

## پیش بینی بارش با استفاده از شبکه عصبی عمیق (خودرمزگذار پشته‌ای تنک با نویزدا مبتنی بر نرون سخت)

شراره ملبوسی<sup>۱\*</sup>، سید جواد سید مهدوی<sup>۲</sup>، مرتضی پاکدامن<sup>۳</sup>

۱- کارشناس ارشد پژوهشی، پژوهشکده اقلیم شناسی مشهد، ایران

۲- دانشگاه آزاد اسلامی واحد مشهد، گروه برق، مشهد، ایران

۳- استادیار، پژوهشکده اقلیم شناسی و تغییر اقلیم، مشهد، ایران

### چکیده

در جهان امروز، توسعه سریع و پایدار، هدف اصلی تمامی کشورها می‌باشد. اصلی‌ترین محدودیت پیش روی توسعه پایدار، محدودیت‌های اقلیمی، از جمله بارش ناکافی همراه با پراکندگی نامناسب مکانی- زمانی است و بیشترین همبستگی را با حوادث ناگوار طبیعی دارد. یکی از روشهای مورد استفاده برای پیش‌بینی در حوزه‌های مختلف شبکه‌های عصبی مصنوعی می‌باشد که این شبکه‌ها به خاطر استفاده از معماری سطحی و کم‌عمق با ویژگیهای دستکاری شده ممکن است نتوانند دقت لازم را ارائه دهند. شبکه عصبی عمیق مشکلاتی مثل بیش‌برازش را برطرف می‌کند و همچنین هرچقدر عمق شبکه‌ها بیشتر باشند سطوح انتزاع بیشتری را یاد می‌گیرند. هدف اصلی این تحقیق، بالا بردن دقت پیش‌بینی بارش ساعتی منطقه خراسان رضوی با استفاده از یکی از روش‌های شبکه عصبی عمیق است. در این تحقیق ما یک معماری شبکه عصبی عمیق با روش خودرمزگذار پشته‌ای نویزدا مبتنی بر نرون سخت بصورت تنک (RSDSAE) را برای پیش‌بینی بارش کوتاه مدت ارائه می‌دهیم. به منظور بهبود دقت، شبکه‌های عصبی سخت (RNNs) به عدم قطعیت بارش کمک می‌کنند و برای بهبود سرعت و صحت از ترکیب الگوریتم تنک (Spars) با مدل فوق مورد استفاده قرار گرفته است و همچنین از داده‌های بارش پیش‌بینی شده خروجی مدل WRF استفاده کردیم و آزمایش‌ها بر روی داده‌های باران و توسط دو معیار  $MAE$ ،  $RMSE$  و با پنج خودرمزگذار به ترتیب ۰/۷۶۶۲ و ۰/۷۹۱۲ محاسبه گردیده است که توانسته عملکرد بهتری نسبت به مدل (RSDAE) از خود نشان دهد.

**کلید واژه‌ها:** شبکه عصبی عمیق، پیش‌بینی بارش، خودرمزگذار تنک پشته‌ای نویزدا.

## مقدمه

پیش‌بینی بارش‌ها یک نگرانی جدی است که توجه دولت‌ها، صنایع، نهادهای مدیریت ریسک و همچنین جامعه علمی را به خود جلب کرده است. بارندگی یک عامل آب و هوایی است که بر بسیاری از فعالیت‌های انسانی مانند تولید کشاورزی، ساخت و ساز، تولید برق، جنگلداری و گردشگری، و غیره تاثیر می‌گذارد. به این ترتیب، پیش‌بینی بارش ضروری است. بنابراین، داشتن یک روش مناسب برای پیش‌بینی بارش، می‌تواند اقدامات پیشگیرانه و تضعیف برای پدیده‌های مخرب طبیعی را انجام دهد. برای دستیابی به این پیش‌بینی‌ها، روش‌ها و الگوریتم‌های متعددی ارائه شده است، در طی چند سال گذشته، شبکه‌های عصبی عمیق (DNN) به عنوان مکانیسم موفق برای حل مشکلات پیچیده در زمینه‌هایی نظیر بینایی ماشین، تشخیص تصویر، پردازش زبان طبیعی و بیوانفورماتیک مورد استفاده قرار گرفته است (Hernández, et al., 2016). شبکه عصبی عمیق یک اصطلاح عمومی است که برای اشاره به مجموعه‌ای از معماری‌های چند لایه استفاده می‌شود که با استفاده از الگوریتم‌های بدون نظارت آموزش دیده‌اند. یکی از چالش‌ها در آموزش شبکه‌های عصبی عمیق، کندی روند آموزش شبکه به دلیل تغییر توزیع ورودی‌های هر لایه در طول آموزش است. در پژوهش‌های پیشین روشی با عنوان نرمالیزاسیون دسته‌ای مبتنی بر نرمال‌کردن ورودی هر لایه شبکه عصبی به عنوان جزئی از مدل و در طول هر دسته داده‌های آموزش پیشنهاد شده است (Ioffe & Szegedy, 2015). این روش امکان استفاده از نرخ‌های یادگیری بالاتر را فراهم کرده و اهمیت مقداردهی اولیه پارامترها را کمتر می‌کند که بدین ترتیب به صورتی طبیعی، به تسریع فرایند یادگیری کمک میکند. یکی دیگر از چالش‌ها در آموزش شبکه‌های عصبی عمیق با تعداد وزنها و پارامترهای زیاد بیش برآزش است. در واقع در این وضعیت، مدل به جای یادگیری الگوهای موجود، داده‌ها را به حافظه می‌سپارد. روش‌های مختلفی، همچون اعتبارسنجی متقابل برای جلوگیری از بیش برآزش وجود دارد (Srivastava, et al., 2014). روشی با عنوان حذف تصادفی برای جلوگیری از این پدیده در شبکه‌های عصبی عمیق پیشنهاد شده است که در آن، طی

آموزش شبکه‌عصبی، خروجی برخی نرون‌ها به طور تصادفی از فرآیند آموزش حذف می‌شوند. این موضوع از وابستگی بیش از حد مدل به داده‌های آموزش جلوگیری می‌کند (Srivastava, et al., 2014). عملکرد این روش در بسیاری از کاربردها نظیر بینایی ماشین و پردازش اسناد مورد بررسی قرار گرفته و پیشرفت چشمگیر عملکرد مدل‌های پیشنهادی در این موارد نشان داده شده است. اکثر مدل‌های موجود شبکه عصبی دارای سه اشکال هستند: (Khodayar, et al., 2017)

- ۱- اکثراً معماری کم‌عمق و سطحی است.
- ۲- بعضی از روش‌ها نیاز به ویژگی‌های دست‌ساز مهندسی که خسته‌کننده است دارند.
- ۳- اکثر روشها، مستقیماً دانش راجع به عدم قطعیت بارش ندارند.

برای رفع مشکلات یک و دو، از روش پیشنهادی خودرمزگذار پشته‌ای تنک نويزدا (SDSAE) و برای رفع مشکل سه، روش پیشنهادی را با نرون سخت ترکیب می‌کنیم. اتوانکودرها برای یادگیری ویژگی‌های بدون نظارت از داده‌های بارش بدون برچسب استفاده می‌کنند و یک لایه رگرسیون نظارت شده در بالای اتوانکودرها برای پیش‌بینی بارندگی اعمال می‌کنیم. به منظور بهبود دقت، شبکه‌های عصبی سخت با مدل یادگیری عمیق پیشنهادی ترکیب شده‌اند تا به عدم قطعیت بارندگی کمک کنند. در سازمان هواشناسی کشور از مدل‌های دینامیکی مانند WRF و ... برای پیش‌بینی منطقه‌ای استفاده می‌شود ولی چون خطا دارند می‌توان با این روشهای آماری خطا را کم کرد.

به طور معمول در یادگیری با نظارت یک شبکه عصبی با لایه‌های بسیار عمیق با وزن‌های تصادفی از یک توزیع گاوسی آغاز و مقداردهی می‌شود. سرعت الگوریتم پس انتشار خطا در شبکه مقداردهی شده بسیار کند است و در بهینه محلی گیر می‌کند، تابع هدف هم بسیار آسیب‌پذیر است که توسط میلیون‌ها متغیر آزاد با وابستگی‌های پیچیده مقداردهی می‌شود و به شبکه‌های عمیق عصبی منجر می‌شود که مدت زیادی طول می‌کشد تا نتایج بدست آمده آموزش داده شود. هیلتون در سال ۲۰۰۶ نشان داد که اگر شما هر لایه شبکه را در روش بدون نظارت برای یادگیری نمایش تنک<sup>۱</sup>

<sup>1</sup>- Spars

ورودی‌های  $X$  خودرمزگذار آموزش داده می‌شود تا ورودی‌های  $X$  خود را بازسازی کند، بنابراین بردارهای خروجی همان ابعاد بردار ورودی را دارند. آموزش یک خودرمزگذار، پیش‌آزمون بدون نظارت است؛ زیرا در طی فرآیند هیچ اطلاعات هدف داری استخراج نمی‌شود. لازم به ذکر است که فرآیند آموزش، کاهش ابعاد داده‌های ورودی در شبکه است، بسیار شبیه تجزیه و تحلیل مولفه‌های اصلی (PCA) است، اما خیلی بهتر از PCA کار می‌کند (Hinton & Salakhutdinov, 2006). فرآیند کلی یک اتوکر خودکار در شکل ۱ نشان داده شده است. در طول فرآیند، خودرمزگذار با به حداقل رساندن خطای بازسازی، بهینه می‌شود.

