سیستم های فازی و کاربردها
سال هشتم، شماره اول (بهار و تابستان ۱۴۰۴)
صص. ۹۳ تا ۱۱۹
شاپای چاپی: ۲۷۱۷-۴۴۰۹
شاپای الکترونیکی: ۲۷۱۷-۳۹۸۴
DOI:10.22034/jfsa.2025.483155.1248

بررسی تأثیر انتخاب ابرپارامترها در پیش‌بینی سری زمانی با استفاده از شبکه‌های عصبی LSTM

محمدرضا پورمیر و نجمه قنبری*

گروه مهندسی کامپیوتر، دانشکده فنی و مهندسی، دانشگاه زابل، زابل، ایران

تاریخ دریافت: ۱۴۰۳/۰۷/۲۴ تاریخ پذیرش: ۱۴۰۴/۰۴/۲۴

نوع مقاله: علمی-پژوهشی

چکیده. یکی از مسائل مهم در حوزه داده‌کاوی پیش‌بینی سری‌های زمانی می‌باشد و در سال‌های اخیر توجه محققان زیادی را به خود جلب کرده است. معمولا داده‌های سری‌های زمانی مرتبط با زمان و دارای الگوهای خاصی می‌باشند که با دانستن این الگوها و استفاده از آنها امکان پیش‌بینی رویدادهای آینده فراهم می‌شود. یکی از رایج‌ترین شبکه‌های عصبی عمیق برای پیش‌بینی سری‌های زمانی شبکه عصبی حافظه طولانی کوتاه‌مدت (LSTM) می‌باشد. شبکه LSTM یک شبکه عصبی بازگشتی است که می تواند اطلاعات پیچیده مرتبط با زمان را حفظ و الگوهای پیچیده در سری‌های زمانی را درک کند. یکی از چالش‌های مهم شبکه‌های عصبی در پیش‌بینی سری‌های زمانی تعیین مقدار درست ابرپارامترها می‌باشد که بر کارایی و عملکرد آنها تاثیر زیادی دارد. ابرپارامترهای مختلفی می‌تواند در نظرگرفته شود. ابرپارامترهایی که در این مقاله در نظر گرفته شده‌اند عبارتند از: تعداد نرون‌ها در لایه پنهان، پارامتر پراکندگی، پارامتر کاهش وزن در تابع هزینه و وزن جریمه پراکندگی. برای تنظیم مقدار صحیح ابرپارامترها به گونه‌ای که بیشترین دقت در مساله حاصل شود از روش بهینه‌سازی بیزی استفاده (ادامه دارد)

2010 *Mathematics Subject Classification.* xxxxx ; xxxxx
* Corresponding author
E-mails: pormir@uoz.ac.ir, n.ghanbari@uoz.ac.ir.
عبارات و کلمات کلیدی. : پیش‌بینی سری‌های زمانی، ابر پارامترها، یادگیری عمیق، شبکه عصبی حافظه طولانی کوتاه مدت، بهینه‌سازی بیزی .

© ۱۴۰۴ (انجمن سیستم‌های فازی)

می‌شود. روش بهینه‌سازی بیزی با استفاده از تئوری احتمالات برای ابرپارامترها توزیع احتمالی تعیین می‌کند. به منظور ارزیابی عملکرد روش پیشنهادی از مجموعه داده‌های تصادفی استفاده شده است. روش پیشنهادی قابل اجرا و تعمیم به مجموعه داده‌های واقعی مثل داده‌های آب و هوا، محیط زیست و انرژی می‌باشد. نتایج به دست آمده کارایی و عملکرد مناسب روش پیشنهادی را نشان می‌دهد

۱. مقدمه

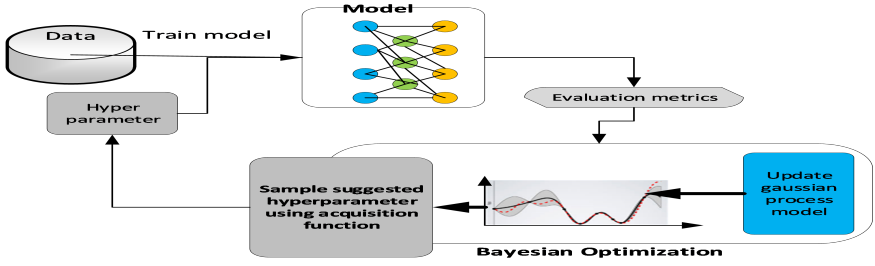
در یادگیری ماشین، مسائل بهینه‌سازی با توابع جعبه‌سیاه معمولاً چالش‌برانگیز هستند. در این موارد، تابع هدف ($f(x)$) به صورت تحلیلی معلوم نیست و مشتقات آن نیز قابل محاسبه نیست. برای ارزیابی تابع در نقاط مختلف، معمولاً از نمونه‌برداری استفاده می‌شود. اگر ارزیابی تابع ضعیف باشد، می‌توانیم از روش‌هایی مانند جستجوی شبکه، جستجوی تصادفی یا تخمین گرادیان عددی برای نمونه‌برداری استفاده کنیم. اما اگر ارزیابی قوی باشد، مثلاً در تنظیم ابرپارامترهای[1] شبکه‌های عصبی عمیق، حفاری کاوشگر در مختصات جغرافیایی مشخص یا ارزیابی اثربخشی داروهای کاندید، تعداد نمونه‌های استخراج شده از تابع جعبه‌سیاه (f) را به حداقل ممکن برسانیم.

شبکه‌های عصبی بازگشتی (RNN[2]) دسته‌ای از شبکه‌های عصبی مصنوعی هستند که دارای حالت‌های پنهان بوده و در زمینه‌های پردازش زبان طبیعی و تشخیص گفتار محبوب می‌باشند. شبکه LSTM[3] نوع خاصی از RNN است که قادر به یادگیری وابستگی‌های طولانی مدت می‌باشد. این شبکه‌ها دارای طیف گسترده‌ای از ابرپارامترها مانند نرخ یادگیری، تعداد واحدهای پنهان، طول ورودی، اندازه دسته‌ای و غیره هستند که انتخاب و بهینه‌سازی ابرپارامترها اغلب تفاوت بین عملکرد متوسط و پیشرفته را ایجاد می‌کند. یک ابرپارامتر یک متغیر پیکربندی در مدل می‌باشد که مقدار آن را می‌توان از داده‌های داده شده تخمین زد. ابرپارامترها هنگام پیش‌بینی توسط مدل مورد نیاز هستند. مقداردهی ابرپارامترها نقش مهمی در عملکرد مدل دارد، چرا که مهارت مدل را در حل مسئله تعیین می‌کند. این مقادیر معمولاً از طریق داده‌های آموزشی تخمین زده یا در جریان آموزش مدل یاد گرفته می‌شوند. برخلاف گذشته که تنظیم آن‌ها به‌صورت دستی و توسط متخصص انجام می‌شد، امروزه بیشتر این تنظیمات به‌عنوان بخشی از فرایند یادگیری مدل انجام می‌گیرد و همراه با مدل ذخیره می‌شوند. به همین دلیل ابرپارامترها یکی از اجزای کلیدی الگوریتم‌های یادگیری ماشین به شمار می‌روند. برای تنظیم ابرپارامترها الگوریتم‌های جستجوی متعددی وجود دارند. یکی از

