

پیش بینی قیمت برق در نیروگاه سیکل ترکیبی شهدای پاکدشت

با بهره گیری شبکه عصبی بیزین

بابک کاشانی زاده، فرید بشیری، محسن امیری، حمیدرضا مقدم

شرکت همانیرو، شرکت تولید برق دماوند

در روشهای پیش بینی قیمت متداول همچون انواع شبکه عصبی و سری های زمانی الگوریتم پیش بینی غالباً به گونه ای می باشد که میانگین و واریانس تابع گوسی را به عنوان پیش بینی قیمت هر ساعت ارائه می کند که این موضوع نمی تواند فیدبک صحیحی از وضعیت قیمت در آن ساعت باشد و برای حصول به این دیدگاه لازم است از روشهای تولید سناریو بهره گیری شود که این روش موجب پیچیدگی امر گردیده و ماهیت ذاتی مکانیزم پیش بینی را تحت شعاع قرار می دهد. اما بهره گیری از شبکه عصبی بیزین موجب می گردد تا بتوان یک تابع احتمالی (گوسی و یا غیر گوسی) برای پیش بینی قیمت در هر ساعت ارائه نمود که می تواند تاثیر بسزایی در مکانیزم پیشنهاد قیمت نهایی داشته باشد. در این مقاله پیش بینی قیمت با روش شبکه عصبی بیزین و بر مبنای داده های قیمت یکی از شرکتهای بزرگ تولیدکننده برق در ایران انجام می پذیرد.

واژه های کلیدی: تجدید ساختار، پیش بینی قیمت، شبکه عصبی بیزین

۱- مقدمه :

یکی از مهم ترین ویژگی های سیگنال قیمت انرژی الکتریکی، بی ثباتی بسیار آن می باشد. وجود همین بی ثباتی است که پیش بینی قیمت را با دشواری های بسیار روبرو ساخته است. علت اصلی چنین بی ثباتی هایی راهبردهای انتخابی پیشنهاد قیمت توسط بازیگران بازار و همچنین خیل عظیمی از عدم قطعیت های مؤثر بر قیمت می باشد.

تا کنون روشهای گوناگونی بمنظور پیش بینی قیمت ارائه شده است روش ARIMA از ابتدایی ترین روشها بوده است که در این راه مورد استفاده قرار گرفته است. در این روش مصالحه ای بین مدل سازی رفتار سری زمانی مورد مطالعه و بالا بردن دقت پیش بینی می باشد. مرجع [۱] با تشریح چگونگی بکارگیری روش ARIMA در فرآیند پیش بینی قیمت، مدل مذکور را برای پیش بینی قیمت ۲۴ ساعت آینده بازارهای برق اسپانیا و کالیفرنیا مورد استفاده قرار داده است.

در ادامه مجموعه روش های هوشمند مورد استفاده در مراجع گوناگون پیش بینی قیمت، در ۳ دسته شبکه های عصبی مصنوعی ۱ (ANN) منطبق فازی و ترکیب این دو روش یعنی روش عصبی - فازی مورد استفاده قرار گرفته است. از جمله رایج ترین انواع شبکه های عصبی مصنوعی مورد استفاده در فرآیند پیش بینی قیمت، شبکه عصبی پرسپترون چند لایه ۲ (MLP) و شبکه های عصبی بازگشتی ۳ (RNN) می باشند [۲۰، ۱۷]

۲- شبکه عصبی بیزین :

شبکه های عصبی بیزین در مواردی که داده های یادگیری کم باشد، می توانیم پیش بینی های دقیقی را ارائه نماید. به همین دلیل این نوع شبکه ها به طور گسترده ای برای پیش بینی سری های زمانی به کار گرفته می شود.

قبل از هر چیزی باید به این نکته اشاره کرد که روش معمول پس انتشار خطا، که برای آموزش شبکه های عصبی MLP (پرسپترون چند لایه) استفاده می شود، از لحاظ آماری به صورت تخمین بیشترین احتمال وقوع ۴، تفسیر می شود. به عبارت دیگر در این روش، در ازای یک سری داده یادگیری، بهترین دسته از مقادیر وزنه های شبکه عصبی یافته می شود، بطوریکه تابع هزینه را حداقل کند. سپس از این پارامترهای بدست آمده برای حالت تست استفاده خواهد شد. در این چارچوب، مدل پیچیده تر به شکل بهتری بر داده های موجود منطبق می شود، ولی به این معنا نیست که برای داده های جدید نیز خطای کمتری تولید کند. در واقع مدل های خیلی ساده و خیلی پیچیده توانایی کمتری برای تعمیم پذیری دارند.

در یادگیری بیزین یک تابع توزیع احتمال روی فضای وزنها در نظر گرفته می شود که نشان دهنده میزان باور نسبی ما به مقادیر متفاوت وزنه های شبکه می باشد. قبل از در نظر گرفتن داده های یادگیری، این باور توسط توزیع پیشین $p(w)$ روی وزنه های شبکه بیان می شود. $\underline{w} = [w_1, w_2, \dots, w_N]$ برداری شامل کلیه پارامترهای شبکه اعم از وزنها و بایاسها می باشد. اگر زوج داده های

¹- Artificial Neural Networks

²- Multi-layer Perceptron

³- Recurrent Neural Networks

⁴ Maximum Likelihood

ما نیز از ساده ترین و یکی از کارآمدترین توابع، یعنی تابع توزیع گوسی با میانگین صفر برای وزنهای شبکه استفاده کرده و آن را به این شکل نمایش می‌دهیم

$$p(\underline{w}) = \frac{1}{Z_w(\alpha)} \exp\left(-\frac{\alpha}{2} \|\underline{w}\|^2\right) \quad (4)$$