#### الگوریتم خودرمزگذار نویزدا پشته‌ای

خودرمزگذار پشته‌ای برای مدل‌سازی انتزاع‌های سطح بالا و کاهش ابعاد داده‌ها با استفاده از چندین لایه پردازش می‌باشد (Lin, et al., 2017). خودرمزگذار عمیق، برای افزایش استحکام داده‌های نویزی توسط لیکان پیشنهاد شده است. مدل خودرمزگذار پشته‌ای نویزدا از چندین خودرمزگذار پشته‌ای نویزدا چند منظوره (Vincent, et al., 2008) ساخته شده است که در بالای یکدیگر قرار دارند و ورودی را از ورودی خراب بازسازی می‌کند (Sagha, et al., 2017). خودرمزگذار نویزدا پشته‌ای، زیرساخت را برای حل مسائل نظیر خطر افزایش بیش‌برازش و حساسیت زیاد به نویز را فراهم می‌کند (Sagha, et al., 2017). چارچوب خودرمزگذار پشته‌ای نویزدا در شکل ۱ نشان داده شده است.

قبل از طبقه‌بندی آموزش دهید، مشکل یادگیری را بسیار کاهش خواهید داد. در مقالات گذشته از روش‌های خودرمزگذار پشته‌ای یا خودرمزگذار پشته‌ای نویزدار و یا ترکیب آنها با نرون سخت در خصوص پیش‌بینی باد (Khodayar, 2017) و (Khodayar, et al., 2017) و همچنین از شبکه‌های عصبی سخت برای پیش‌بینی دما (Zhang, et al., 2017) و از روش خودرمزگذار پشته‌ای نویزدار تنک برای داده‌های با ابعاد بالا هم استفاده کرده‌اند (Meng L., 2016) که با شبکه دو لایه و سه لایه و سطوح خراب شده طراحی کردند که طبق آزمایشاتی که انجام دادند دقت این مدل یا روش از روش‌های پشته‌ای دیگر بالاتر بوده است.

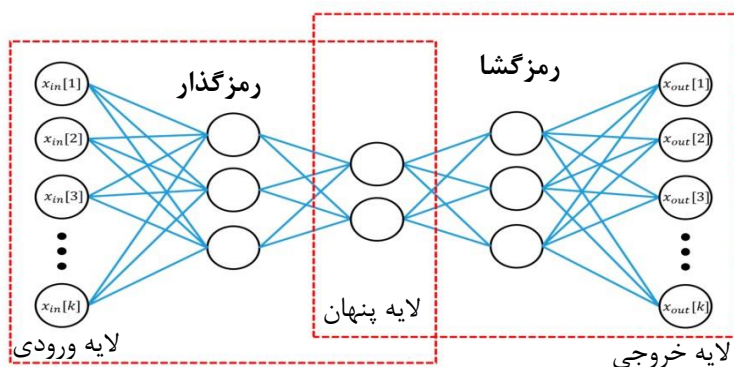
ما در این تحقیق برای پیش‌بینی بارندگی از روش خودرمزگذار پشته‌ای نویزدار با نرون سخت برای مقابله با عدم قطعیت و بالا بردن دقت پیش‌بینی و ترکیب آن با الگوریتم تنک استفاده نموده‌ایم. خودرمزگذار تنک یک شبکه تک لایه را هدایت می‌کند تا یک سری کد داده‌ها را یاد می‌گیرد و خطای بازسازی را به حداقل می‌رساند و تعدادی از کدهای مورد نیاز برای بازسازی را محدود می‌کند.

#### روش پیشنهادی

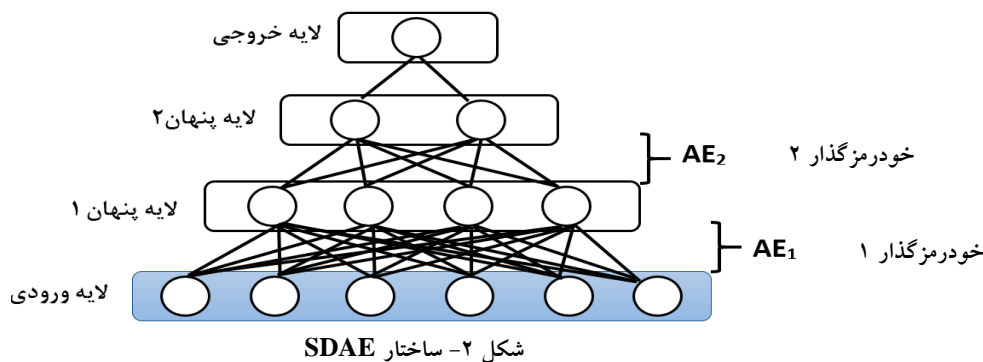
معرفی الگوریتم‌های روش پیشنهادی

الگوریتم خودرمزگذار

خودرمزگذار عمیق توسط هیتون و همکارانش در سال ۱۹۹۸ ارائه شده است. در مقالات اخیر (Zhang, et al., 2014)، (Jiang, et al., 2013)، (Zhou, et al., 2014) و (Guo, et al., 2016) به طور گسترده مورد مطالعه قرار گرفته شده است. در خودرمزگذارها، مقدار هدف  $Y$  با



شکل ۱- ساختار خودرمزگذار



بدست می‌آید و ترم جریمه استفاده شده بر اساس اختلاف

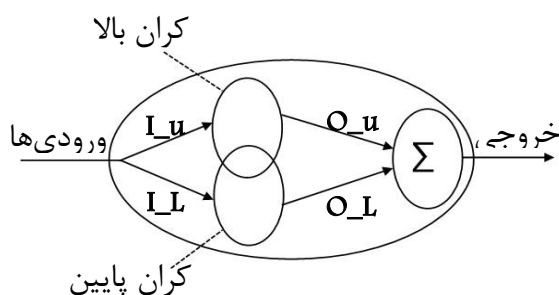
:KL

$$KL(r||\bar{r}_j) = \sum_{j=1}^s (r \log \frac{r}{\bar{r}_j} + (1-r) \log \frac{1-r}{1-\bar{r}_j}) \quad ۱$$

در این ترم جریمه اگر  $\bar{r}_j = r$  باشد  $KL(r||\bar{r}_j) = 0$  برقرار است در غیر اینصورت به صورت یکنواخت افزایش می‌یابد و همچنین  $\bar{r}_j$  از  $r$  واگرا می‌شود. (Kumar, et al., 2014).

### نرون سخت

نظریه مجموعه سخت برای حل ابهام و عدم اطمینان در داده‌ها بکار می‌رود (Vidhya & Geetha, 2017). نرون سخت مطابق شکل ۳ در ابتدا توسط لینگراس پیشنهاد شد. مهدی خدایار و همکارانش در سال ۲۰۱۷ معماری عصبی عمیق سخت برای پیش‌بینی کوتاه مدت سرعت باد و به منظور رسیدگی به عدم اطمینان در داده‌های سرعت باد، از یک نرون سخت استفاده کرده است (Khodayar, 2017).



شکل ۳- ساختار نرون سخت

(۱) دریافت داده‌های پیش‌بینی خروجی WRF، بارش باران و داده‌های دیگری که با داده‌های بارش همبستگی دارند و نرمال‌سازی آنها.

### خودرمزگذار تنک

خود رمز گذار تنک ابتدا در سال ۱۹۹۷ ارائه شد (OLSHAUSEN & FIELD, 1997). خودرمزگذار تنک افزونگی را کاهش می‌دهد (Ibrahim, et al., 2018) و دارای مزایای بالقوه بالایی هستند (Guo, et al., 2016):

(۱) استفاده از بازنمایش‌های با ابعاد بزرگ  
(۲) بازنمایش‌های تنک به ما یک تفسیر ساده از داده‌های ورودی پیچیده را با اصطلاحات یک تعداد از "بخش‌ها" ارائه می‌دهند.

چشم انداز بیولوژیکی بازنمایش‌های تنک در ابتدا از نواحی بصری استفاده می‌کند.