[1] Hyperparameters
[2] Recurrent Neural Networks
[3] Long Short-Term Memory

روش‌های رایج، روش جستجوی شبکه‌ای است که به‌عنوان یک روش جامع و سیستماتیک برای یافتن مقادیر بهینه شناخته می‌شود. در این روش فضای جستجو شامل مجموعه‌ای از مقادیر ممکن برای هر پارامتر تعریف می‌شود. سپس مدل برای تمام ترکیب‌های ممکن از این مقادیر آموزش داده و ارزیابی می‌شود تا بهترین عملکرد و نتایج حاصل شود. اگرچه این روش از نظر دقت قابل توجه است، اما با افزایش تعداد پارامترها یا گسترده شدن دامنه‌ی آن‌ها، تعداد آزمایش‌های موردنیاز به‌صورت نمایی افزایش یافته و در نتیجه هزینه محاسباتی آن بسیار بالا می‌رود. یکی از روش‌های رایج به منظور کاهش هزینه محاسباتی در جستجوی شبکه استفاده از جستجوی تصادفی است. در این روش به‌جای بررسی همه‌ی ترکیب‌های ممکن، مجموعه‌ای از ترکیب‌های ابرپارامترها به‌صورت تصادفی از میان دامنه‌ی تعریف‌شده انتخاب می‌شود. به عبارت دیگر هدف این است که با نمونه‌برداری تصادفی بتوان یک ترکیب بهینه یا نزدیک به بهینه را به دست آورد. جستجوی تصادفی با کاهش تعداد آزمون‌ها از مسئله‌ی انفجار ابعاد جلوگیری می‌کند و از این جهت کارآمدتر از جستجوی شبکه‌ای می باشد. اما محدودیت اصلی این روش آن است که انتخاب ابرپارامترها در هر تکرار بدون در نظر گرفتن نتایج حاصل از تکرارهای قبلی کاملاً مستقل و به صورت تصادفی انجام می‌شود. بنابراین تضمینی وجود ندارد که این روش ما را به سمت بهترین ترکیب ممکن هدایت کند. بهینه‌سازی بیزی یک روش آماری پیشرفته برای انتخاب مؤثر ابرپارامترهاست که تلاش می‌کند در هر مرحله بر اساس اطلاعات به‌دست‌آمده از مراحل قبلی تصمیمات بهتری بگیرد. در این روش انتخاب‌های گذشته و عملکرد حاصل از آن‌ها بررسی می‌شود تا بتوان پیش‌بینی کرد کدام ترکیب از ابرپارامترها احتمال موفقیت بیشتری دارد. بر این اساس مجموعه‌های جدیدی از ابرپارامترها انتخاب و در تکرارهای بعدی مورد آزمایش قرار می‌گیرند. این فرآیند با ایجاد یک مدل آماری از عملکرد ابرپارامترها آغاز شده و سپس نمونه‌های منتخب با استفاده از تابع هدف واقعی ارزیابی می‌شوند. پس از توضیح اهمیت ابرپارامترها در بهبود عملکرد مدل، دیدگاهی در مورد نحوه‌ی برخورد با آن‌ها و شواهدی برای تقریب بهینه ارائه می‌شود.

جستجوی بیزی روشی است که با استفاده از بهینه‌سازی بیزی تلاش می‌کند عملکرد بهتری نسبت به جستجوی شبکه‌ای و جستجوی تصادفی در انتخاب ابرپارامترها داشته باشد. امروزه استفاده از آمار بیزی در بسیاری از الگوریتم‌های یادگیری ماشین رایج است و جستجوی بیزی نیز یکی از کاربردهای مهم این روش به شمار می‌رود. در این روش هدف انتخاب آن دسته از ابرپارامترهایی است که بیشترین احتمال درست‌نمایی (MLE) یا بیشترین تخمین پسین (MAP) را فراهم می‌کنند. در شکل ۱ نحوه عملکرد بهینه‌ساز بیزی در فرآیند بهینه‌سازی ابرپارامترهای یک مدل یادگیری ماشین نشان داده شده است.



شکل ۱: نمایشی از بهینه‌سازی بیزی جهت بهینه‌سازی ابرپارامترها

همان‌طور که گفته شد بهینه‌سازی ابرپارامترها در یادگیری ماشین فرآیندی است که در آن بهترین و مناسب‌ترین مقادیر برای پارامترهای یک مدل انتخاب می‌شود تا عملکرد مدل بهینه و بیشینه شود. منطق فازی با قابلیت‌های خود در مدل‌سازی عدم قطعیت و استدلال تقریبی، می‌تواند به عنوان یک ابزار مؤثر و قدرتمند در این زمینه مورد استفاده قرار گیرد. کاربرد منطق فازی در بهینه‌سازی ابرپارامترها می تواند شامل موارد زیر باشد:

- **مدل‌سازی عدم قطعیت:** منطق فازی می تواند با استفاده از متغیرهای زبانی و قوانین فازی، عدم قطعیت موجود در فرآیند انتخاب بهترین ابرپارامترها را به‌خوبی مدل‌سازی کند.
- **استفاده از دانش خبره:** با استفاده از منطق فازی می‌توان دانش خبره در مورد انتخاب ابرپارامترها را به صورت قوانین فازی بیان کرد و از این دانش برای هدایت فرآیند بهینه‌سازی استفاده کرد.
- **جستجوی هوشمندانه:** منطق فازی می‌تواند به عنوان یک استراتژی جستجو در فضای ابرپارامترها به کار رود و به طور هوشمندانه به جستجوی مجموعه‌های بهینه از ابرپارامترها بپردازد.
- **انعطاف‌پذیری در تنظیم پارامترها:** منطق فازی به دلیل توانایی خود در کار با مقادیر تقریبی و مبهم، انعطاف‌پذیری بیشتری در تنظیم ابرپارامترها نسبت به روش‌های سنتی ارائه می‌دهد.

در حوزه یادگیری ماشین، بهینه‌سازی بیزی یک روش قدرتمند است که در مسائل بهینه‌سازی با توابع جعبه‌سیاه کاربرد دارد. در این روش، توزیع پیشین برای تابع هدف $(f(x))$ تعیین می‌شود و با استفاده از نمونه‌برداری، توزیع پسین به‌روزرسانی می‌شود. این توزیع پسینی بهترین تقریب را از

تابع $(f(x))$ارائه می‌دهد. مدلی که برای تقریب تابع هدف استفاده می‌شود، مدل جایگزین نامیده می‌شود. بهینه‌سازی بیزی همچنین از یک تابع اکتساب استفاده می‌کند که نمونه‌گیری را به مناطقی هدایت می‌کند که در آن بهبودی نسبت به بهترین مشاهده فعلی محتمل است.

بهینه‌سازی بیزی یک چارچوب قدرتمند برای حل مسائل بهینه‌سازی است که امکان رسیدن به کمینه‌ی سراسری را با کمترین تعداد گام ممکن فراهم می‌کند. در این روش مسائل بهینه‌سازی از طریق روشی به نام «بهینه‌سازی جایگزین» حل می‌شوند. در این روش یک تابع جایگزین بر پایه‌ی نقاط نمونه‌برداری‌شده از تابع اصلی ساخته می‌شود. این تابع به ما کمک می‌کند تا تشخیص دهیم کدام نواحی احتمال بیشتری برای داشتن مقدار کمینه دارند. در ادامه از این نواحی امیدوارکننده نمونه‌گیری بیشتری انجام شده و تابع جایگزین بر اساس اطلاعات جدید به‌روزرسانی می‌شود. اساس آمار و مدل‌های بیزی بر این اصل استوار است که دانش اولیه با استفاده از داده‌های جدید به‌روزرسانی می‌شود تا دانشی تازه و دقیق‌تر حاصل شود. توابع جایگزین معمولاً با استفاده از فرآیندهای گاوسی مدل‌سازی می‌شوند. این فرآیندها را می‌توان به انداختن تاس تشبیه کرد، با این تفاوت که به جای عددی بین ۱ تا ۶، خروجی آن‌ها مجموعه‌ای از توابع مختلف مانند توابع سینوسی، لگاریتمی و غیره است که با داده‌های موجود هم‌خوانی دارند. نتیجه‌ی این فرآیند چندین تابع با احتمال‌های مختلف است. تابع جایگزین که نقش توزیع احتمال پیشین را دارد، به کمک یک تابع اکتسابی به‌روزرسانی می‌شود. این تابع اکتسابی از طریق فرآیندی متوازن بین اکتشاف و بهره‌برداری مجموعه‌ای از نقاط جدید را بررسی می‌کند. در مرحله‌ی بهره‌برداری، نمونه‌گیری از نواحی‌ای صورت می‌گیرد که مدل جایگزین در آن‌ها هدف مطلوب‌تری را پیش‌بینی می‌کند. در این مرحله از نقاط امیدوارکننده‌ی شناخته‌شده (نقاطی که احتمال وجود کمینه در آن‌ها بیشتر است) استفاده می‌شود. با این حال، اگر این نواحی و اطلاعات موجود به اندازه کافی بررسی شده باشند، ادامه‌ی جستجو در آن‌ها سود چندانی نخواهد داشت. هدف از اکتشاف نمونه‌گیری از ناحیه‌هایی است که ابهام بالایی دارند. به این ترتیب، اطمینان حاصل می‌شود که هیچ ناحیه‌ی وسیعی بدون جستجو باقی نمانده، چرا که ممکن است کمینه‌ی محلی دقیقاً در همین نواحی قرار داشته باشد. اگر تابع اکتسابی بیش از حد روی بهره‌برداری تمرکز کند و اکتشاف را نادیده بگیرد، مدل ممکن است پس از یافتن یک کمینه‌ی اولیه (که اغلب کمینه‌ی محلی است) در همان ناحیه متوقف و از ادامه‌ی جستجو صرف‌نظر شود. از سوی دیگر اگر تأکید بیش از اندازه بر اکتشاف باشد، مدل ممکن است هیچ‌گاه روی یک کمینه (چه محلی و چه سراسری) متمرکز نشود. بنابراین دستیابی به نتایج مطلوب زمانی امکان‌پذیر است که میان اکتشاف و بهره‌برداری مصالحه باشد. به‌طور خلاصه مراحل بهینه‌سازی بیزی عبارتست از:

