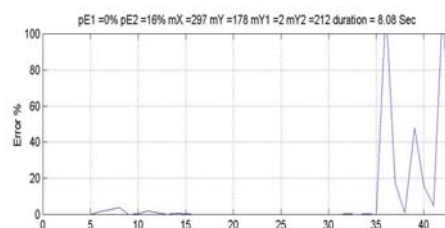
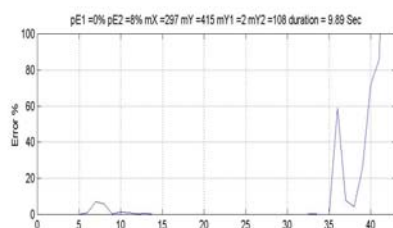
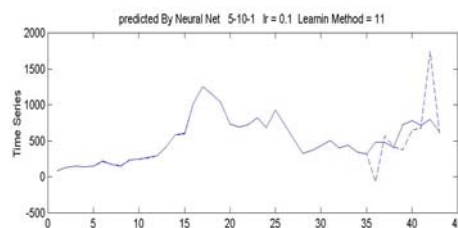
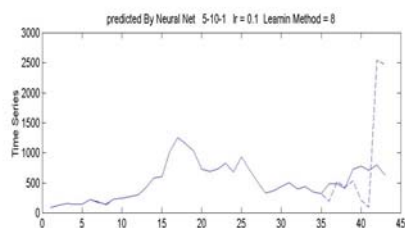


در یک نتیجه گیری مختصر از چهار آزمون پیش‌بینی، موارد زیر قابل ذکر است:

- الگوریتم **Conjugate Gradient** در آموزش نمونه‌های مجموعه آموزشی (۱ تا ۳۰) نتایج بسیار مطلوب دارد، اما، در تعمیم پذیری نمونه‌های مجموعه آزمون در پیش‌بینی ۵ نمونه اول مجموعه آزمون (۳۱ تا ۳۵) یعنی ۴۰٪ اولیه نتایج مطلوب و در باقی نمونه‌ها (۳۶ تا ۴۳) نتایج نامطلوب دارد.
- الگوریتم **Levenberg-Marquardt** نتایج بسیار مشابه با الگوریتم **Conjugate Gradient** در زمان بسیار کوتاه و حجم محاسبات کم دارد.
- گرچه از لحاظ میزان آموزش داده‌های مجموعه آموزشی در الگوریتم **Automated Regularization** نتایج چندانی مطلوبی حاصل نمی‌شود، اما نتایج تعمیم پذیری بسیار خوبی در مورد نمونه‌های مجموعه آزمون ارائه می‌کند.

Archive of Science

نمودار ۲- پیش‌بینی سری زمانی "تشکیل سرمایه ثابت ناخالص کل ساختمان به قیمت ثابت ۱۳۶۹"



در این سری زمانی نیز نتایج مشابه آزمون‌های سری زمانی قبل مشاهده می‌شود.



### ۶. نتیجه گیری

انتخاب یک شبکه عصبی با الگوریتم آموزشی و ساختار مناسب می تواند ابزار بسیار توانمندی برای پیش‌بینی یک سری زمانی فراهم کند. نتیجه پیش‌بینی می تواند تا آن اندازه قوی باشد که کل سری زمانی را قدم به قدم پی‌گیری کند. در این رهگذر، استفاده از الگوریتم آموزشی **Conjugate Gradient Back Propagation** و **Levenberg-Marquardt** و نیز الگوریتم آموزشی **Automated Regularization Back Propagation** با ساختار شبکه عصبی **Multi Layer Perceptron** با یک خروجی و چهار ورودی نتایج قابل اطمینانی را فراهم می‌کند.