برای پیاده سازی این محدودیت، یک پارامتر اضافی به نام پارامتر اسپارسیته که  $r$  است در واحد پنهان اضافه می‌شود بعد از هر دوره،  $\bar{r}$  را محاسبه می‌کنیم که متوسط فعال‌سازی نوروهای پنهان  $z^h$  در کل مجموعه داده است. هنگامی که  $\bar{r}$  از  $r$  دور می‌شود، تابع هزینه را جریمه می‌کنیم. این کار با اضافه کردن یک ترم جریمه به تابع هزینه خودرمزگذار

مراحل انجام روش پیشنهادی

روش پیشنهادی دارای پنج مرحله اصلی می‌باشد که در ادامه بصورت خلاصه نام برده می‌شود:

مدل SAE مطابق شکل ۵ یک شبکه عصبی چند لایه عمیق با چندین AE است که دارای چندین لایه پنهان و یک لایه رگرسیون در بالا برای پیش‌بینی با نظارت می‌باشد (Y. Bengio, 2007). بصورت کلی هر AE که دارای K لایه پنهان است بصورت  $AE^K$  نمایش داده می‌شود. پارامترهای  $W_1^k$  و  $b_1^k$  وزن‌ها و بایاس‌های لایه رمزگذاری هستند در حالی که  $W_2^k$  و  $b_2^k$  پارامترهای مشابه برای لایه رمزگشایی هستند و همچنین فرض می‌کنیم  $W_1^k = (W_2^k)^T$  که "وزن‌های گره خورده" نامیده می‌شوند.

آموزش SAE دارای دو مرحله بزرگ و اساسی است: (۱) مرحله یادگیری ویژگی بدون نظارت که تنها با داده‌های ورودی بدون برچسب انجام می‌شود و (۲) مرحله تنظیم نهایی که بر روی مقادیر هدف بارش باران برای یادگیری با نظارت از پارامترهای شبکه عصبی اعمال می‌شود. در مرحله یادگیری ویژگی بدون نظارت ابتدا از لایه اول و با  $k=1$  شروع می‌کنیم، سپس هر خودرمزگذار  $AE^k$  بترتیب توسط الگوریتم پس‌انتشار خطا (BP) همراه با روش گرادیان نزولی تصادفی آموزش می‌بینند. بردار نمایش پنهان هر  $AE^k$  ورودی کدگذاری شده AE را به AE بعدی برای مثال  $AE^{k+1}$  منتقل می‌کند.

(۲) اعمال نویز به داده‌های ورودی.

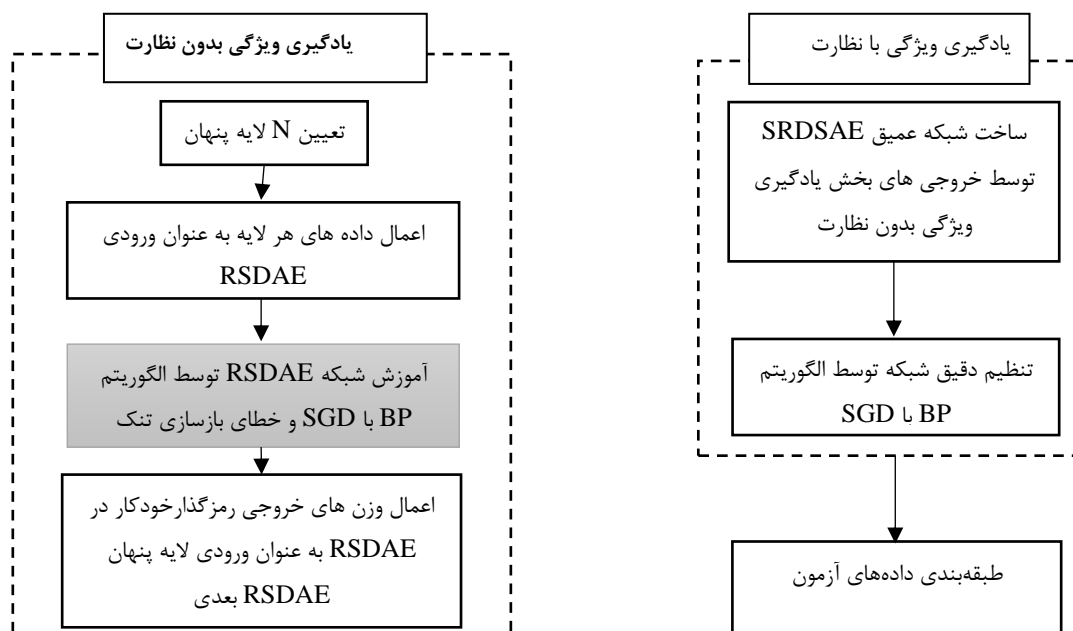
(۳) مقداردهی پارامترهای شبکه عمیق (تعداد لایه‌های پنهان، تعداد نرون‌های هر لایه).

(۴) یادگیری بدون نظارت با RSDAE برای ساخت شبکه عمیق چند لایه (مقداردهی پارامترهای شبکه RSDAE، محاسبه خروجی لایه پنهان توسط تابع فعال‌ساز سیگموئید، محاسبه وزن‌های خروجی و بروزرسانی وزن و بایاس توسط BP با روش SGD، استفاده از خروجی RSDAE فعلی به عنوان ورودی RSDAE بعدی، در نهایت مراحل فوق به تعداد لایه‌های پنهان تکرار می‌شود و خروجی آخرین لایه به عنوان ورودی بخش یادگیری با نظارت در ساختار شبکه عمیق در نظر گرفته می‌شود.

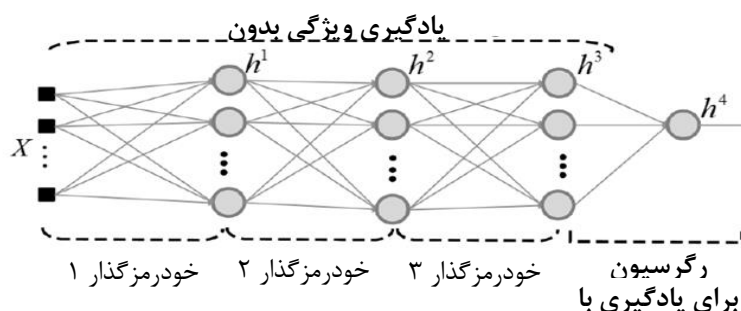
(۵) یادگیری با نظارت (تنظیم دقیق) توسط الگوریتم BP با روش SGD: اعمال داده‌های تست به RSDAE و محاسبه خطای پیش‌بینی.

#### فلوچارت روش پیشنهادی

فلوچارت روش پیشنهادی بصورت گرافیکی در شکل ۴ قابل مشاهده است.



شکل ۴- فلوچارت روش پیشنهادی



شکل ۵- ساختار SAE با سه AE

خروجی‌های پیش‌بینی مورد نظر به عنوان خروجی‌های مطلوب شبکه عصبی بیشتر آموزش داده می‌شود تا پارامترهای بهینه شبکه بدست بیاید. برای این کار ابتدا یک لایه رگرسیون خطی به آخرین لایه ANN اضافه می‌شود. پارامترهای این لایه بصورت تصادفی از  $[-1,1]$  مقداردهی اولیه می‌شود بصورتی که هیچ دانش قبلی درباره این مقادیر وجود ندارد. سپس، تمام مدل توسط داده‌های بارندگی به عنوان خروجی هدف تنظیم نهایی می‌شود. الگوریتم BP همراه با روش SGD برای تنظیم وزن‌ها و بایاس‌های تمام لایه‌ها در شبکه عصبی چندلایه اعمال می‌شود. از آنجا که پارامترهای لایه‌های اولیه به آرامی بروزرسانی می‌شود یک عبارت مومنتوم برای قوانین BP اعمال می‌شود. بنابراین برای پارامتر  $\theta$  قوانین بروزرسانی در دوره  $t$  بصورت معادله (۲-۳) می‌باشد (Khodayar, et al., 2017):

$$\mathcal{L}(h^{k-1}, r^k) = \frac{1}{M} \sum_{i=1}^M \left( \frac{1}{2} \|h^{k-1} - r^k\| \right)^2 + \frac{\lambda}{2} \sum_{i=1}^{n_k} \sum_{j=1}^{n_{k-1}} (w_{1,i,j}^k)^2 \quad ۲$$

$$\theta(t) = \theta(t-1) - \eta \frac{\partial E(t-1)}{\partial \theta(t-1)} + \gamma \Delta \theta(t-1) \quad ۳$$

در یادگیری توابع هدف پیچیده و توانایی شبکه عصبی سخت برای کنترل و مدیریت عدم اطمینان در داده‌ها، ترکیب شبکه عصبی سخت با معماری عمیق را برای کاربردهای مختلف ترویج کرد که ما از این ترکیب برای پیش‌بینی بارندگی استفاده نموده‌ایم. در شکل ۶ نرون سخت

نتیجه پارامترهای  $w_1^k$  و  $b_1^k$  از هر  $AE^k$  برای مقداردهی اولیه پارامترهای لایه  $k$  از شبکه عصبی مصنوعی چند لایه استفاده می‌شود. با گرفتن یک مجموعه آموزش دارای  $M$  نمونه، معیار خطا برای هر  $AE^k$  با ورودی  $h^{k-1}$  نمایش پنهان  $h^k$  و بردار خروجی  $r^k$  بصورت معادله ۲ نمایش داده می‌شود (Khodayar, et al., 2017).