- تعریف توزیع پیشین برای تابع جایگزین (تابع مبتنی بر فرآیند گاوسی)

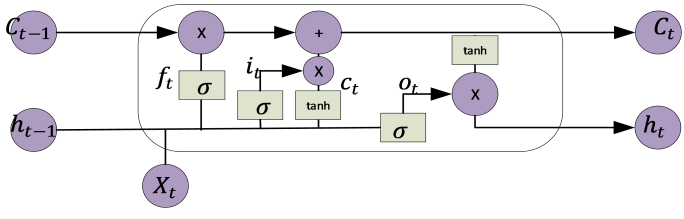
- انتخاب چند نقطه‌ی داده به‌گونه‌ای که تابع اکتسابی هنگام اجرا روی توزیع پیشین به مقدار بیشینه برسد.
- ارزیابی این نقاط در تابع هدف (تابع هزینه) و به‌دست آوردن نتایج
- به‌روزرسانی توزیع پیشین فرآیند گاوسی با داده‌های جدید برای تولید توزیع پسین
- تکرار مراحل بالا تا اتمام شرط توقف
- تفسیر توزیع نهایی فرآیند گاوسی به منظور یافتن کمینه‌ی سراسری

مدل‌های سری زمانی به منظور پیش‌بینی مقادیر آینده بر اساس داده‌های گذشته طراحی می‌شوند. این مدل‌ها ورودی‌هایی که به صورت متوالی در طول زمان ثبت شده‌اند را مورد بررسی قرار می‌دهند تا پیش‌بینی رویدادهای آینده را انجام دهند. با این حال، مدل‌هایی مانند مدل میانگین متحرک یکپارچه اتورگرسیون (ARIMA[1]) محدودیت‌های خاص خود را دارند و برای داده‌هایی که ویژگی‌های آماری آن‌ها در طول زمان تغییر می‌کند مناسب نیستند. به عبارت دیگر در صورتی که میانگین و انحراف استاندارد داده‌ها در طول زمان ثابت نباشد (مانند قیمت سهام غیر ثابت در طول زمان)، مدل‌های مبتنی بر ARIMA نمی‌توانند رویدادهای آینده را به طور صحیح پیش‌بینی کنند. شبکه‌های عصبی LSTM یکی از انواع شبکه‌های عصبی بازگشتی (RNN) می‌باشند که برای مدل‌سازی داده‌های ترتیبی یا دنباله دار در یادگیری عمیق مورد استفاده قرار می‌گیرند. این شبکه‌ها برای اولین بار توسط هوچریترو و اسمیدوبر در سال ۱۹۹۷ معرفی شدند [۱۳]. شبکه‌های عصبی LSTM یا به طور کلی RNNها دارای حلقه‌های بازخوردی هستند که آنها را قادر می‌سازد اطلاعات را در یک دوره زمانی ذخیره کنند یا به عبارت دیگر اطلاعات قبلی را در محاسبات بعدی استفاده کنند. به حلقه‌های بازخوردی ظرفیت حافظه گفته می‌شود. شبکه‌های عصبی معمولی از حافظه ترتیبی برخوردار نیستند و معمولا در کاربردهایی که داده‌های ترتیبی ندارند (مانند تصاویر) مورد استفاده قرار می‌گیرند.

معماری شبکه‌های LSTM شامل سلول‌های حافظه است که توسط سه دروازه کنترل می‌شود: دروازه ورودی، دروازه فراموشی و دروازه خروجی. این دروازه‌ها تصمیم می‌گیرند که چه اطلاعاتی به سلول حافظه اضافه یا حذف شود و کدام اطلاعات سلول حافظه به عنوان خروجی ارسال شوند. به عبارت دیگر، دروازه ورودی تعیین می‌کند که چه اطلاعاتی به سلول حافظه اضافه شود. دروازه فراموشی تعیین می‌کند که چه اطلاعاتی از سلول حافظه حذف شود. همچنین، دروازه خروجی تعیین می‌کند که چه اطلاعاتی از سلول حافظه به عنوان خروجی ارسال شود. این دروازه‌ها امکان شناسایی وابستگی‌های طولانی‌مدت در داده‌ها و پیش‌بینی‌های طولانی‌مدت را به شبکه LSTM می‌دهد. علاوه بر لایه‌های قابل مشاهده، یک حالت پنهان (Hidden State) در شبکه‌های LSTM

[1] Autoregressive Integrated Moving Average

وجود دارد. حالت پنهان موجب افزایش توانایی شبکه LSTM در پیش‌بینی سری‌های زمانی می‌شود. این حالت پنهان اطلاعات قبلی را در خود ذخیره می‌کند و به عنوان حافظه کوتاه‌مدت شبکه عمل می‌کند. این اطلاعات پنهان به مرور زمان به شکل بازگشتی به لایه‌های اصلی بازگردانده می‌شوند. حالت پنهان بر اساس ورودی، حالت پنهان قبلی و وضعیت فعلی سلول حافظه به‌روز می‌شود. شکل ۲ ساختار یک سلول حافظه نمونه در شبکه LSTM را نشان می‌دهد. در شکل، h_{t-1} حالت پنهان در زمان $t-1$، c_{t-1} حالت سلول حافظه در زمان $t-1$، X_t ورودی در زمان t، i_t دروازه‌ی ورودی، Q_t دروازه‌ی خروجی و f_t دروازه‌ی فراموشی می‌باشند. توابع سیگموئید (σ) و تانژانت هایپربولیک ($\tanh$) نیز توابع فعال‌سازی می‌باشند. محاسبات و فرمول‌های ریاضی این شکل به‌طور کامل‌تر در [۸] آمده است.



شکل ۲: ساختار یک سلول حافظه LSTM

الگوریتم‌های یادگیری ماشین معمولا دارای پارامترهایی می‌باشند. تنظیم و انتخاب این پارامترهای سطح بالا عملکرد مدل را به طور قابل توجهی تحت تاثیر قرار می‌دهد. به عبارت دیگر پارامترهای سطح بالا تنظیماتی هستند که وظیفه کنترل یادگیری مدل را بر عهده دارند. نرخ یادگیری یا تعداد نرون‌ها در شبکه‌های عصبی نمونه‌هایی از این پارامترها هستند. توسعه الگوریتم‌هایی با کمترین تعداد ممکن از این پارامترها یک هدف مطلوب در یادگیری ماشین می‌باشد. تنظیم ابرپارامترها[1] فرآیند جستجوی دقیق‌ترین ابرپارامترها برای یک مجموعه داده با الگوریتم یادگیری ماشین است. یک روش برای انجام این کار تغییر یک به یک ابرپارامترها به طور متناوب و ارزیابی مدل با هر یک از این تغییرات تا زمانی که بهترین دقت حاصل شود.