که $Z_w(\alpha)$ ثابت نرمال سازی است و به گونه ای محاسبه می‌شود که $p(\underline{w})$ تابع توزیع باشد. توابع هموار قدرت تعمیم پذیری بیشتری نسبت به انواع پیچیده تر دارند. یکی از دلایل استفاده از تکنیک تنظیم (رگولار سازی) نیز رسیدن به نگاهی هموار برای شبکه عصبی است. همانطور که گفته شد، این کار با در نظر گرفتن مقادیر کوچک برای وزنهای شبکه امکان پذیر است. در واقع تابع توزیع گوسی مانند فانکشنال تنظیم کاهش وزن عمل می‌کند. α نیز معکوس واریانس توزیع پیشین وزنهای شبکه است و نقشی مشابه ضریب تنظیم را بر عهده دارد. از آنجا که این ضریب، کنترل توزیع پارامترهای دیگر مثل وزنها و بایاسها را بر عهده دارد، به آن ابر پارامتر^۶ می‌گویند.

$$E_w = \frac{1}{2} \|\underline{w}\|^2 = \frac{1}{2} \sum_{i=1}^N w_i^2 \quad (5)$$

N تعداد کل پارامترهای شبکه است. زمانی که $\|\underline{w}\|$ بزرگ باشد E_w بزرگ خواهد بود، لذا $p(\underline{w})$ کوچک می‌شود و به این معنا می‌باشد که بزرگ بودن وزنهای شبکه دور از انتظار است. گوسی بودن تابع توزیع پیشین این مزیت را دارد که ثابت نرمال‌ساز آن به سادگی محاسبه می‌شود

$$Z_w(\alpha) = \left(\frac{2\pi}{\alpha}\right)^{N/2} \quad (6)$$

در قدم بعدی لازم است که مدلی برای نویز جمع شونده در خروجی در نظر بگیریم. در حالت کلی می‌توانیم، تابع احتمال وقوع $p(D|\underline{w})$ را به فرم نمایی زیر در نظر بگیریم

$$p(D|\underline{w}) = \frac{1}{Z_D(\beta)} \exp(-\beta E_D) \quad (7)$$

E_D در رابطه اخیر تابع خطا و β ابر پارامتر دیگری است که مدل نویز را کنترل می‌کند. $Z_D(\beta)$ نیز عامل نرمال ساز است و به طریق زیر محاسبه می‌شود

$$Z_D(\beta) = \int \exp(-\beta E_D) dD \quad (8)$$

اگر نویز جمع شونده در خروجی را گوسی در نظر بگیریم، برای یک زوج داده یادگیری مثل (y, \underline{x}) می‌توان نوشت

$$p(y|\underline{x}, \underline{w}) \propto \exp\left(-\frac{\beta}{2} (y - f(\underline{x}, \underline{w}))^2\right) \quad (9)$$

یادگیری $D = \{(\underline{x}_1, y_1), \dots, (\underline{x}_n, y_n)\}$ باشند، تابع توزیع پسین وزنهای شبکه از روی قانون بیز به طریق زیر یافته می‌شود.

$$P(\underline{w}|D) = \frac{P(\underline{w})P(D|\underline{w})}{P(D)} \quad (2)$$

که مخرج کسر، عامل نرمال ساز است و می‌تواند به شکل زیر نوشته شود

$$P(D) = \int P(D|\underline{w})P(\underline{w}) d\underline{w} \quad (3)$$

$P(D|\underline{w})$ نشان دهنده مدل نویز روی داده های خروجی است و معادل تابع احتمال وقوع^۱ در روش قبل است. روند یادگیری در روش بیزین به این شرح است که ابتدا با یک تابع توزیع پیشین روی وزنهای شبکه شروع می‌کنیم. از آنجا که در این مرحله اطلاعات کمی راجع به مقدار وزنها داریم، این توزیع گسترده می‌باشد. با مشاهده داده ها و از طریق رابطه (۲) تابع توزیع پسین وزنها محاسبه می‌شود که گستردگی کمتری نسبت به قبل خواهد داشت. همانطور که مشخص است انتخاب تابع توزیع پیشین نقش مهمی در یادگیری بازی می‌کند.

۲-۱- تابع توزیع پیشین

تابع توزیع پیشین، بازتابی از ذهنیت قبلی ما راجع به مقادیری است که آن کمیت اختیار می‌کند. ولی در مورد وزنهای شبکه های عصبی، ذهنیتی راجع به اینکه چه مقداری باید داشته باشند نداریم. نیل در [۸] نشان داد که برای شبکه عصبی با یک لایه مخفی که توابع تحریک تانژانت هیپربولیک دارند، استفاده از توزیع گوسی برای تابع توزیع پیشین وزنهای لایه مخفی به خروجی، به یک پروسه گوسی^۲ برای تابع خروجی منجر خواهد شد. این پروسه، با توجه به واریانس توزیع پیشین گوسی وزنهای لایه ورودی به لایه مخفی، می‌تواند هموار^۳، براونی^۴ و یا قسمتی براونی^۵ باشد. اگر مقدار این واریانس خیلی بزرگ باشد، همواره در نواحی اشباع تابع تحریک نرون خواهیم بود و رفتار خروجی، براونی خواهد بود.