در اینجا، عبارت خطای تنظیم  $L_2$  ضریب  $\lambda \geq 0$  دارد و  $w_{1,i,j}^k$  یک عنصر (جزء) از ماتریس وزن برای لایه رمزگذاری  $K$  که وزن اتصال واحد  $i$  در لایه  $k$ ام به واحد  $j$  در لایه قبلی نمایش می‌دهد.

عبارت تنظیم، یک پناستی پیچیدگی برای خطای بازسازی اضافه می‌کند و از بیش‌برازش توسط کاهش بزرگنمایی وزن‌ها جلوگیری می‌کند. در مرحله تنظیم نهایی، شبکه عصبی چند لایه که توسط مرحله یادگیری ویژگی بدون نظارت مقداردهی اولیه شد، با استفاده از الگوریتم SGD

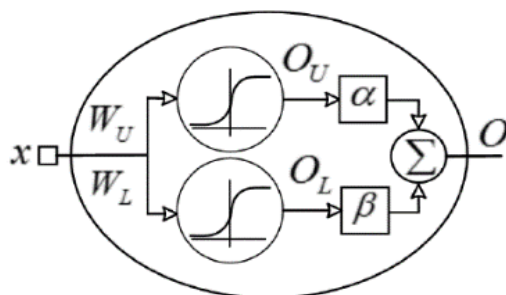
در اینجا  $0 \leq \gamma < 1$  که ضریب مومنتوم است. همچنین  $E$  خطای مربعات در هر نمونه است و  $\theta$  نمایش وزن‌ها و بایاس‌های شبکه عصبی چند لایه می‌باشد.

در سال ۱۹۹۶ لینگراس و همکارانش شبکه عصبی سخت (RNN) را با استفاده از مفهوم کران پایین و بالا معرفی کردند (P. Lingras, 1996). موفقیت شبکه عصبی عمیق

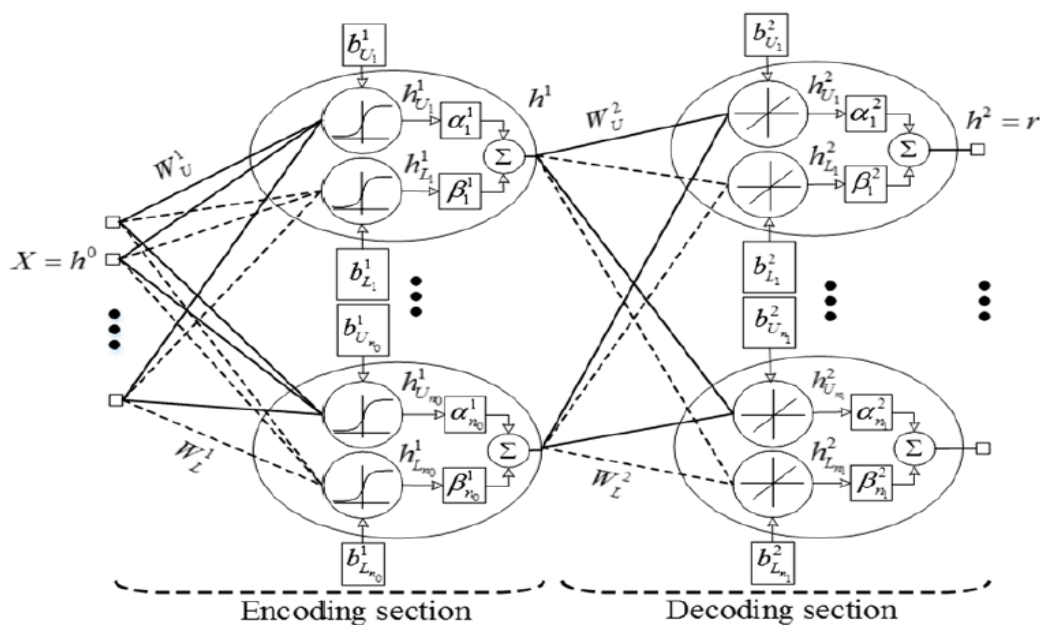
پایین  $O_L$  برای خروجی نرون  $O$  تعیین می‌شود. یک توسعه سخت از AE خودرمزگذار سخت (RAE) نامیده می‌شود که نرون‌های سخت در لایه پنهان و لایه خروجی اعمال می‌گردند. و  $W_U^k$  و  $b_U^k$  و  $W_L^k$  و  $b_L^k$  پارامترهای کران بالا و  $\alpha^k$  و  $\beta^k$  پارامترهای کران پایین از لایه  $k$  هستند. در شکل (۷-۲) ساختار RSDAE با برداری ورودی  $h^0 = X$  نمایش داده شده است (Khodayar, et al., 2017).

اعمال شده را برای تشخیص الگوی سخت در روش پیشنهادی را نمایش می‌دهد.

این نرون شامل یک نرون کران بالا با پارامترهای  $\theta_U = \{W_U, b_U, \alpha\}$  و یک نرون کران پایین با پارامترهای  $\theta_L = \{W_L, b_L, \alpha\}$  می‌باشد. در اینجا  $W_U$  و  $b_U$  وزن‌ها و بایاس‌های کران بالا و  $W_L$  و  $b_L$  پارامترهای مشابه برای نرون کران پایین هستند. ضرایب خروجی  $0 \leq \alpha$  و  $\beta \leq 1$  تعیین سهم از خروجی کران بالا  $O_U$  و خروجی کران



شکل ۶- نرون سخت با شش پارامتر قابل تنظیم



شکل ۷- ساختار RSDAE (Khodayar, et al., 2017)

خروجی کران بالا و پایین از اولین لایه پنهان بصورت معادله ۴ و معادله ۵ نمایش داده می‌شود (Khodayar, et al., 2017):

$$h_U^1 = \text{Max}(f^1(W_U^1 X + b_U^1), f^1(W_L^1 X + b_L^1)) \quad 4$$

$$h_L^1 = \text{Min}(f^1(W_U^1 X + b_U^1), f^1(W_L^1 X + b_L^1)) \quad 5$$

نمایش پنهان در لایه پنهان بصورت معادله ۶ محاسبه می‌شود (Khodayar, et al., 2017):

$$h^1 = \alpha^1 h_U^1 + \beta^1 h_L^1 \quad 6$$

برای فرآیند رمزگشایی سخت در لایه خروجی، خروجی‌های کران پایین و بالا بصورت معادله ۷ و ۸ محاسبه می‌شوند (Khodayar, et al., 2017):

$$h_U^2 = \text{Max}(f^2(W_U^2 h^1 + b_U^2), f^2(W_L^2 h^1 + b_L^2)) \quad 7$$

$$h_L^2 = \text{Min}(f^2(W_U^2 h^1 + b_U^2), f^2(W_L^2 h^1 + b_L^2)) \quad 8$$

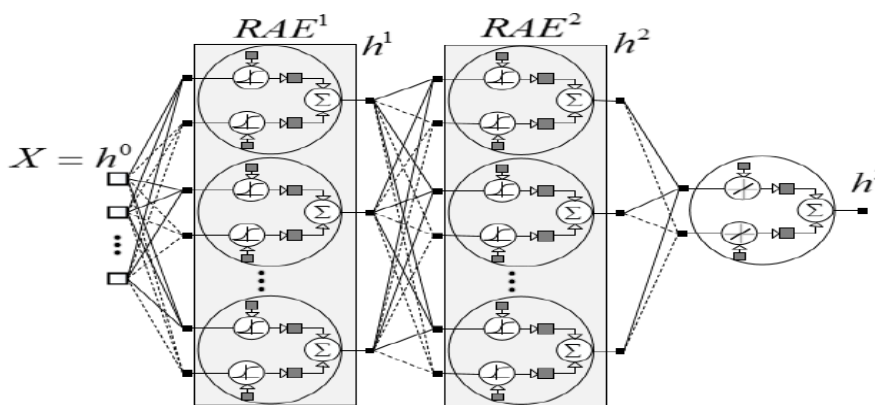
بنابراین، ورودی بازسازی شده بصورت ذیل محاسبه می‌شود (Khodayar, et al., 2017):

$$r = \alpha^2 h_U^2 + \beta^2 h_L^2 \quad 9$$

مربوط به لایه  $k$ ام با بردار ورودی  $h^{k-1}$  و بردار خروجی  $rk$  بصورت RSDAE $k$  مشخص می‌شود. ما از RSDAE $1$  شروع می‌کنیم، و بترتیب RSDAE های بعدی توسط الگوریتم BP همراه SGD به منظور یادگیری ویژگی‌های سخت از مجموعه داده بارندگی وارد می‌شوند.