یک راه دیگر برای تنظیم ابرپارامترها استفاده از مدل های کاملا بیزی می‌باشد. مدل‌های کاملا بیزی با استفاده از پارامترهای هسته‌ای (GP[2]) نتایج خوبی را ارائه می‌دهند. در این تحقیق از

[1] Hyperparameter-tuning
[2] Gaussian Process

بهینه‌سازی بیزی برای تنظیم و بررسی ابرپارامترهای شبکه عصبی LSTM استفاده می‌شود. تاثیر خود هسته نیز مورد بررسی قرار می‌گیرد. به جای هسته پیش فرض تابع کوواریانس مربع نمایی از یک هسته $Matern\ 5/2$ استفاده می‌شود. به منظور انتخاب توابع اکتساب، از تابع بهبود مورد انتظار برای ایجاد مصالحه و تعادل بین اکتشاف و بهره‌برداری استفاده می‌شود. همچنین، موضوع مهم زمان و هزینه محاسباتی در رویکرد این مقاله در نظر گرفته می‌شود. بعد از درک کامل روش بهینه‌سازی بیزی، از این روش برای تنظیم ابرپارامترها در پیش‌بینی سری‌های زمانی به کمک LSTM استفاده می‌شود. موارد ذکر شده در ادامه مقاله به صورت دقیق‌تری توضیح داده می‌شوند.

۲. مروری بر کارهای انجام شده

در این مقاله دو موضوع مورد توجه قرار گرفته است: پیش‌بینی سری‌های زمانی با شبکه عصبی LSTM و بهینه‌سازی پارامترهای شبکه عصبی که یک چالش مهم در مورد شبکه‌های عصبی می‌باشد. یک روش بسیار متداول برای پیش‌بینی سری‌های زمانی استفاده از شبکه‌های عصبی LSTM می‌باشد [۱، ۲، ۷، ۱۵]. به عنوان مثال در [۷] با استفاده از دو ماژول LSTM که یکی از ماژول‌ها برای پیش‌بینی بود، پیش‌بینی شاخص بورس اوراق بهادار انجام شد. در یک تحقیق دیگر از LSTM و یادگیری انتقال‌دهنده در پیش‌بینی سری‌های زمانی مربوط به دما استفاده شد [۱۵]. در [۱] با استفاده از دو شبکه عصبی بازگشتی (RNN) و حافظه طولانی کوتاه مدت (LSTM) یک مدل ترکیبی به منظور پیش‌بینی سری‌های زمانی مربوط به بازار فارکس ارائه شد. در [۲] به منظور پیش‌بینی سری زمانی یونسفر در دوره فعالیت‌های شدید خورشیدی یک روش جدید به کمک LSTM پیشنهاد شد. همچنین در یک تحقیق دیگر از LSTM در پیش‌بینی مسائل مربوط به زمان‌بندی پروژه‌های عمرانی که یکی از مسائل مهم و مطرح در این زمینه می‌باشد استفاده شده است [۳].

یک چالش مهم برای شبکه‌های LSTM انتخاب پارامترهای مناسب می‌باشد. اغلب این پارامترها به صورت تجربی و تصادفی و با سعی و خطا به دست می‌آیند. مثلا در [۱۴] مقدار مناسب ابرپارامترها بر مبنای دقت پیش‌بینی داده‌های مربوط به اعتبارسنجی به دست آمده است. دقت بالا برای پیش‌بینی مقدار نرخ تورم استان کرمانشاه متناسب با کمینه‌شدن معیار RMSE می‌باشد. در [۱۹] نیز به منظور پیش‌بینی انتقال بیماری کوید -۱۹ از شبکه‌های LSTM استفاده شده است و به منظور به دست آوردن بهترین ابرپارامترها چندین مدل به رقابت می‌پردازند. با مقایسه نتایج این مدل‌ها بهترین ابرپارامترهای شبکه LSTM انتخاب می‌شوند. در [۱۷] انتخاب ابرپارامترهای مربوط به شبکه عصبی LSTM به منظور پیش‌بینی بازار سهام دوره‌ای نیز به صورت

تجربی انجام می‌شود. در بعضی تحقیقات از روش‌های ابتکاری برای بهینه‌سازی ساختار شبکه‌های LSTM و انتخاب بهینه ابرپارامترها استفاده شده است [۹، ۱۲، ۱۸]. مثلا در [۹] بهینه‌سازی کلونی مورچه‌ها برای تقویت معماری شبکه LSTM به کار رفته است. در [۱۸] برای بهینه‌سازی ساختار LSTM از الگوریتم ژنتیک به منظور سری‌های زمانی تک متغیره استفاده شده است. همچنین در [۱۲] به کمک الگوریتم ژنتیک یک مدل LSTM عمیق مبتنی بر توجه برای پیش‌بینی سری‌های زمانی پیشنهاد شده است. سه مجموعه داده در زمینه‌های محیط زیست و انرژی به منظور ارزیابی این مدل مورد استفاده قرار گرفته‌اند. در [۲۰] به منظور پیش‌بینی سری‌های زمانی شبکه‌های عصبی و روش‌های ابتکاری به صورت ترکیبی به کار رفته‌اند. از مباحث دیگری که در زمینه پیش‌بینی سری‌های زمانی مطرح شده است خوشه‌بندی سری‌های زمانی نادقیق با روش‌های ترکیبی است [۱۶]. در [۴] مرور و مقایسه‌ای بر روی پیش‌بینی نرخ ارز با استفاده از شبکه‌های عصبی عمیق انجام شده است. در [۵، ۶] یک بهبود برای افزایش دقت پیش‌بینی سری‌های زمانی صورت گرفته است. در این مقاله برای تعیین ابرپارامترهای بهینه از روش بهینه‌سازی بیزی استفاده می‌شود. از مزایای روش بهینه‌سازی بیزی کارایی بالا، انعطاف پذیری آن در مواجه‌شدن با توابع هدف غیرخطی و حتی پیچیده، قابلیت تنظیم و تطبیق با انواع مسائل و مدیریت عدم قطعیت داده‌ها و مدل می‌باشد. همچنین در روش بیزی به منظور افزایش سرعت همگرایی در فرآیند بهینه‌سازی از اطلاعات پیشین استفاده می‌شود. روش بهینه‌سازی بیزی در ادامه مقاله توضیح داده می‌شود.

۳. مفاهیم اولیه

۱.۳. بهینه‌سازی بیزی با فرآیند گاوسی[1]. بهینه‌سازی بیزی یک روش رایج برای تنظیم ابرپارامترها و بهینه‌سازی توابع پیچیده در کاربردهای یادگیری ماشین از جمله شبکه‌های عصبی LSTM می‌باشد. اساس کار بهینه‌سازی بیزی بر مبنای تئوری احتمالاتی بیزی است و ابر پارامترهای بهینه را به منظور بهینه‌سازی تابع هدف تعیین می‌کند.

ایده اصلی پشت بهینه‌سازی بیزی این است که از یک مدل جایگزین[2] برای تقریب یک تابع یا فرآیند هدف واقعی استفاده و سپس از این تقریب برای تعیین آزمایش بعدی استفاده شود. به طور معمول، مدل‌های گاوسی[3]، درخت تصمیم[4] یا سایر مدل‌های احتمالی مشابه به عنوان مدل‌های جایگزین استفاده می‌شوند. سپس مدل جایگزین با داده‌های جدید به روز رسانی می‌شود. با این

[1] Bayesian optimization with gaussian process
[2] Surrogate Model
[3] Gaussian Process
[4] Decision Tree

کار تقریب بهتری از تابع هدف جهت تنظیم ابرپارامترها به دست می‌آید. به عبارت دیگر، مدل جایگزین با چند نقطه مقداردهی اولیه می‌شود و سپس از یک تابع اکتساب[1] برای تعیین نقطه بعدی برای ارزیابی استفاده می‌شود. اکتساب بهترین جواب بهینه را از اطلاعات و ارزیابی‌های قبلی پیدا می‌کند. همچنین تابع اکتساب، اکتشاف[2] را متعادل می‌کند. اکتشاف به جستجوی مناطق ناشناخته در فضای مساله می‌پردازد و با این کار ممکن است مقادیر بهینه جدید پیدا شود. برای اکتشاف به طور تصادفی از فضای مساله نمونه‌برداری می‌شود. به منظور حداکثر کارایی در بهینه‌سازی بیزی باید همیشه بین اکتشاف و اکتساب یک مصالحه برقرار باشد.