همچنین ثابت کرده است که اگر معکوس واریانس توابع گوسی لایه مخفی به خروجی متناسب با تعداد نورونهای لایه مخفی باشد، می‌توان نورونهای لایه میانی را به هر تعداد زیاد کرد ولی شبکه دچار یادگیری بیش از حد نشود. در واقع با افزایش نورونهای لایه مخفی، واریانس وزنهای لایه خروجی کاهش می‌یابد.

¹ Likelihood function

² Gaussian process

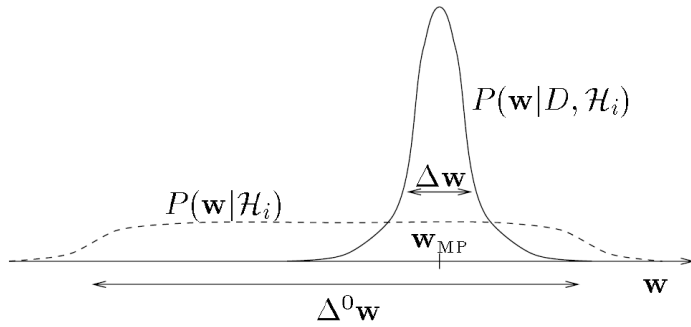
³ Smooth

⁴ Brownian

⁵ Fractional Brownian

⁶ Hyper parameter

به سمت w_{MP} خواهد رفت ولی اگر تعداد داده های یادگیری کم باشد، بخش دوم عبارت (۱۴) که مربوط به دانش قبلی در مورد وزنهای شبکه است نقش زیادی در یافتن w_{MP} بازی خواهد کرد. یادگیری بیزین با توابع توزیع گوسی پیشین را می توان به صورت شماتیک به صورت شکل ۱-۱ نمایش داد. همانطور که دیده می شود، واریانس تابع پسین نسبت به پیشین کاهش یافته است. مقدار w_{MP} جایی است که توزیع پسین حداکثر می شود. ولی در عمل ممکن است این توزیع بیش از یک مقدار بیشینه داشته باشد.



شکل ۱: یادگیری بیزین برای وزنهای شبکه

۳-۲- تقریب گوسی برای توزیع پسین

با توجه به مدل گوسی انتخابی برای نویز و توابع پیشین وزنهای شبکه، رابطه توزیع پسین در عبارت (۱۳) و (۱۴) بیان شد. در ادامه خواهیم دید که برای محاسبه تابع توزیع پیش بینی خروجی نیاز به محاسبه انتگرال توزیع پسین روی فضای کلیه وزنهای شبکه داریم. برای اینکه این انتگرال به طور تحلیلی قابل محاسبه باشد، نیاز به فرضیه های ساده کننده داریم. [۹] این کار را با فرض تقریب گوسی برای تابع توزیع پسین انجام داد. به این منظور بسط تیلور $S(w)$ را حول مقدار w_{MP} تا مرتبه دوم می نویسیم

$$S(\underline{w}) = S(\underline{w}_{MP}) + \frac{1}{2}(\underline{w} - \underline{w}_{MP})^T A (\underline{w} - \underline{w}_{MP}) \quad (15)$$

ماتریس A هسیان تابع خطای کلی S به ازای w_{MP} است.

$$A = \nabla \nabla S_{MP} = \beta \nabla \nabla E_D^{MP} + \alpha I \quad (16)$$

اکنون می توانیم تابع توزیع پسین را به صورت گوسی نمایش دهیم

$$p(\underline{w}|D) = \frac{1}{Z_S^*} \exp(-S(\underline{w}_{MP}) - \frac{1}{2} \Delta \underline{w}^T A \Delta \underline{w}) \quad (17)$$

که Z_S^* نیز ثابت نرمال ساز در حالت تقریب گوسی است. اگر تعداد داده ها به سمت بینهایت میل کند، توزیع پسین نیز به سمت توزیع گوسی خواهد رفت. در عمل نیز اگر تعداد داده ها خیلی زیاد باشد می توان توزیع گوسی را تقریب مناسبی

که f نشان دهنده نگاهی از فضای ورودی به خروجی شبکه عصبی است. β معکوس واریانس نویز است و از همین طریق مدل نویز را کنترل می کند. اگر برای همه داده ها نویز را گوسی و آنها را مستقل از هم فرض کنیم، می توان نوشت

$$p(D|\underline{w}) = \prod_{i=1}^n p(y_i | \underline{x}_i, \underline{w}) = \frac{1}{Z_D(\beta)} \exp\left(-\frac{\beta}{2} \sum_{i=1}^n (y_i - f(\underline{x}_i, \underline{w}))^2\right) \quad (10)$$

همانطور که مشاهده می شود، با فرض نویز گوسی در خروجی شبکه، تابع خطای E_D به صورت مجموع مربعات خطا روی داده های یادگیری خواهد بود. ثابت نرمال ساز نیز برابر خواهد بود با

$$Z_D(\beta) = \left(\frac{2\pi}{\beta}\right)^{n/2} \quad (11)$$

اکنون می توان نشان داد که روش مرسوم یادگیری در شبکه های عصبی که با استفاده از الگوریتم های مبتنی بر گرادیان، تابع هزینه را حداقل می کند و به نوعی بیانگر تخمین بیشترین احتمال وقوع (w_{ML}) می باشد، فقط از اطلاعات $p(D|\underline{w})$ استفاده می کند. این مطلب را می توان با یافتن تابع احتمال وقوع نشان داد

$$L(\underline{w}) = \ln p(D|\underline{w}) = \quad (12)$$

$$-\sum_{i=1}^n \frac{\beta}{2} |y_i - f(\underline{x}_i, \underline{w})|^2 + C$$

C ثابتی است که به w بستگی ندارد. حداکثر کردن این تابع، در واقع همان حداقل کردن مجموع مربعات خطا می باشد.