### ساختار روش پیشنهادی (RSDSAE)

در این قسمت، ما یک شبکه عصبی عمیق DNN با خصوصیتی همچون مقاوم بودن و کاهش خطای پیش‌بینی توسط اسپارس کردن RSDAE طراحی کرده‌ایم و در شکل ۸ ساختار RSDSAE را نشان داده‌ایم که شامل دو RSDAE می‌باشد. بصورت کلی در این شکل RSDAE



شکل ۸- ساختار RSDSAE با دو RSDAE

برای آموزش هر RSDSAE از خطای بازسازی معادله ۱۰ استفاده می‌نماییم:

$$J_{RE} = \frac{1}{M} \sum_{i=1}^M \left( \frac{1}{2} \|h^k - r^k\|^2 \right) + \frac{\lambda}{2} \left( \sum_{i=1}^{n_k} \sum_{j=1}^{n_{k-1}} (W_{U,i,j}^k)^2 + \sum_{i=1}^{n_k} \sum_{j=1}^{n_{k-1}} (W_{L,i,j}^k)^2 + \beta \sum_{j=1}^s KL(r||\bar{r}) \right) \quad 10$$

که در اینجا معادله اسپارس به شرح زیر است (Kumar, et al., 2014):

$$KL(r||\bar{r}_j) = \sum_{j=1}^s \left( r \log \frac{r}{\bar{r}_j} + (1-r) \log \frac{1-r}{1-\bar{r}_j} \right) \quad 11$$

گام  $t+1$  توسط قانون زنجیره‌ای تعیین می‌شود که بصورت معادله ۱۲ محاسبه می‌شود (Khodayar, et al., 2017): شبیه به همین فرمول برای وزن کران پایین  $W_L^k$  با جایگزین کردن کران بالا با پارامترهای کران پایین اعمال می‌شود. بر این اساس، فرمول مشابه برای پارامترهای بایاس با جایگزین

یک نرون سخت با تابع فعال‌سازی خطی در لایه رگرسیون اعمال می‌شود. بعد از مقداردهی اولیه تصادفی از لایه رگرسیون سخت، RSDSAE با استفاده از BP به همراه گرادیان نزولی و مونتوم برای تنظیم نهایی استفاده می‌شود. قانون بروزرسانی برای وزن کران بالا  $W_U^k$  از SRAEK در



با اسپارس کردن RSDAE باعث کاهش خطای پیش‌بینی می‌شویم که در فصل چهارم نتایج بدست آمده را مورد تجزیه و تحلیل قرار داده‌ایم.

$$w_U^k(t+1) = W_U^k(t) - \eta \frac{\partial E}{\partial W_U^k}(t) + \gamma \Delta W_U^k(t) \quad ۱۲$$

$$\frac{\partial E}{\partial W_U^k} = \frac{\partial E}{\partial h^k} \left( \frac{\partial h^k}{\partial h_U^k} \frac{\partial h_U^k}{\partial W_U^k} + \frac{\partial h^k}{\partial h_L^k} \frac{\partial h_L^k}{\partial W_U^k} \right) = \frac{\partial E}{\partial h^k} \left( \alpha^k \frac{\partial h_U^k}{\partial W_U^k} + \beta^k \frac{\partial h_L^k}{\partial W_U^k} \right) \quad ۱۳$$

$$\alpha^k(t+1) = \alpha^k(t) - \eta \frac{\partial E}{\partial \alpha^k}(t) + \gamma \Delta \alpha^k(t) \quad ۱۴$$

$$\frac{\partial E}{\partial \alpha^k} = \frac{\partial E}{\partial h^k} \frac{\partial h^k}{\partial \alpha^k} = -\eta \frac{\partial E}{\partial h^k} h_U^k \quad ۱۵$$

$$\beta^k(t+1) = \beta^k(t) - \eta \frac{\partial E}{\partial \beta^k}(t) + \gamma \Delta \beta^k(t) \quad ۱۶$$

$$\frac{\partial E}{\partial \beta^k} = \frac{\partial E}{\partial h^k} \frac{\partial h^k}{\partial \beta^k} = -\eta \frac{\partial E}{\partial h^k} h_L^k \quad ۱۷$$

هواشناسی در محدوده‌ای به عرض جغرافیایی ۱۸ تا ۴۸ درجه و در طول جغرافیایی ۲۳ تا ۷۰ درجه محاسبه شده‌اند. در این بازه شبکه‌بندی، ۹۹ نقطه شبکه‌ای از غرب به شرق و ۸۷ نقطه شبکه‌ای از شمال به جنوب وجود دارد، حدوداً هر نقطه شبکه‌ای ۱۵\*۳۰ کیلومتر مربع را پوشش می‌دهد.

#### آموزش و ارزیابی الگوریتم

از خروجی مدل پیش‌بینی میان‌مقیاس، هشت روز از داده‌های ساعتی بارش منطقه‌ای را به عنوان داده‌های متغیر هدف برای آموزش و امتحان الگوریتم‌ها در نظر گرفتیم. داده‌های مورد استفاده را از خروجی دو اجرای مدل پیش‌بینی منطقه‌ای (مدل میان‌مقیاس پیش‌بینی جوی) به تاریخ‌های ۲۶ و ۲۸ اکتبر ۲۰۱۸ برای هر اجرا با ۴ روز پیش‌بینی تهیه نمودیم. خروجی‌های این مدل با پسوند nc می‌باشند که ویژگی‌های مورد استفاده این تحقیق توسط اسکریپت‌نویسی در نرم افزار NCL با درون‌یابی نقاط شبکه‌ای و تبدیل شبکه‌ها به نقاط ایستگاه‌های هواشناسی استان خراسان‌رضوی، داده‌های پیش‌بینی شده استخراج گردید که این ویژگی‌های استخراج شده به عنوان داده‌های ورودی الگوریتم در این تحقیق استفاده شد.

کردن وزن‌ها بایاس‌ها نوشته می‌شود. در گام  $t+1$ ،  $\alpha^k$  و  $\beta^k$  بصورت ذیل بروزرسانی می‌شوند (Khodayar, et al., 2017):

#### ارزیابی روش پیشنهادی

داده‌ها و محدوده مورد مطالعه

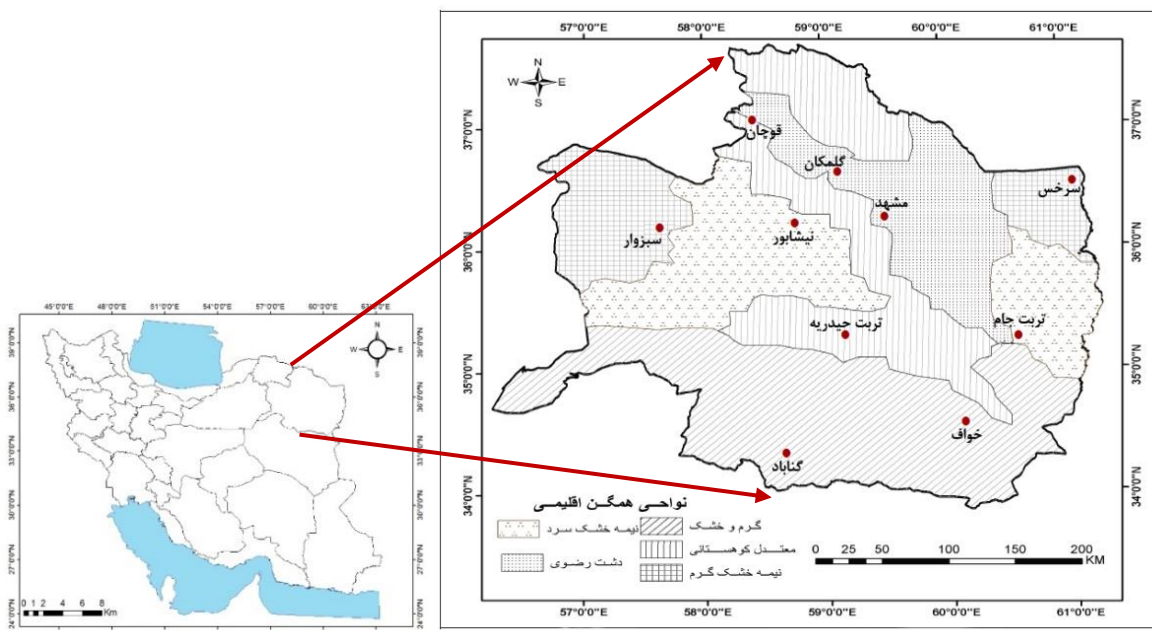
منطقه مورد مطالعه در این تحقیق، استان خراسان‌رضوی می‌باشد. بر اساس آمار دوره ۲۰۱۰-۱۹۸۱، دمای میانگین سالانه استان با تعداد ۱۰ ایستگاه‌های که شامل این دوره اقلیمی می‌باشند، به طور متوسط ۱۵.۵ درجه‌سانتیگراد و بارش میانگین سالانه ۲۳۳.۹ میلیمتر است. شکل ۳ منطقه مورد مطالعه و طبقه‌بندی اقلیمی استان را نشان می‌دهد. از داده‌های خروجی شبکه‌بندی شده WRF مطابق جدول ۳ که همان مدل پیش‌بینی میان‌مقیاس منطقه‌ای است به عنوان متغیر هدف برای آموزش و امتحان الگوریتم‌ها استفاده شده است.