بهینه‌سازی بیزی بر مبنای استفاده از معادله ساده مربوط به قضیه معرف بیز است معادله (۱.۳). در این معادله M معرف مدل (نظریه یا فرضیه) مساله، E معرف مشاهدات یا داده‌ها و p نشان دهنده احتمال شرطی یا غیر شرطی می‌باشد.

$$P(M \mid E) \propto P(E \mid M)P(M) \tag{۱.۳}$$

معادله (۱.۳) با توجه به تابع هزینه مقادیر بهینه ابرپارامترها را تعیین می‌کند. اگرچه تابع هزینه یا هدف ناشناخته است، اما توزیع پیشین بیانگر باور ما در مورد فضای توابع هدف ممکن و محتمل است. فلسفه اساسی این است که از تمام اطلاعات موجود از ارزیابی‌های قبلی تابع هدف استفاده شود و صرفاً بر شیب محلی و تقریب‌های هسین تکیه نشود. با این کار حداقل توابع غیرمحدب دشوار با ارزیابی‌های نسبتاً کمی پیدا می‌شود. به عبارت دیگر با در نظر گرفتن داده‌های آموزشی توزیع احتمال پارامترها به‌روزرسانی و پارامترهای بهینه حاصل می‌شود.

اجزای اصلی بهینه‌سازی بیزی عبارتند از:

(۱) تابع هدف: تابع یا فرآیندی است که نیاز به بهینه‌سازی دارد، اما ارزیابی آن به منظور بهینه‌سازی پرهزینه یا زمان بر است. تابع هدف معمولاً یک جعبه سیاه است، به این معنی که شکل ریاضی آن ناشناخته است و فقط ورودی‌ها و خروجی‌های آن قابل مشاهده است.

(۲) مدل جانشین یا جایگزین: یک مدل رگرسیونی است که برای تقریب تابع هدف استفاده می‌شود. متداول‌ترین مدل جایگزین مورد استفاده در بهینه‌سازی بیزی، یک فرآیند گاوسی است که یک مدل غیرپارامتری انعطاف‌پذیر بوده و می‌تواند روابط پیچیده و غیرخطی بین ورودی‌ها و خروجی‌های تابع هدف را ثبت کند.

---

[1] Exploitation
[2] Exploration

(۳) تابع اکتساب: برای تعیین نقطه بعدی برای ارزیابی در فضای جستجوی مساله استفاده می‌شود. تابع اکتساب، اکتشاف و بهره‌برداری را متعادل می‌کند. مصالحه بین اکتساب و اکتشاف باید وجود داشته باشد.

(۴) داده‌های آموزشی: بهینه‌سازی بیزی به برخی داده‌های اولیه برای ساخت مدل جایگزین نیاز دارد. این داده‌ها را می‌توان با ارزیابی تابع هدف در چند نقطه از فضای جستجو به دست آورد. با توجه به اینکه بودجه کل آزمایش اغلب محدود است، معمولاً تصمیم‌گیری در مورد این نقاط آموزشی اولیه به توجه و دقت زیادی نیاز دارد.

(۵) معیار توقف: در بهینه‌سازی بیزی یک معیار توقف برای تعیین زمان توقف جستجو مورد نیاز است. این معیار می‌تواند رسیدن به همگرایی جوابها، یا تعداد آزمایش‌های معین و یا محدودیت‌ها باشد.

۲.۳. فرآیند گاوسی. فرآیند گاوسی (GP) یک مدل احتمالی است که توزیع بر روی توابع را تعریف می‌کند. در یک مدل GP فرض بر این است که یک تابع می‌تواند به عنوان مجموعه‌ای از متغیرهای تصادفی با توزیع گاوسی چند متغیره نمایش داده شود. GP قادر به ایجاد توزیع بر روی توابعی است که با تابع میانگین m و تابع کوواریانس k مشخص شده‌اند (معادله (۲.۳)). در معادله (۲.۳) تابع f تابع هدف می‌باشد.

$$f(x) \sim GP(m(x), K(x, x')) \tag{۲.۳}$$

به طور شهودی، در GP فرض بر این است که نقاط داده با همبستگی بالا بین متغیرهای کمکی، مقادیر مشابهی از متغیرهای خروجی دارند. همچنین GP گسترشی از توزیع گاوسی چند متغیره به یک فرآیند تصادفی با ابعاد نامتناهی است که شامل هر ترکیب محدودی از ابعاد یک توزیع گاوسی خواهد بود. یک مدل GP با یک تابع میانگین و یک تابع کوواریانس که تابع هسته نام دارد، تعریف می‌شود (معادله (۳.۳)).

$$p(f \mid X) = \mathcal{N}(f \mid \mu, \Sigma) \tag{۳.۳}$$

در معادله (۳.۳)، f تابع هدف و X مجموعه‌ای از مشاهدات برای متغیرهای کمکی می‌باشد. انتخاب تابع هسته مناسب بر اساس ویژگی‌های تابع هدف و داده‌های مساله مورد نظر می‌باشد. هسته‌ها انواع مختلفی دارند و اطلاعات مربوط به داده‌ها را به فضایی با ابعاد بالاتر انتقال می‌دهند. به طور شهودی هسته تابعی است که شباهت بین جفت‌های بردار متغیرهای کمکی را مشخص می‌کند. به عبارت دیگر، هسته باید با توجه به ورودی، میزان شباهت دو نقطه داده را تعیین کند.

هسته‌ها ابزاری را برای تعریف توزیع پیشین بر روی تابع هدف یا فرآیند بهینه‌سازی شده فراهم می‌کنند. هسته‌های زیادی از جمله هسته خطی، چندجمله‌ای، گاوسی، سیگموئید و غیره به منظور استفاده وجود دارد. برای اجرای موفقیت‌آمیز بهینه‌سازی بیزی، انتخاب یک هسته مناسب ضروری و در عین حال دشوار است. هسته‌های گاوسی در مورد داده‌هایی که جداپذیر با سطح‌های غیرخطی هستند، مناسب است. در بهینه‌سازی بیزی به طور گسترده از فرآیندهای گاوسی استفاده می‌شود.

هسته درجه دوم گویا (RQ[1]) تعمیم‌یافته هسته RBF[2] یا گاوسی است. به این معنا که می‌توان آن را به عنوان مجموع نامتناهی از هسته‌های RBF با مقیاس‌های طولی مختلف تفسیر کرد. هسته RQ به‌صورت معادله (۴.۳) تعریف می‌شود:

$$k(x,x') = \sigma^2 \left(1 + \frac{\|x-x'\|^2}{2\alpha\ell^2}\right)^{-\alpha} \qquad (۴.۳)$$