Archive of SID

## منابع

۱. اصغری اسکوئی، محمدرضا. (۱۳۷۸). شبکه‌های عصبی و کاربرد آنها. گروه آمار، ریاضی و کامپیوتر
۲. اصغری اسکوئی، محمدرضا. (۱۳۷۴). شناخت الگوی سیگنال EMG چند کاناله توسط شبکه‌های عصبی در دست سایبرنتیک، هفتمین کنفرانس بین‌المللی مهندسی پزشکی، دانشگاه صنعتی شریف
3. Demuth, Howard & Beale, Mark. (1998). *Neural Network Toolbox For Use with MATLAB*. MathWork Inc.
4. Essenreiter, Robert. (1996). *Time Series Prediction with Neural Nets*. Wed May 29 20:43:00 CES.
5. Hush, Don R. & Home, Bill G. (1993). *Progress in Supervised Neural Networks*, IEEE Signal Processing Mag, Jan.
6. Moshiri, Saeed, Normane E. Cameron & David Scuse. (1999). *Static, Dynamic, and Hybrid Neural Networks in Forecasting Inflation*, Computational Economics.
7. Zhang, Xiru & Thearling, Kurt. (1994). *Non-Linear Time-Series Prediction by Systematic Data Exploration on a Massively Parallel Computer*. Santa Fe Institute Technical Report 94-07-045

Archive.org

**پیوست**

در این پیوست فهرست کامل روش‌های مختلف آموزش شبکه‌های عصبی، به همراه تقسیم بندی آنها و برخی مشخصه‌های اصلی هر یک جهت کاربرد آنها در نرم افزار MATLAB آمده است.

**Neural Nets Training Methods**

Categories:

➤ **Incremental Training**

1. Learngd (passes , lr)
2. Learnadm (passes , lr , mc)

➤ **Batch Training**

3. Traingd (epochs, show, goal, time, min\_grad, max\_fail, lr)
4. Trainadm (epochs, show, goal, time, min\_grad, max\_fail, lr, mc, max\_perf\_inc)

➤ **Faster Training**

○ Heuristic Techniques

▪ Variable Learning Rate

5. Traingda (epochs, show, goal, time, min\_grad, max\_fail, lr, max\_perf\_inc, lr\_dec, lr\_inc)
6. Traingdx (epochs, show, goal, time, min\_grad, max\_fail, lr, max\_perf\_inc, lr\_dec, lr\_inc, mc)

▪ Resilient Backpropagation (Rprop)

7. Trainrp (epochs, show, goal, time, min\_grad, max\_fail, delt\_inc, delt\_dec, delta0, deltamax)

- Numerical Optimization Techniques

- Conjugate Gradient

- 8. Traincgf (epochs, show, goal, time, min\_grad, max\_fail, srchFcn, scal\_tol, alpha, beta, delta, gama, low\_im, up\_lim, maxstep, minstep, bmax)

- 9. Traincgp (epochs, show, goal, time, min\_grad, max\_fail, srchFcn, scal\_tol, alpha, beta, delta, gama, low\_im, up\_lim, maxstep, minstep, bmax)

- 10. Traincgb (epochs, show, goal, time, min\_grad, max\_fail, srchFcn, scal\_tol, alpha, beta, delta, gama, low\_im, up\_lim, maxstep, minstep, bmax)

- 11. Trainscg (epochs, show, goal, time, min\_grad, max\_fail, sigma, lambda)

- Quasi-Newton

- 12. Trainbfg (epochs, show, goal, time, min\_grad,

max\_fail, srchFcn,  
scal\_tol, alpha,  
beta, delta, gama,  
low\_im, up\_lim,  
maxstep, minstep,  
bmax) – srchbac

13. Trainoss (epochs,  
show, goal, time,  
min\_grad,  
max\_fail, srchFcn,  
scal\_tol, alpha,  
beta, delta, gama,  
low\_im, up\_lim,  
maxstep, minstep,  
bmax) – srchbac

- Levenberg-Marquardt

14. Trainlm (epochs,  
show, goal, time,  
min\_grad,  
max\_fail, mu,  
mu\_dec, mu\_inc,  
mu\_max,  
mem\_reduc)

➤ **Improving Generalization**

- Regularization

15. Trainbfg  
performFc  
n = 'msereg' ,  
ratio = 0.5

- Automated Regularization

16. Trainbr

- Early Stopping