نحوه شبیه‌سازی پارامترهای جوی با استفاده از مدل پیش‌بینی و تحقیق جوی (WRF)

مدل WRF یک مدل مشهور میان‌مقیاس منطقه‌ای برای پیش‌بینی وضعیت آب و هوایی است و برای طیف وسیعی از کاربری‌ها و برای مقیاس‌هایی از چند متر تا چند هزار کیلومتر مناسب است. جهت اجرای این مدل پیش‌بینی و پژوهش وضع هوا، شرایط اولیه جو و شرایط مرزی شامل دما و رطوبت خاک از داده‌های بازتحلیل مرکز ملی آمریکا NCEP با گام زمانی ۳ ساعته در دو تاریخ در سال ۲۰۱۸ استفاده گردید. با اجرای این مدل پیش‌بینی، داده‌های

جدول ۱- متغیرهای پیش‌بینی شده مورد استفاده در آزمایشها

ویژگیها	واحد	سطح فشار(هکتوپاسکال)
MONTH	-	
DAY	-	
HOUR	-	
LAT	Degree	
LON	Degree	
T	K	Ground, 925, 850, 700, 500, 300, 100
UM	%	100, 300, 500, 700, 850, 925
P	hpa	100, 300, 500, 700, 850, 925
Direct Speed	Degree	Ground, 925, 850, 700, 500, 300, 100
Wind Speed	m/s	Ground, 925, 850, 700, 500, 300, 100
U10	m/s	
V10	m/s	
TD	K	
TH2	K	
TSK	K	
SST	K	
TSLP	K	
TMN	K	
GRDFLX	W/m-2	
SWDOWN	W/m-2	
GLW	W/m-2	
OLR	W/m-2	
NOAHRES	W/m-2	
LH	W m-2	
HFX	W/m-2	
ACGRDFLX	J m-2	
ACHFX	J m-2	
ACLHF	J m-2	
ACSNOM	Kg/m-2	
SNOW	Kg/m-2	
CANWAT	Kg/m-2	
QFX	Kg/m-2 s-1	
SNOWH	m	
HGT	m	
PBLH	m	
RAINC	mm	
RAINNC	mm	
GRAUPELNC	mm	
COSZEN	-	
ALBEDO	-	
ALBCK	-	
EMISS	-	



شکل ۹- طبقه‌بندی اقلیمی استان خراسان رضوی بر اساس روش‌های آماری و موقعیت آن استان در ایران

جهت مقایسه دقیقتر استفاده کرده‌ایم. بدین منظور تعداد لایه‌های پنهان در شبکه عمیق از بین مقادیر {۱، ۲، ۳، ۴، ۵} انتخاب و تعداد نرون‌های موجود در هر لایه پنهان به صورت تصادفی از بین مجموعه زیر انتخاب می‌شود.

{۵، ۱۰، ۱۵، ۲۰، ۲۵، ۳۰، ۳۵، ۴۰، ۴۵، ۵۰، ۵۵، ۶۰، ۶۵، ۷۰، ۷۵}

همچنین در الگوریتم BP مقدار پارامتر تنظیم وزن  $\lambda$  در عبارت تنظیم برابر با ۰.۴ و مقدار نرخ یادگیری برابر با ۰.۵ در نظر گرفته می‌شود. در روش پیشنهادی RSDSAE برای محاسبه تنک در بخش خطای بازسازی نیاز به تعریف پارامتر  $p$  و  $\beta$  داریم که پارامتر  $p$  برابر با عدد ۰.۰۵ و پارامتر  $\beta$  برابر با مقدار ۰.۵ قرار می‌گیرد. هر دو روش پیشنهادی و روش پایه به تعداد ۱۰۰ بار تکرار شده‌اند.

### تجزیه و تحلیل نتایج

روش پیشنهادی RSDSAE و روش RSDAE جهت مقایسه بهتر در پیش‌بینی وضعیت بارش بر اساس دو پارامتر ارزیابی RMSE و MAE مورد مقایسه و تجزیه و تحلیل قرار گرفته‌اند. تمامی آزمایش‌ها در شرایط یکسان انجام شده است و نتایج حاصل در جدول ذیل از میانگین ۱۰ بار اجرای کامل الگوریتم‌ها بدست آمده است. نتایج حاصل از مقایسه دو روش پیشنهادی و پایه بر اساس معیار ارزیابی

جهت ارزیابی روش پیشنهادی SRDSAE و RSDAE از دو معیار میانگین مربعات خطا (RMSE) میانگین خطای مطلق استفاده می‌کنیم که در ذیل معادله‌های مربوط به این دو معیار نشان شده است:

$$RMSE = \sqrt{\frac{1}{M} \sum_{n=1}^M e(n)^2} \quad 18$$

$$MAE = \frac{1}{M} \sum_{n=1}^M |e(n)| \quad 19$$

در اینجا  $e(n) = t(n) - y(n)$ ،  $M$  تعداد نمونه‌ها در مجموعه تست است.  $t(n)$  و  $y(n)$  بترتیب خروجی بدست آمده و خروجی اصلی از مدل برای  $n$  نمونه است.

### روش انجام محاسبات

پیاده‌سازی روش پیشنهادی RSDSAE و روش RSDAE در محیط نرم افزار متلب نسخه ۲۰۱۷ و با مشخصات سیستمی با ۷ cpu هسته ای و ۸ گیگابایت و گرافیک ۱ گیگابایت انجام شده است. در الگوریتم روش پیشنهادی RSDSAE و روش پایه RSDAE برای مقادیر لایه‌های پنهان اعم از تعداد لایه‌های پنهان و همچنین تعداد نرون‌های موجود در هر لایه پنهان از یک استراتژی یکسان

اساس این دو نمودار به خوبی می‌بینیم که روش پیشنهادی RSDSAE توانسته است به دقت بالاتری در پیش‌بینی میزان بارش دست پیدا کند و مقادیر دو پارامتر ارزیابی RMSE و MAE را بهبود بخشد.

باران باعث بهبود زندگی عموم مردم و حتی برنامه‌ریزی صحیح حکومت‌ها برای استفاده صحیح از بارش می‌باشد. الگوریتم و روش‌های مختلفی در طی سال‌های اخیر برای مسائل مربوط به پیش‌بینی ارائه شده است. روش‌های یادگیری عمیق جزو روش‌های محبوب برای مسائل پیش‌بینی می‌باشد. ما در این تحقیق برای پیش‌بینی باران از روش خودرمزگذار پشته‌ای تنک نوپزدار و ترکیب این روش با نرون سخت و الگوریتم تنک برای بالا بردن دقت و صحت پیش‌بینی استفاده نموده‌ایم. ما در این تحقیق روش پیشنهادی را بر روی مجموعه داده‌های خروجی مدل پیش‌بینی میان مقیاس WRF در محدوده استان خراسان رضوی اعمال کردیم و در نهایت توانستیم با اجرای الگوریتم پیشنهادی، خطای پیش‌بینی را کاهش داده و صحت عملکرد پیش‌بینی را بهبود دهیم.

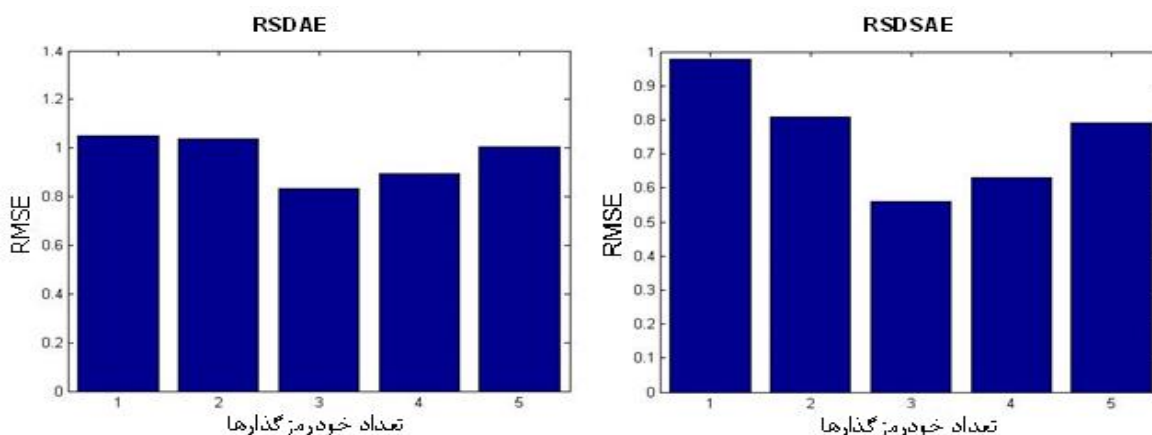
RMSE و MAE و با تعداد خودرمزگذارهای مختلف در جدول ۲ قرار داده شده است. همانطور که در نتایج دیده می‌شود روش پیشنهادی توانسته است با تنک کردن شبکه RSDAE منجر به کاهش خطای پیش‌بینی و در پی آن کاهش معیار RMSE و MAE شود و یا به عبارتی موجب بهبود پیش‌بینی ریزش باران شود.