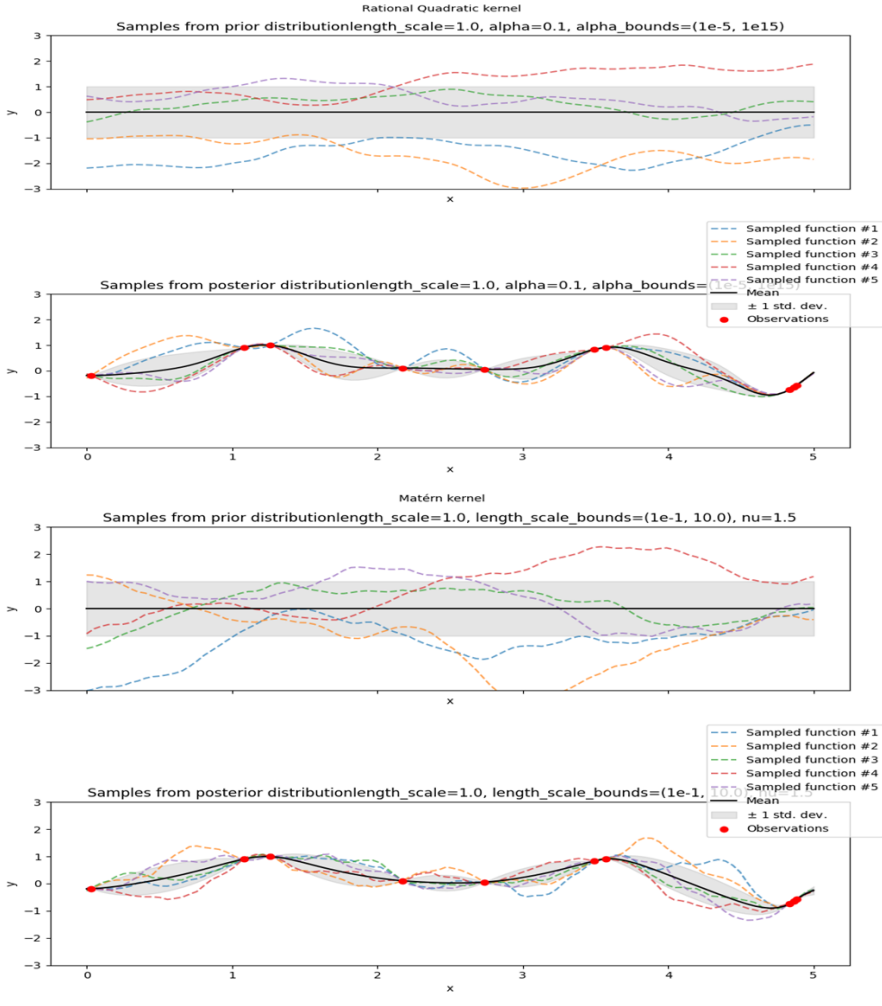
که در آن $\|x-x'\|^2$ مجذور فاصله اقلیدسی بین دو بردار است. σ^2 واریانس است و برای مقیاس هسته به کار می‌رود. l مقیاس طول می‌باشد و میزان سرعت کاهش شباهت بین دو نقطه ورودی با میزان افزایش فاصله آن‌ها را نشان می‌دهد. هسته RQ نسبت به هسته RBF بدون پیچیدگی بیش از حد، انعطاف‌پذیری بیشتری دارد و بنابراین در فرآیندهای گاوسی کارایی بهتری دارد. یک هسته ثابت، انعطاف‌پذیر و همه‌کاره دیگر، هسته ARD Matern[3] است که می‌تواند طیف گسترده‌ای از عملکردها را مدل کند و توسط معادله (۵.۳) بیان می‌شود:

$$k(x,x') = \sigma^2 \cdot \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}\|x-x'\|}{\ell}\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}\|x-x'\|}{\ell}\right) \qquad (۵.۳)$$

هسته ARD Matern مفروضات کمی در مورد تابع مدل‌سازی دارد. مقیاس طول و پارامترهای هموار این هسته، امکان مدل‌سازی توابع صاف با کوواریانس دوربرد و همچنین توابع با تغییرات ناگهانی را فراهم می‌کند؛ بنابراین هسته ARD Matern نیز یک انتخاب محبوب برای فرآیندهای گاوسی می‌باشد. نقطه‌ضعف هسته ARD Matern، انعطاف‌پذیری زیاد آن است و به‌منظور جلوگیری از برازش بیش از حد، نیاز به داده و تلاش دارد. استفاده از هسته‌های مختلف در ابعاد مختلف به‌منظور فراهم کردن نمایش دقیق‌تری از دانش قبلی فرآیند مدل‌سازی‌شده اغلب انجام می‌شود. به‌طور خاص، اگر ورودی دارای ویژگی‌ها یا واحدهای متفاوتی باشد، می‌توان با استفاده از هسته‌های متفاوت و مناسب برای هر کدام از انواع ویژگی‌ها، همه ویژگی‌ها را در یک فرآیند گاوسی گنجاند. در ادامه، به‌منظور به‌کارگیری مناسب‌ترین هسته جهت بهینه‌سازی ابرپارامترها با استفاده از شبکه

[1] Rational Quadratic
[2] Radial Bbasis Function
[3] Automatic Relevance Determination

عصبی LSTM، هسته‌های مختلف مورد بررسی قرار می‌گیرد. فرض شده است که مدل یک تابع سینوسی باشد و نقاط با استفاده از توزیع یکنواخت[1] در نرم‌افزار پایتون ایجاد شده است. نتایج بررسی هسته‌های مختلف جهت اجرای موفقیت‌آمیز بهینه‌سازی بیزی در شکل ۳ نشان داده شده است.



شکل ۳: نتایج مربوط به بررسی و مقایسه دو هسته RQ و ARD Matern جهت اجرای موفقیت آمیز بهینه‌سازی بیزی و عملکرد بهتر هسته ARD Matern

[1] Uniform

شکل ۳ مشاهده می‌شود که هسته ARD Matern نتایج و عملکرد بهتری را ارائه داده است.

۳.۳. توابع اکتساب[1]. توابع اکتسابی با داده‌های ما مطابقت دارند و با ارزیابی این توابع می‌توان در مورد اینکه آزمایش بعدی برای اجرا کدام باشد تصمیم‌گیری کرد. بین اکتشاف (جاییکه واریانس زیاد است) و بهره‌برداری (جایی که میانگین کم است) مصالحه برقرار و نقطه بعدی برای محاسبه تابع هدف انتخاب می‌شود (معادله (۶.۳)). در واقع تابع هدف با یک تابع شبه هدف جایگزین می‌شود.

$$x_{next} = \arg\max_{x} a(x) \tag{۶.۳}$$

یک چالش مهم در بهینه‌سازی به حداکثر رساندن تابع اکتساب است. تابع اکتساب با استفاده از رویکردهای استاندارد می‌تواند به حداکثر برسد. بسیاری از توابع اکتساب را می‌توان در چارچوب تئوری تصمیم بیزی به عنوان ارزیابی زیان مورد انتظار مرتبط با ارزیابی تابع $(f(x))$ در نقطه (x) تفسیر کرد. بنابراین نقطه‌ای با کمترین ضرر مورد انتظار انتخاب می‌شود.

۴.۳. احتمال بهبود[2]. فرض می‌شود:

$$f_{\min} = \min f \tag{۷.۳}$$

در معادله (۷.۳)، $f_{\min}$ حداقل مقدار f می‌باشد که تاکنون مشاهده شده است.
Probability of Improvement، احتمال بهبود تابع هدف f را در نقطه‌ای که به احتمال زیاد در این مقدار بهبود می‌یابد، ارزیابی می‌کند. ارزیابی تابع هدف f در نقطه x به کمک تابع زیر (معادله (۸.۳)) صورت می‌گیرد:

$$u(x) = \begin{cases} ۱ & \text{اگر } f(x) < f_{\min} \\ ۰ & \text{اگر } f(x) > f_{\min} \end{cases} \tag{۸.۳}$$

در معادله (۸.۳)، اگر $f(x)$ کمتر از $f_{\min}$ باشد، پاداش واحد دریافت می‌شود و در غیر این صورت هیچ پاداشی دریافت نمی‌شود. با استفاده از معادله (۸.۳)، تابع اکتساب Probability of Improvement به عنوان تابعی از x و به‌صورت معادله (۹.۳) تعریف می‌شود:

$$a_{\text{Pi}}(x) = \mathbb{E}[u(x) \mid x, D] = \int_{-\infty}^{f_{\min}} \mathcal{N}(f; \mu(x), K(x,x))\, df \tag{۹.۳}$$

[1] Acquisition Functions
[2] Probability of improvement

نقطه‌ای با بیشترین احتمال بهبود (حداکثر مطلوبیت مورد انتظار) به‌منظور اقدام بعدی الگوریتم بیزی انتخاب می‌شود.

۵.۳. بهبود مورد انتظار[1]. در تابع اکتساب قبلی، برای بهبود مینیمم تابع هدف در مرحله جاری، پاداش دریافتی مستقل از اندازه پیشرفت می‌باشد. این پاداش گاهی اوقات منجر به رفتارهای عجیب و غریب می‌شود و ممکن است باعث گیر افتادن الگوریتم در مینیمم‌های محلی شود و قابلیت تعمیم خود را از دست بدهد. یک تابع اکتساب جایگزین که اندازه پیشرفت را محاسبه می‌کند، Expected Improvement است.