۲-۲- تابع توزیع پسین

برای یافتن تابع توزیع پسین وزنهای شبکه، کافی است که روابط (۴) و (۱۰) را در قانون بیز (۲) جایگذاری کنیم. در این صورت داریم

$$p(\underline{w}|D) = \frac{1}{Z_S(\alpha, \beta)} \exp(-\beta E_D - \alpha E_w) \quad (13)$$

$$= \frac{1}{Z_S(\alpha, \beta)} \exp(-S(\underline{w}))$$

که در آن Z_S ثابت نرمال ساز است و S نیز عبارتست از

$$S(\underline{w}) = \beta E_D + \alpha E_w = \quad (14)$$

$$\frac{\beta}{2} \sum_{i=1}^n (y_i - f(\underline{x}_i, \underline{w}))^2 + \frac{\alpha}{2} \sum_{i=1}^N w_i^2$$

در این مرحله یافتن یک مقدار یکتا (w_{MP}) به عنوان بهترین وزن شبکه، معادل حداقل کردن مجموع مربعات خطا به همراه فانکشنال تنظیم کاهش وزن است. با دقت در عبارت (۱۴) ملاحظه می شود که مقدار بخش اول، با افزایش داده های یادگیری زیاد می شود، در صورتیکه بخش دوم ثابت باقی می ماند. بنابراین می توان گفت که اگر α, β ثابت باشند، با افزایش داده های یادگیری تخمین w_{MP}

$$P(\underline{w}|D, \underline{x}_{n+1}) = P(\underline{w}|D) \quad (21-1)$$

با توجه به رابطه اخیر، می‌توانیم رابطه (۲۰-۱) را بازنویسی کنیم

$$P(y_{n+1}|\underline{x}_{n+1}, D) = \quad (22)$$

$$\int_{\mathbb{R}^N} P(y_{n+1}|\underline{x}_{n+1}, D, \underline{w})P(\underline{w}|D)d\underline{w}$$

عبارت اول داخل انتگرال، همان مدل نویز برای داده تست بوده که در رابطه (۹) بیان شد و عبارت دوم نیز تابع توزیع پسین وزنه‌های شبکه می‌باشد. می‌توانیم میانگین تابع اخیر را به عنوان خروجی نقطه ای در نظر بگیریم

$$\hat{y} = E\{P(y_{n+1}|\underline{x}_{n+1}, D)\} = \quad (23)$$

$$\int_{\mathbb{R}^N} f(\underline{x}_{n+1}, \underline{w})P(\underline{w}|D)d\underline{w}$$

برای محاسبه تابع توزیع پیش بینی خروجی (رابطه ۲۲) به کمک تقریب گوسی (رابطه ۱۷) می‌توانیم بنویسیم

$$P(y_{n+1}|\underline{x}_{n+1}, D) \propto \quad (24)$$

$$\int_{\mathbb{R}^N} \exp\left(-\frac{\beta}{2}(y_{n+1} - f(\underline{x}_{n+1}, \underline{w}))^2\right) \times$$

$$\exp\left(-\frac{1}{2}\Delta \underline{w}^T A \Delta \underline{w}\right) d\underline{w}$$

اگر عرض تابع توزیع پسین را خیلی باریک فرض کنیم، می‌توانیم از تقریب مرتبه اول تابع غیرخطی f ، حول \underline{w}_{MP} استفاده کنیم. به این ترتیب می‌توانیم رابطه ۲۴ را به این شکل بنویسیم

$$P(y_{n+1}|\underline{x}_{n+1}, D) = \quad (25)$$

$$\frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left(-\frac{(y_{n+1} - f(\underline{x}_{n+1}, \underline{w}_{MP}))^2}{2\sigma^2}\right)$$

که در آن واریانس برابر است با

$$\sigma^2 = \frac{1}{\beta} + g^T A^{-1} g \quad (26)$$

و g گرادیان تابع f در \underline{w}_{MP} است. عبارت ۲۶ از دو بخش تشکیل شده است. بخش اول به مشخصات نویز خروجی بستگی دارد و بخش دوم آن ناشی از عرض تابع توزیع پسین پارامترهای شبکه است. اگر نویز خروجی خیلی زیاد (یعنی β کم باشد) و یا توزیع پسین خیلی باریک باشد، در اینصورت عامل غالب بخش اول است و واریانس تابع توزیع پیش بینی خروجی توسط واریانس نویز تعیین می‌شود. ولی اگر میزان نویز کم (β بزرگ) و یا توزیع پسین خیلی گسترده باشد، در اینصورت عامل غالب بخش دوم بوده و واریانس تابع توزیع پیش بینی خروجی توسط عرض توزیع پسین مشخص می‌گردد.

برای توزیع پسین دانست ولی زمانی که تعداد داده ها کم باشد این طور نیست و نیاز به روش دیگری خواهد بود.