در شکل‌های ۱۰ و ۱۱ مقادیر خروجی معیار ارزیابی RMSE و MAE برای تعداد رمزگذارهای خودکار مختلف بر روی نمودار نمایش داده شده است. همانطور که در نمودارها مشاهده می‌شود مقدار RMSE تا تعداد ۳ لایه پنهان کاهش یافته و پس از آن مقدار RMSE افزایش می‌یابد بر این اساس می‌توان نتیجه گرفت که بهترین تعداد برای لایه‌های پنهان در شبکه‌های عمیق RSDSAE و RSDAE برابر با سه لایه می‌باشد و در این تعداد لایه، به حداقل مقدار خطا و به دنبال آن به حداقل مقدار RMSE و MAE دست پیدا می‌کنیم و پیش‌بینی میزان بارش با دقت بالاتری انجام می‌شود.

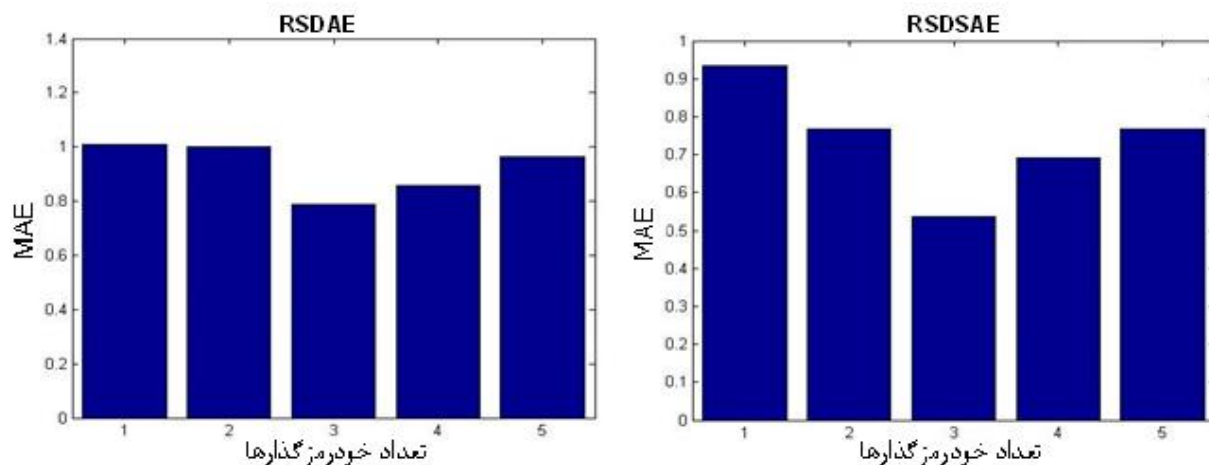
همچنین شکل ۱۲ برای مقایسه هر چه بهتر مقادیر پارامترهای ارزیابی RMSE و MAE در هر دو روش پایه و روش پیشنهادی در کنار یکدیگر طراحی شده‌اند. بر

جدول ۲- نتایج روش پیشنهادی RSDSAE و روش RSDAE با تعداد خودرمزگذارهای مختلف

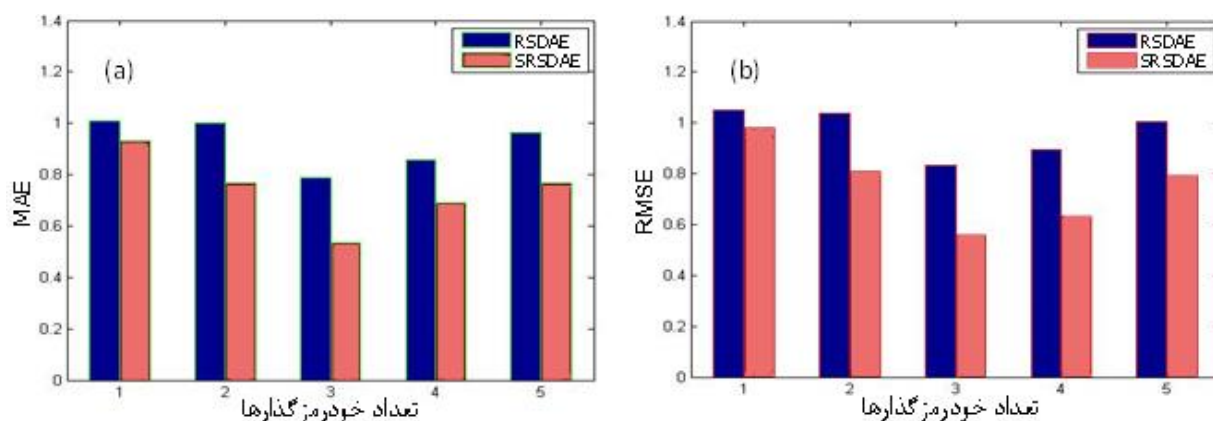
Number of AEs	RMSE		MAE	
	RSDAE	RSDSAE	RSDAE	RSDSAE
1	1.0510	<b>0.9793</b>	1.0104	<b>0.9333</b>
2	1.0383	<b>0.8088</b>	0.9992	<b>0.7677</b>
3	0.8317	<b>0.5607</b>	0.7873	<b>0.5356</b>
4	0.8962	<b>0.6316</b>	0.8564	<b>0.6916</b>



شکل ۱۰- نمودار مقایسه دو روش RSDAE و RSDSAE بر اساس معیار RMSE با تعداد خود رمزگذارهای مختلف



شکل ۱۱- نمودار مقایسه دو روش RSDAE و RSDSAE بر اساس معیار MAE



شکل ۱۲- نمودار مقایسه روش RSDAE و RSDSAE با استفاده از معیارهای (a) MAE و (b) RMSE

استان خراسان رضوی اعمال کردیم و در نهایت توانستیم با اجرای الگوریتم پیشنهادی، خطای پیش‌بینی را کاهش داده و صحت عملکرد پیش‌بینی را بهبود دهیم. با توجه به اهمیت پیش‌بینی بارش باران برای عموم مردم و حکومت‌ها، پیشنهاد ما برای کارهای آینده:

- ۱- استفاده از مدل اسپارس RBF مبتنی بر نرون سخت
- ۲- استفاده از مدل اسپارس CNN مبتنی بر نرون سخت
- ۳- یکی از پیشنهادها دیگر ما استفاده از مدل‌های پیش‌بینی هواشناسی دیگر به جای WRF از مدل‌های دیگر مانند GFS یا ECMWF و غیره و همچنین پیش‌بینی پارامترهای دیگر هواشناسی مثل دما برای خشکسالی و سرعت باد و غیره با روش‌های فوق می‌باشد.
- ۴- همچنین می‌توان از Ensemble برای ترکیب مدل‌های مختلف پیش‌بینی شبکه‌ای و اعمال این روش‌ها در این مدل‌ها استفاده کرد.