مجدداً فرض می‌شود که $f_{\min}$ کمترین مقدار تابع $f(x)$ می‌باشد که تاکنون مشاهده شده است. Expected Improvement، تابع $f(x)$ را در نقطه‌ای که انتظار بیشترین بهبود را بر اساس $f_{\min}$ داشته باشد، به کمک معادله (۱۰.۳) ارزیابی می‌کند:

$$u(x) = \max(0, f_{\min} - f(x)) \tag{۱۰.۳}$$

در معادله (۱۰.۳)، اگر $f(x) < f_{\min}$ باشد، پاداشی برابر با improvement دریافت و در غیر این صورت هیچ پاداشی دریافت نمی‌کند. در ادامه، تابع اکتساب Expected Improvement، مطلوبیت مورد انتظار را به عنوان تابعی از x و با استفاده از معادله (۱۱.۳) محاسبه می‌کند:

$$a_{\mathrm{Ei}}(x) = \mathbb{E}[u(x) \mid x, D] = \int_{-\infty}^{f_{\min}} (f_{\min} - f(x))\, \mathcal{N}(f; \mu(x), K(x,x))\, \mathrm{df}$$

$$= (f_{\min} - \mu(x))\, \varphi(f_{\min}; \mu(x), K(x,x)) + K(x,x)\, \mathcal{N}(f_{\min}; \mu(x), K(x,x)) \tag{۱۱.۳}$$

نقطه‌ای با بالاترین بهبود مورد انتظار (حداکثر مطلوبیت مورد انتظار) به‌منظور اقدام بعدی الگوریتم بیزی انتخاب می‌شود. معادله (۱۱.۳) دارای دو جزء می‌باشد. برای افزایش مقدار این معادله، می‌توان تابع میانگین $\mu(x)$ در جزء اول را کاهش داد یا تابع واریانس $K(x,x)$ در جزء دوم را افزایش داد. این دو عبارت در معادله (۱۱.۳) به‌عنوان کدگذاری صریحی از مبادله بین بهره‌برداری (ارزیابی در نقاط با میانگین پایین) و اکتشاف (ارزیابی در نقاط با عدم قطعیت بالا) تفسیر می‌شوند. معاوضه‌ی بهره‌برداری–اکتشاف یک ملاحظه کلاسیک در حل چنین مسائلی می‌باشد. با این کار، معیار بهبود مورد انتظار به‌طور خودکار هر دو پارامتر بهره‌برداری و اکتشاف را در حل نظری تصمیم بیزی لحاظ می‌کند.

[1] Expected improvement

۶.۳. هزینه‌های مدل‌سازی[1]. در مسائل بهینه‌سازی، دو تابع هدف واقعی $f(x)$ و تابع مدت‌زمان $c(x) : X \to \mathbb{R}^+$ ناشناخته هستند. با این وجود، در حل این مسائل می‌توان از ماشین‌ها یا مدل‌های فرآیند گاوسی برای مدل‌سازی $\ln(c(x))$ در کنار $f(x)$ استفاده کرد. فرض می‌شود که این توابع مستقل از یکدیگر هستند. تحت فرض استقلال، می‌توان به‌راحتی مدت معکوس پیش‌بینی‌شده را محاسبه و از آن برای محاسبه بهبود مورد انتظار به‌عنوان تابعی از x استفاده کرد.

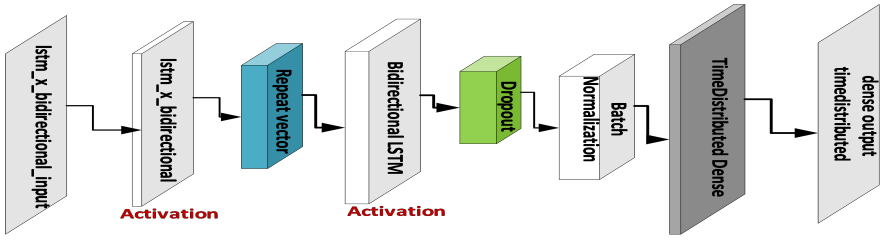
در ادامه مقاله اصول اولیه پیش‌بینی سری‌های زمانی مورد بررسی قرار می‌گیرد. چرا پیش‌بینی اینقدر مهم است؟ صرفاً به این دلیل که می‌توان آگاهانه تصمیم‌گیری کرد. هنگامی که یک زمان یا جزء زمانی به پیش‌بینی مرتبط می‌شود تبدیل به پیش‌بینی سری زمانی شده و داده‌ها به عنوان داده‌های سری زمانی نامیده می‌شوند. از نظر آماری پیش‌بینی سری‌های زمانی فرآیند تجزیه و تحلیل داده‌های سری زمانی با استفاده از آمار و سپس مدل‌سازی برای پیش‌بینی‌ها و تصمیم‌های استراتژیک آگاهانه است. برای انجام پیش‌بینی‌های دقیق، داده‌های سری زمانی در یک دوره جمع‌آوری شده، سپس با تجزیه و تحلیل آنها مدلی برای پیش‌بینی ساخته و ایجاد می‌شود. برای انجام این فرآیند قوانین خاصی وجود دارد و به کمک این قوانین نتایجی با دقت بالا حاصل می‌شود.

۴. روش انجام کار

در بخش‌های قبلی مقاله مطالبی راجع به پیش‌بینی سری‌های زمانی و الگوهای مرتبط با بهینه‌سازی بیزی بیان شد. همچنین به مرور و بررسی مطالعات متعدد مرتبط با پیش‌بینی سری‌های زمانی و کاربردهای آن پرداخته شد. در این بخش به پیاده‌سازی مدل پیشنهادی و استفاده از داده‌ها به منظور ارزیابی مدل پیشنهادی پرداخته می‌شود. به منظور بررسی عمیق‌تر داده‌های نویزی جهت بررسی هدف این مقاله مورد استفاده قرار گرفته است. برای حل مشکل داده‌های نویزی از تکنیک‌های آماده‌سازی داده‌های مرتبط با یادگیری عمیق استفاده شده است. دراین مقاله یک شبکه عصبی بازگشتی LSTM چند لایه به منظور پیش‌بینی سریهای زمانی به عنوان داده‌های ورودی پیاده‌سازی شده است (شکل ۴).

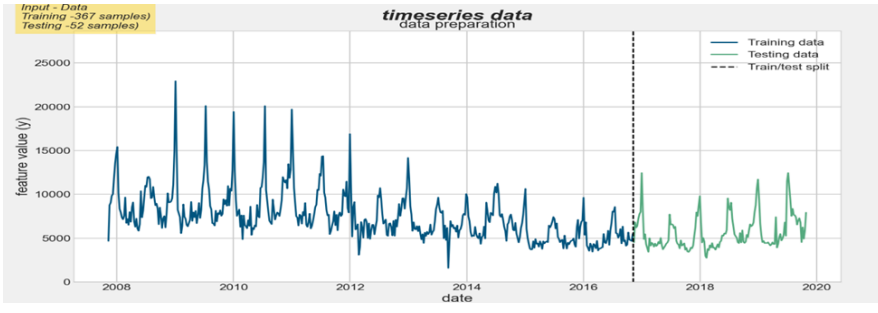
در ابتدا مدل با استفاده از سری زمانی ایجاد شده به کمک مجموعه داده‌های تصادفی تولید شده در شکل ۵ مورد ارزیابی قرار گرفته است. مجموعه داده‌ها دو بعدی می‌باشد. یک بعد مربوط به زمان است و بعد دیگر به صورت تصادفی ایجاد شده است. چون روش پیشنهادی در این مقاله یک روش عمومی و قابل تعمیم است می‌توان به جای اعداد تصادفی ایجاد شده هر داده واقعی از قبیل داده‌های آب و هوا، داده‌های انرژی، داده‌های محیط زیست و یا هر داده واقعی دیگری را قرار

[1] Modelling Costs



شکل ۴: مدل شبکه عصبی بازگشتی LSTM چند لایه به منظور پیاده‌سازی الگوریتم پیشنهادی جهت پیش‌بینی سریهای زمانی

داد. داده‌های تصادفی ایجاد شده قبل از اعمال به مدل با روش $\min - \max$ نرمال‌سازی شده‌اند. در ادامه مدل مجددا با سری زمانی تولید شده و استفاده از Hyperparameter-tuning آموزش داده می‌شود.