در مرجع [۷]، اثبات شده است که ثابت نرمال ساز بر حسب دترمینان ماتریس A برابر خواهد بود با

$$Z_S^*(\alpha, \beta) = e^{-S(\underline{w}_{MP})} (2\pi)^{N/2} |A|^{-1/2} \quad (18)$$

برای یک نگاشت غیرخطی مثل آنچه که یک شبکه عصبی انجام می‌دهد، این امکان وجود دارد که تابع خطا بیش از یک مینیمم محلی داشته باشد. بعضی از آنها ممکن است به دلیل تقارن موجود در ساختار شبکه به وجود آمده باشند. به عنوان مثال برای یک شبکه MLP با توابع تحریک تانزانت هیپربولیک، هر مینیمم محلی به خانواده ای از وزنها شامل $2^H H!$ عنصر تعلق دارد (H تعداد نورونهای میانی است). این عدد به دلیل قابلیت جابجایی نورونهای مخفی و نیز تغییر علامت وزنه‌های متصل به هر نورون مخفی می‌باشد. به غیر از اعضای این خانواده، امکان وجود خانواده های دیگر نیز وجود دارد. بنابراین در نظر گرفتن یک تابع گوسی تکی نمی‌تواند همه این نقاط محلی را پوشش دهد. لذا از مجموع چند تابع گوسی که هر یک از آنها بر روی یکی از نقاط محلی تمرکز دارد، استفاده می‌شود. در این حالت می‌توان تابع توزیع پسین را به شکل زیر نمایش داد

$$p(\underline{w}|D) = \sum_i p(\underline{w}|m_i, D)p(m_i|D) \quad (19)$$

m_i نشان دهنده مینیمم های محلی غیر معادل است. $p(\underline{w}|m_i, D)$ نیز تابعی گوسی حول هر یک از این نقاط است. به این ترتیب پیش بینی نهایی خروجی شبکه نیز، جمع وزندار چند مدل مختلف خواهد بود. اگر شبکه کوچک باشد، m_i ها را می‌توان با چندین بار اجرای فرآیند یادگیری از شرایط اولیه متفاوت یافت ولی در عمل و برای شبکه های بزرگ این کار امکان ندارد. همچنین تعیین ضریب $p(m_i|D)$ نیز مشکل است. در مجموع این روش دقت خوبی ندارد.

۳- تابع توزیع خروجی شبکه

همانطور که دیدیم، در چارچوب بیزین شبکه آموزش دیده با تابع پسین پارامترها نمایندگی می‌شود. بنابراین خروجی شبکه نیز به صورت تابع توزیع خواهد بود. اگر ورودی تست \underline{x}_{n+1} باشد، خروجی آن برابر خواهد بود با

$$P(y_{n+1}|\underline{x}_{n+1}, D) = \quad (20)$$

$$\int_{\mathbb{R}^N} P(y_{n+1}|\underline{x}_{n+1}, D, \underline{w})P(\underline{w}|D, \underline{x}_{n+1})d\underline{w}$$

چون تابع توزیع پسین فقط از روی داده های یادگیری به دست می‌آید، بنابراین داریم

$$p(\alpha, \beta | D) = \frac{p(D | \alpha, \beta) p(\alpha, \beta)}{p(D)} \quad (29)$$

در این مرحله به یک تابع توزیع پیشین برای پارامترهای α, β نیاز داریم. به چنین روندی که وزنه‌های شبکه توسط یک توزیع پیشین وابسته به α, β بیان می‌شود و خود این کمیتها نیز در یک سطح بالاتر توسط توزیع پیشین دیگری نمایندگی می‌شوند، مدل‌های سلسله مراتبی^۴ می‌گوییم. تعداد این سطوح باز هم می‌تواند افزایش یابد. اگر هیچ ایده ای راجع به مقادیری که α, β اختیار می‌کنند نداریم، تابع توزیع یکنواخت را انتخاب می‌کنیم به گونه ای که به تمام مقادیر ممکن، وزن یکسانی اختصاص دهد. به چنین تابعی غیر مفید^۵ گفته می‌شود. در چنین حالتی مخرج رابطه ۲۹ به مقدار α, β بستگی نداشته و لذا حداکثر مقدار توزیع پسین با حداکثر کردن عبارت $p(D|\alpha, \beta)$ حاصل می‌شود.

۴- ورودی‌های پیش‌بینی قیمت

در این متد روند پیش بینی انجام گرفته به صورت ساعتی بوده، یعنی از یک شبکه عصبی کوچک به طور مجزا برای پیش‌بینی هر ساعت استفاده می‌کنیم. لذا پیش‌بینی‌ها از یکدیگر مستقل بوده و می‌توانند به صورت موازی با یکدیگر نیز انجام شوند. علاوه بر ساده‌تر بودن ساختار شبکه در این حالت، از قابلیت تعمیم بهتری به لحاظ تعریف ورودیها نیز برخوردار هستیم. یعنی همواره قادریم ورودیهای دلخواهی را به مجموعه ورودیهای موثر برای پیش بینی قیمت هر ساعت اضافه یا کم کنیم.

به این منظور ابتدا یک دسته بزرگ از انواع ورودی‌های ممکن که می‌تواند روی خروجی موثر باشد، تعریف کردیم. این دسته بزرگ به طور کلی می‌تواند شامل داده‌های قیمت سه ماه گذشته، پیش‌بینی بار، ظرفیت خطوط، تبادلات ساعتی خطوط برای سه ماه گذشته و نوع روز می‌باشد. طبق بررسیهای انجام شده به جای اینکه با مقادیر مطلق کمیت‌های قیمت سر و کار داشته باشیم، بهتر دیدیم که از مقادیر نسبی آنها استفاده کنیم.