## نتیجه‌گیری

پیش‌بینی بارش باران یکی از چالش‌های مهم و اساسی برای محققین و پژوهشگران می‌باشد بصورتی که پیش‌بینی درست و صحیح بارش باران باعث بهبود زندگی عموم مردم و حتی برنامه‌ریزی صحیح حکومت‌ها برای استفاده صحیح از بارش می‌باشد. الگوریتم و روش‌های مختلفی در طی سالیان اخیر برای مسائل مربوط به پیش‌بینی ارائه شده است. روش‌های یادگیری عمیق جزو روش‌های محبوب برای مسائل پیش‌بینی می‌باشد. ما در این تحقیق برای پیش‌بینی بارش باران از روش خودرمزگذار پشته‌ای تنک نويزردار و ترکیب این روش با نرون سخت و الگوریتم تنک برای بالا بردن دقت و صحت پیش‌بینی استفاده نموده‌ایم. ما در این تحقیق روش پیشنهادی را بر روی مجموعه داده‌های خروجی مدل پیش‌بینی میان مقیاس WRF در محدوده

## منابع

- 13- Hinton, G. E., Osindero, S. & Teh, Y.-W., 2006. A fast learning algorithm for deep belief nets. *Neural Computation*, vol. 18, no. 7, p. pp. 1527–1554.
  - 14- Hinton, G. E. & Salakhutdinov, R. R., 2006. Reducing the dimensionality of data with neural networks. *Science* 313 (5786), p. 504–507.
  - 15- Ibrahim, M., Ibrahim, F. & Al-Jumaily, A. A., 2018. Auto-Encoder based Deep Learning for Surface Electromyography Signal Processing. *Advances in Science, Technology and Engineering Systems Journal*, Volume Vol. 3, No. 1, pp. 94-102.
  - 16- Ioffe, S. & Szegedy, C., 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *International Conference on Machine Learning*, pp. 37: 448-56.
  - 17- Jiang, X., Zhang, Y., Zhang, W. & Xiao, X., 2013. A novel sparse auto-encoder for deep unsupervised learning. in: *Proceedings of the ICACI*.
  - 18- Jordan, M. I. & Mitchell, T. M., 2015. Machine learning: trends, perspectives, and prospects. *Science*, p. 349:255–60.
  - 19- Khodayar, M., Kaynak, O. & Khodayar, M. E., 2017. Rough Deep Neural Architecture for Short-term Wind Speed Forecasting. *IEEE*, pp. 1551-3203 (c).
  - 20- Kumar, V., Nandi, G. & Kala, R., 2014. Static Hand Gesture Recognition using Stacked Denoising Sparse Autoencoders. *Robotics and Artificial Intelligence Laboratory.IEEE*, pp. 978-1-4799-5173-4.
  - 21- LeCun, Y., Bengio, Y. & Hinton, G., 2015. Deep learning. *Nature*, p. 521:436–44.
  - 22- Lee MK, H. D., 2012. Dimensionality reduction of radio map with nonlinear autoencoder. *Electron Lett*, 48(11), pp. 655-657.
  - 23- Lemme A, R. R. S. J., 2012. Online learning and generalization of parts-based image representations.. *Neural Netw*, Volume 33, pp. 194-203.
  - 24- Lin, S.-Y., Chiang, C.-C., Hung, Z.-S. & Zou, Y.-H., 2017. A Dynamic Data-Driven Fine-Tuning Approach for Stacked Auto-Encoder Neural Network. s.l., *The Fourteenth IEEE International Conference on e-Business Engineering*, Department of Information Management.
  - 25- Liu J, C. G. L. Z. L. Y. L. H. L. X., 2013. Predicting protein structural classes with autoencoder neural networks. *IEEE*, Piscataway, pp. 1894-1899.
  - 26- Liu, R., Yang, B., Zio, E. & Chen, X., 2018. Artificial intelligence for fault
- 1- Ahmadi, A. , Daraiee, M. , Salamzadeh, A & Jafary, M. 1392. Artificial intelligence and business opportunities: identifying the functions of artificial intelligence in creating a competitive advantage for technology businesses(a study of the computer games industry).Publication in entrepreneurship development, 6th period,2nd issue, pp.7-26.
  - 2- Russakovsky , O. et al., 2015. ImageNet Large Scale Visual Recognition Challenge.. *International Journal of Computer Vision*, pp. 115(3):211-52.
  - 3- A. Voulodimos, N. D. A. D. a. E. P., 2018. Deep Learning for Computer Vision: A Brief Review. *Hindawi, Computational Intelligence and Neuroscience*, p. 7068349.
  - 4- Alain, G. & Bengio , Y., 2014. What regularized auto-encoders learn from the data-generating distribution.. *The Journal of Machine Learning Research*, pp. 15(1):3563-93.
  - 5- Barcelona & Memisevic , R., 2011. Gradient-based learning of higher-order image features.. *IEEE international conference on computer vision (ICCV)*, p. 1591–1598.
  - 6- Bengio , Y., Courville , A. & Vincent , P., *IEEE Transactions on* 2013. Representation learning: A review and new perspectives.. *Pattern Analysis and Machine Intelligence*, pp. 35(8):1798-828.
  - 7- Bishop, C. M., 2007. *Pattern Recognition and Machine Learning*. New York: (Information Science and Statistics), Springer.
  - 8- Chandra B, S. R., 2014. Adaptive noise schedule for denoising autoencoder. *Springer International Publishing*, pp. 535-542.
  - 9- Deng, J., Zhang, Z., Marchi, E. & Schuller, B., 2013. Sparse autoencoder based feature transfer learning for speech emotion recognition.. *IEEE*, New York, pp. 511-516.
  - 10- Ganin, Y. et al., 2016. Domain-adversarial training of neural networks. *Mach Learn Res*, p. 17:1–35.
  - 11- Guo, Y. et al., 2016. Deep learning for visual understanding, A review. *Neurocomputing* 187, p. 27–48.
  - 12- Hern´andez, E. et al., 2016. Rainfall Prediction: A Deep Learning Approach. in *International Publishing Switzerland* , Springer, pp. HAIS, LNAI 9648, pp. 151–162.

- 42- Tan CC, E. C., 2011. Using autoencoders for mammogram compression.. *J Med Syst*, 35(1), p. 4958.
- 43- Vidhya, K. & Geetha, T., 2017. Rough set theory for document clustering:A review. *Journal of Intelligent & Fuzzy Systems* 32, p. pp 2165–2185.
- 44- Vincent, . P., Larochelle, H., Bengio, Y. & Manzagol, P.-A., 2008. Extracting and composing robust features with denoising autoencoder. New York, USA 1, ACM Press, p. pp. 1096–1103.
- 45- Wu , R. et al., 2015. Deep Image: Scaling up Image Recognition.. *CoRR.*, p. abs 1501.02876.
- 46- Y. Bengio, P. L. D. P. a. H. L., 2007. “Greedy layerwise layerwise training of deep networks.”. s.l., s.n.
- 47- Yang, . J., Yu , K., Gong , Y. & Huang, . T., 2009. Linear spatial pyramid matching using sparse coding for image classification.. s.l., s.n.
- 48- Zhang, J., Shan, S., Kan, M. & Chen, X., 2014. Coarse-to-fine auto-encoder networks (cfan) for real-time face alignment. in: *Proceedings of the ECCV*.
- 49- Zhang, T., Liu, D. & Yue, D., 2017. Rough Neuron based RBF Neural Networks for Short-Term Load Forecasting. s.l., s.n.
- 50- Zhou, Y., Arpit, D., Nwogu, I. & Govindaraju, V., 2014. Is joint training better for deep autoencoders?. pp. 1405,1380.
- 51- Zuo,. H. et al., 2015. Transfer learning in hierarchical feature spaces. Taipei, Taiwan, ACM, p. 183–188.
- diagnosis of rotating machinery: A review.. *Mechanical Systems and Signal Processing* 108, p. 33–47.
- 27- Meng L, 2016. Research of stacked denoising sparse autoencoder. Springer.
- 28- Meng, L., Ding, S., Zhang, N. & Zhang, J., 2016. Research of stacked denoising sparse autoencoder. Springer, The Natural Computing Applications Forum.
- 29- Mesnil, G. et al., 2012. Unsupervised and Transfer Learning Challenge: a Deep Learning Approach. *ICML Unsupervised and Transfer Learning.* pp. 27:97-110.
- 30- Miotto, R. et al., 2017. Deep learning for healthcare: review, opportunities and challenges. Published by Oxford University Press , Oxford , Briefings in Bioinformatics, p. 1–11.
- 31- M, K., 2017. Rough Deep Neural Architecture for Short-term Wind Speed Forecasting.. *IEEE Okyay Kaynak*, pp. 1551-3203.
- 32- Murphy, K. P., 2012. Machine learning: a probabilistic perspective. MIT press.
- 33- Olshausen , B. & Field , D., 1997. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision research.* pp. 37(23):3311-25.
- 34- OLSHAUSEN, B. A. & FIELD, D. . J., 1997. Sparse Coding with an Overcomplete Basis Set: Strategy Employed by V1?. Pergamon, Volume vol. 37, no. 23, pp. pp. 3311-3325.
- 35- P. Lingras, 1996. “Rough Neural Networks.”. *IPMU*, pp. 1445-1450.
- 36- Quoc, V. L., 2013. Building high-level features using large scale unsupervised learning. in: *Proceedings of the ICASSP*.
- 37- Raina , R., Battle , A., Lee , H. & Packer , B., 2007. Self-taught learning: transfer learning from unlabeled data.. s.l., s.n.
- 38- Rifai , S. et al., 2011. Contractive autoencoders: Explicit invariance during feature extraction. s.l., s.n.
- 39- Sagha, H., Cummins, N. & Schuller, B., 2017. Stacked denoising autoencoders for sentiment analysis: a review. *WIREs Data Mining Knowl Discov*, p. e1212.
- 40- Shen, C., 2017. A trans-disciplinary review of deep learning research for water resources scientists. *Civil and Environmental Engineering*, Pennsylvania State University, University Park, p. PA16802.
- 41- Srivastava, N. et al., 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, pp. 15: 1929-58.