۵- استفاده از روش (ARD) برای کاهش ورودی‌های شبکه

از آنجایی که با مجموعه داده های محدودی کار می‌کنیم، یک همبستگی تصادفی میان ورودی های نامربوط و خروجی به وجود می‌آید. به این ترتیب هیچ شبکه عصبی معمولی (حتی با در نظر گرفتن کاهش وزن) قادر نیست که ضرایب مربوط به این ورودی را صفر کند و لذا این ورودی نامطلوب، عملکرد مدل را پایین می‌آورد. به خصوص وقتی که تعداد متغیرها زیاد و تعداد داده ها کم باشد. بنابراین انتخاب بهترین ورودیها یک مسئله مهم می‌باشد. این روش

در هر صورت نکته مهم این است که برخلاف روشهای قبلی که یک وزن بهینه را در نظر می‌گرفتند، در اینجا طبق رابطه (۲۰) اثر همه وزنها در محاسبه خروجی دخالت داده می‌شود. در حالتی نیز که از تقریب گوسی استفاده می‌شود، ابتدا با حداقل کردن تابع خطای $S(w)$ ، وزن w_{MP} یافته می‌شود، سپس از طریق رابطه (۲۰) واریانس تابع توزیع پیشینی خروجی محاسبه می‌شود.

۳-۱- یافتن مقادیر α, β

تا کنون در این مقاله مقادیر این پارامترها را ثابت در نظر گرفتیم. ولی در عمل ممکن است در مورد مقدار مناسب آنها اطلاعاتی نداشته باشیم. این موضوع به خصوص در مورد α صدق می‌کند. مرجع [۱۰] نشان داده است که اگر α خیلی بزرگ باشد، واریانس توزیع پیشین وزنه‌های شبکه خیلی کم خواهد شد. خروجی چنین مدل ساده ای خیلی هموار خواهد بود و امکان هموار بودن بیش از حد^۱ خروجی وجود دارد به نحوی که اصلاً بر داده ها منطبق نباشد. برعکس این مسئله نیز صادق است. یعنی اگر α خیلی کوچک باشد، واریانس توزیع پیشین وزنه‌های شبکه خیلی بزرگ خواهد شد. خروجی چنین مدل پیچیده ای خیلی نوسانی بوده و امکان انطباق بیش از حد^۲ خروجی وجود دارد.

در چارچوب بیزین این امکان وجود دارد که کلیه مقادیر این کمیتها را به حساب بیاوریم. این کار با انتگرال گیری انجام خواهد شد. مثلاً تابع توزیع پسین وزنه‌های شبکه برابر خواهد بود با

$$p(\underline{w}|D) = \iint p(\underline{w}|\alpha, \beta, D) p(\alpha, \beta|D) d\alpha d\beta \quad (27)$$

دو روش برای مواجهه با این انتگرال وجود دارد. یکی حل تحلیلی آن و دیگری روشی به نام تقریب شواهد^۳، که در [۹، ۱۰] مطرح شده است. در این روش فرض می‌شود که توزیع $p(\alpha, \beta|D)$ حول مقادیر α_{MP}, β_{MP} خیلی تیز می‌باشد. لذا رابطه (۲۷) را می‌توان به فرم زیر نوشت

$$p(\underline{w}|D) \approx \quad (28)$$

$$p(\underline{w}|\alpha_{MP}, \beta_{MP}, D) \iint p(\alpha, \beta|D) d\alpha d\beta = p(\underline{w}|\alpha_{MP}, \beta_{MP}, D)$$

این روش به ما می‌گوید که می‌توانیم ابتدا مقادیری از α, β را که تابع توزیع پسین خود را حداکثر می‌کنند، بیابیم و سپس بقیه عملیات را با این مقادیر ثابت انجام دهیم. برای یافتن این مقادیر، توزیع پسین این پارامترها را به کمک قانون بیز می‌تویسیم

⁴ Hierarchical models

⁵ Non-informative

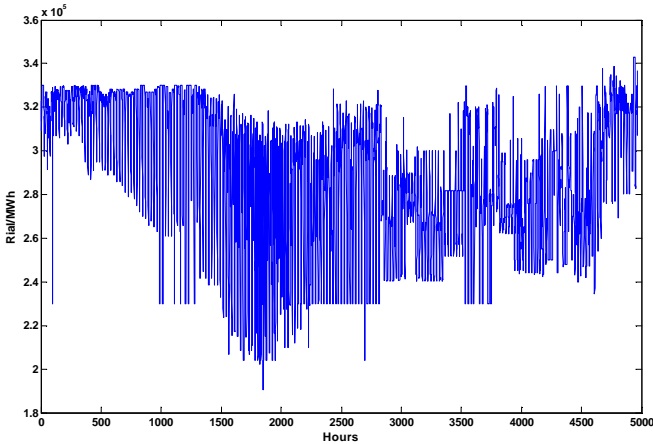
⁶ Automatic relevance determination

¹ Over smooth

² Over fit

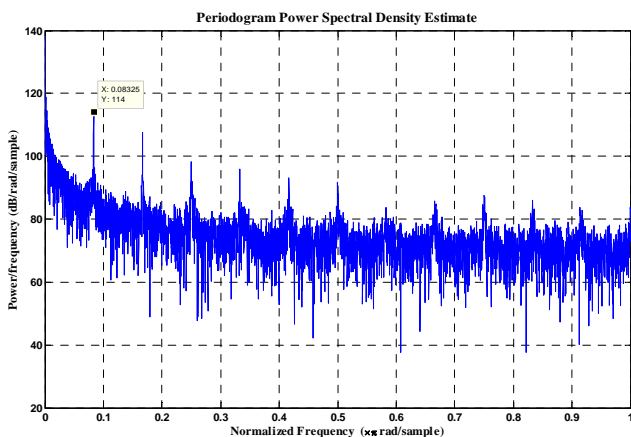
³ Evidence approximation

می‌دهند. در اینجا نیز ما روش خود را باروش سری زمانی مقایسه می‌کنیم. در حالت سری زمانی نرم‌افزار از تابع MATLAB استفاده می‌شود که سری زمانی بازگشت یک سری زمانی مفروض را بدست می‌آورد. از این تابع کمک گرفته و سری زمانی بازگشت قیمت برق مورد مطالعه را نیز بدست می‌آوریم. سری زمانی لگاریتم قیمت برق ایران و بازگشت قیمت برق ایران در شکل‌های (۲) نشان داده شده است.



شکل (۲) منحنی قیمت ساعتی برق ایران

دوره یا سیکل فصلی سری زمانی قیمت برق برابر با ≈ 12 می‌باشد. همچنین سایر پیک‌های محلی نیز بصورت مضربی از 0.08325 می‌باشند. لذا سری زمانی ما دارای دوره فصلی ۱۲ ساعته می‌باشد. همین عمل را برای اختلاف اول و بازگشت سری زمانی قیمت برق انجام می‌دهیم و به نتایج مشابه برای آنها می‌رسیم یعنی آنها نیز دارای خاصیت فصلی ۱۲ ساعته (نیم‌روزه) هستند. جالب اینجاست که علیرغم استفاده از روش تعیین اختلاف برای سری زمانی قیمت برق که منجر به تشکیل دو سری زمانی اخیر شده است، اما دوره فصلی در آنها همچنان وجود دارد.



شکل (۳) پریودگرام قیمت ساعتی برق ایران

اما در مدل ارائه شده در این مقاله نتایج این الگوریتم بصورت یک تابع گوسی برای هر ساعت ارائه گردیده که نشان دهنده قیمت و میزان احتمال آن قیمت در هر نقطه می‌باشد.

به طور موثری در عمل به کار رفته است و نتایج، حاکی از برتری این روش نسبت به روشهایی مثل آنالیز مولفه های اصلی (PCA) و همبستگی متقابل می‌باشد [۱۷-۱۹].

در روش ARD، یک توزیع پیشین برای پارامترها در نظر می‌گیریم به طوریکه نشان دهنده عدم قطعیت در تاثیر یک ورودی باشد. به این منظور کافی است برای هر ورودی یک پارامتر مثل α معرفی کنیم که نشان دهنده کاهش وزن باشد. یعنی مقادیر بزرگ آن مانع از تاثیر آن ورودی در خروجی خواهد شد و لذا از انطباق بیش از حد جلوگیری می‌کند. به عبارت دیگر این پارامتر، معکوس واریانس توزیع پیشین وزنهای متصل شده به هر ورودی است.

$$p(\{\underline{w}_i\}|\alpha_c, H) = \quad (30)$$

$$\frac{1}{Z_{w(c)}} \exp(-\sum_c \alpha_c E_{w(c)}) \quad c=1,2,\dots,C$$

C نشان دهنده تعداد کلاسهای متفاوتی است که در نظر می‌گیریم. در اینجا C همان تعداد ورودیها خواهد بود. چون از هر ورودی به تمام نورونهای میانی یک وزن متصل می‌باشد، بنابراین بعد بردار وزن متعلق به هر کلاس، برابر تعداد نورونهای میانی است. با فرض توزیع گوسی برای هر وزن می‌توان نوشت

$$E_{w(c)} = \sum_{\substack{i=1 \\ w \in c}}^{n_{\text{hidden}}} |w_i|^2 / 2 \quad (31)$$

به این ترتیب فرمول تجدید مقادیر α, β به شکل زیر تغییر خواهد کرد

$$\alpha_c^{MP} = \frac{\gamma_c}{2E_{w(c)}^{MP}} \quad (32)$$

منظور از \underline{w}^{MP} در اینجا دسته وزنی است که تابع هزینه $M = \beta E_D + \sum_c \alpha_c E_{w(c)}$ را حداقل می‌کند. γ_c نیز تعداد موثر

پارامترها در هر کلاس بوده و از رابطه زیر حساب خواهد شد

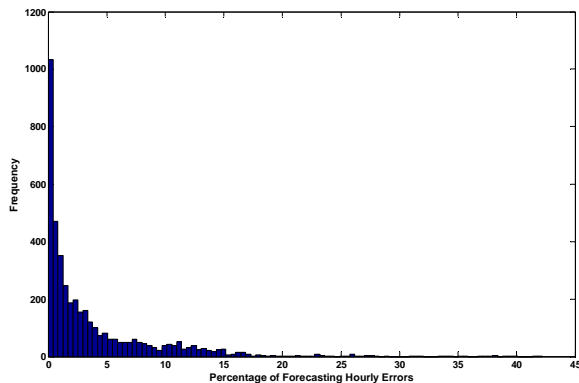
$$\gamma_c = N_c - \alpha_c \text{Tr}_c(A^{-1}) \quad (33)$$

منظور از N_c تعداد پارامترهای هر کلاس می‌باشد که در اینجا برابر تعداد نورونهای میانی می‌باشد. مجموع عناصر روی قطر نیز فقط برای اعضای همین کلاس محاسبه خواهد شد.

۶- تست الگوریتم مورد مطالعه

بمنظور تست الگوریتم از داده‌های یکی از بزرگترین نیروگاههای تولید کنند انرژی در ایران سیکل ترکیبی شهدای پاکدشت (دماوند) استفاده می‌گردد. این الگوریتم با داده‌های این شرکت در سال ۱۳۹۲ مورد تست گردیده است. در مجموع ۴۹۶۸ مشاهده از قیمت برق داریم. شکل (۲) نمودار قیمت برق ایران را نشان می‌دهد. همانگونه که پیش‌تر بیان شد، در بازارهای مالی، سری‌های زمانی قیمت را معمولاً بر اساس لگاریتم یا بازگشت آنها مورد بررسی قرار

اطلاعات برای تخصیص پله‌های قیمت و مدیریت ریسک بهره‌گیری نمود که تاثیر بسزایی در پیشنهاد قیمت نهایی خواهد داشت و بترتیب میانگین و میانه خطاهای پیش‌بینی قیمت برق ایران کمتر از ۳٪ هستند و هیستوگرام درصد خطای پیش‌بینی قیمت ساعتی آن درنمای ذیل نمایش می‌دهد.



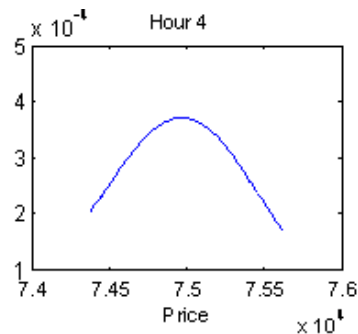
شکل (۶) هیستوگرام درصد خطای پیش‌بینی قیمت ساعتی

۷- نتیجه گیری

بررسی کلی نتایج و مقایسه نمودارهای هیستوگرام و پیش‌بینی قیمت بازار برق برای نیروگاه سیکل ترکیبی شهدای پاکدشت نشان می‌دهد که مدل شبکه عصبی بیزین با میزان خطا $MAPE=0.3$ روش مناسب برای پیش‌بینی قیمت در بازار برق ایران می‌باشد.

۸- مراجع :

- [1]. A.Lapedes, R.Farber, "Nonlinear signal processing using neural networks: prediction and system modeling", Technical Report LA-UR-87-2662, Los Alamos National Laboratory, Los Alamos, NM, 1987.
- [2]. G. Zhang, B. E. Patuwo, and M.Y. Hu, "Forecasting with artificial neural networks: The state of the art", Elsevier, Int. J. Forecast., vol. 14, pp. 35–62, 1998.
- [3]. L. Gorr, "Research prospective on neural network forecasting", International Journal of Forecasting, Vol. 10, pp. 1–4, 1994.
- [4]. F.Acernese, R.de rosa, L.milano, "A hierarchical Bayesian learning framework for autoregressive neural network modeling of time series", IEEE Proceedings of 3rd international Symposium on image and signal processing, 2003
- [5]. F.Acernese, A.eleuteri, l.milano, "A hierarchical Bayesian learning scheme for autoregressive neural network: application to the CATS benchmark", IEEE Proceedings, 2004
- [6] A. K. Palit, D. Popovic, Computational Intelligence in Time Series Forecasting, Theory and engineering applications, Springer, 2005.
- [7] M. Falk, A first course on Time series analysis, University of Wurzburg, Germany, 2006.
- [8] C. Chatfield, The analysis of time series: an introduction, 5th ed., chapman & hall, 1995.
- [9] R. H. Shumway, D. S. Stuffer, Time series analysis and its applications, 2nd ed., springer, 2007

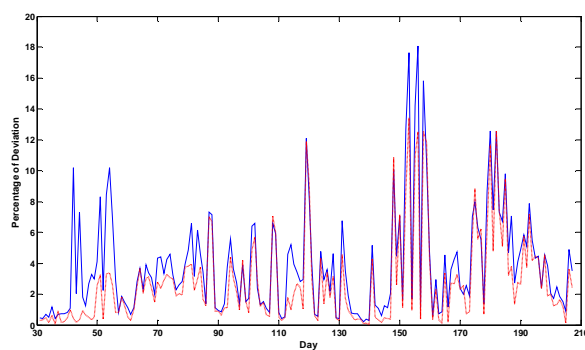


شکل (۴) منحنی پیش‌بینی قیمت برای یک ساعت

به منظور تست نتایج نتایج پیش‌بینی قیمت را برای یک روز جمعه که نوسانات قیمت در آن بیشتر از سایر روزها می‌باشد و برای یک روز اردیبهشت ماه ۹۲ نمایش منحنی خروجی در ازای نقاطی با بیشترین احتمال در هر ساعت به قرار ذیل بوده است. این نتایج برای ۶۰ نقطه احتمالاتی و مشتمل بر ضریب وزنی قیمت، ضریب احتمالاتی و ضریب چگالی قیمت می‌باشد.

Point	10	20	30	40	50	60
Price Factor	0.955	0.960	0.965	0.970	0.975	0.980
Corresponding spot price	5.484	6.421	7.370	8.292	9.144	9.884
Factor Price	0.955	0.960	0.965	0.970	0.975	0.980
Price Probability	5.240	6.167	7.114	8.044	8.915	9.685
Price Probability Normalize	0.003	0.004	0.004	0.005	0.005	0.006
Lost Cumulative probability	0.027	0.062	0.102	0.149	0.200	0.257
Win Cumulative probability	0.973	0.938	0.898	0.851	0.800	0.743

حال می‌توان داده‌های قیمت بازار را با هر دوروش سری زمان و شبکه عصبی بیزین ارائه شده در این مقاله مقایسه نمود. این مقایسه در شکل (۵) آمده است.



شکل (۵) مقایسه پیش‌بینی سری زمانی (آبی) و مدل شبکه عصبی بیزین بررسی نتایج نشان می‌دهد که میزان خطا در روش شبکه عصبی بیزین $MAPE=0.3$ می‌باشد که در مقایسه با روش سری زمانی و میزان خطا $MAPE=0.33$ همچنین بدلیل در اختیار بودن تابع احتمالی قیمت در هر ساعت از این