شکل ۵: نمایشی از مجموعه داده‌های تصادفی تولید شده جهت بررسی مدل پیشنهادی و تقسیم داده‌ها به داده‌های آموزشی و داده‌های ارزیابی

ساختار مدل مورد استفاده برای آموزش با مجموعه سری‌های زمانی در هر دو مورد یکسان است. تفاوت در مقادیر ابرپارامترهای انتخاب شده برای آموزش می‌باشد. پارامترها و متغیرهای مورد نیاز برای تخمین با استفاده از یک الگوریتم بهینه‌سازی اصلاح و در طول آموزش به‌روزرسانی می‌شوند. ابرپارامترها مستقیماً وبراساس داده‌ها برآورد نمی‌شوند. آنها خارج از مدل هستند و قبل از شروع مرحله یادگیری تنظیم می‌شوند. با این حال برخی از آنها را می‌توان در طول آموزش تنظیم کرد. تنظیمات ابرپارامترها تاثیر زیادی بر یادگیری مدل و چگونگی پیش‌بینی ها داده‌ها دارد. مدل‌های GP به منظور کارایی و عملکرد مناسب شبکه عصبی برای پیش‌بینی توزیع گاوسی علاوه بر

مقدار میانگین از یک تابع پیشبین واریانس نیز استفاده می‌کنند. این حسن موجب می‌شود مدل‌های GP به نتایجی بهتر از پیش‌بینی مورد انتظار از یک آزمایش نمونه برسند.

فرض بر این است که تابع $f(x)$ از یک فرآیند گاوسی پیشین حاصل شده است. مشاهدات به صورت زیر تعریف می‌شود (معادله (۱.۴)):

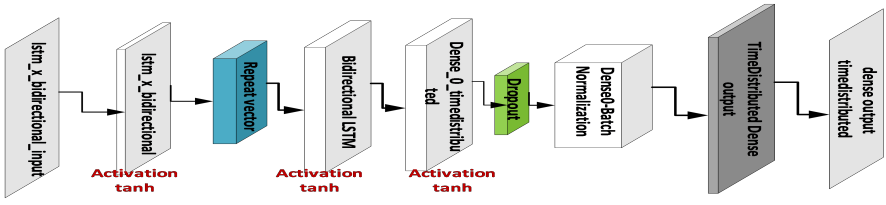
$$\{x_n, y_n\}_{n=1}^{N}, \quad y_n \sim \mathcal{N}(f(x_n), \nu) \tag{۱.۴}$$

در معادله (۱.۴)، ν واریانس و تابعی از مشاهدات است. فرآیند گاوسی (GP) به‌صورت شهودی اغلب مشابه یک تابع در نظر گرفته می‌شود، با این تفاوت که به‌جای برگرداندن یک اسکالر $f(x)$ برای x دلخواه، میانگین و واریانس یک توزیع نرمال را بازمی‌گرداند. برای ساده‌تر شدن، فرض می‌شود میانگین توزیع پیشین تابع، مقدار صفر است ($m(x) = 0$)، بنابراین به انتخاب تابع هسته $K(x)$ پرداخته می‌شود. یکی از مؤلفه‌های کلیدی بهینه‌سازی بیزی، تابع اکتساب است که فرآیند جستجو را با ایجاد تعادل و مصالحه بین اکتشاف و بهره‌برداری از فضای جستجو هدایت می‌کند. در قسمت‌های قبلی نقش توابع اکتساب در بهینه‌سازی بیزی بررسی شد، همچنین چند نمونه رایج مورد بحث و بررسی قرار گرفت.

بهینه‌سازی بیزی یک فرآیند تکراری است. به این صورت که یک مدل جایگزین احتمالی، اغلب یک فرآیند گاوسی (GP) را با یک تابع اکتساب ترکیب می‌کند تا نقطه بعدی برای ارزیابی در یک تابع هدف انتخاب شود. مدل جایگزین نشان‌دهنده درک فعلی و عدم قطعیت ما از تابع هدف است در حالی که تابع اکتساب به ایجاد تعادل بین کاوش مناطق جدید فضای جستجوی ورودی و بهره‌برداری از مناطق با عملکرد پیش‌بینی‌شده بالا کمک می‌کند. تحت فرآیند گاوسی، این توابع تنها از طریق تابع میانگین پیش‌بینی به مدل بستگی دارند. همانگونه که درقسمت قبل بررسی شد، به جای انتخاب پیش فرض هسته نمایی مربعی، از هسته ARD Matern 5/2 و به شکل زیر استفاده می‌شود .(معادله (۲.۴))

$$K_{M52}(x, x') = \theta_0 \left(1 + \sqrt{5 \sum_{n=1}^{N} \frac{(x_n - x'_n)^2}{\theta_n^2}} + \frac{5}{3}(x, x')\right) \exp\left(-\sqrt{5 \sum_{n=1}^{N} \frac{(x_n - x'_n)^2}{\theta_n^2}}\right) \tag{۲.۴}$$

در معادله (۲.۴)، N تعداد ویژگی‌ها (در اینجا تعداد ابرپارامترهایی که باید بهینه شود) می‌باشد. مدل مورد استفاده براساس Hyperparameter-tuning و به صورت زیر تعریف می‌شود (شکل ۶):



شکل ۶: مدل شبکه عصبی چند لایه ایجاد شده بر اساس الگوریتم پیشنهادی جهت تنظیم ابرپارامترها

۱.۴. فاز آموزش. یکی از محدودیت‌های اصلی بهینه‌سازی بیزی مبتنی بر GP این است که هزینه محاسباتی این تکنیک به صورت مکعبی با افزایش تعداد مشاهدات، افزایش می‌یابد. این محدودیت کارایی روش را برای مسائلی که در آنها تعداد مشاهدات کم است محدود می‌کند. برای رفع این مشکل مدل GP با مدلی که اغلب ویژگی‌های مطلوب GP مانند انعطاف‌پذیری و عدم قطعیت را حفظ می‌کند، جایگزین می‌شود. به عبارت دیگر یک رگرسیون خطی بیزی به آخرین لایه پنهان شبکه عصبی اضافه و فقط وزن‌های خروجی شبکه به حاشیه برده می‌شود. برای سایر پارامترهای باقی‌مانده از تخمین نقطه ای استفاده می‌شود. این رویکرد منجر به رگرسیون پایه تطبیقی شده و امکان تبادل و مصالحه بین زمان ارزیابی و ظرفیت مدل را فراهم می‌کند. به این ترتیب با استفاده از توابع غیرخطی بسیار انعطاف‌پذیر و قدرتمند تعریف شده توسط شبکه عصبی اساس کار تشکیل می‌شود. توابع پایه از طریق وزن‌ها و بایاس‌های شبکه عصبی عمیق پارامترسازی شده و این پارامترها با استفاده از الگوریتم پس انتشار[1] و نزول گرادیان تصادفی با تکانه[2] آموزش داده می‌شوند. این روش را می‌توان به عنوان حداکثر تخمین پسینی (MAP[3]) از تمام پارامترهای شبکه مشاهده کرد. بطور کلی هنگامی که این «شبکه عصبی تابع پایه» آموزش داده شد، لایه خروجی پارامتری MAP با یک رگرسیون خطی بیزی که نشان‌دهنده عدم اطمینان در وزن‌ها می‌باشد، جایگزین می‌شود. روش حداکثر احتمال پسین (MAP) یک رویکرد بیزی است که برای تخمین محتمل‌ترین مقدار یک پارامتر با توجه به داده‌های مشاهده‌شده و دانش قبلی درباره آن پارامتر به کار می‌رود. وزن‌ها و بایاس‌های شبکه عصبی با استفاده از الگوریتم پس‌انتشار و نزول گرادیان تصادفی آموزش داده می‌شوند تا یک برآورد MAP از تمام پارامترها حاصل شود.

MAP یک رویکرد بیزی می‌باشد که برای برای تخمین محتمل‌ترین مقدار یک پارامتر با توجه به داده‌های مشاهده شده و دانش قبلی در مورد آن پارامتر استفاده می‌شود. تخمین MAP می‌تواند با

[1] Backpropagation
[2] SGD with Momentum
[3] Maximum a posteriori estimation

متعادل کردن احتمال داده‌ها با توزیع قبلی پارامتر حاصل شود. با این کار راهی برای ترکیب دانش قبلی در فرایند برآورد حاصل می‌شود.

فرض می‌شود داده‌های ورودی به شکل زیر باشد (معادله (۳.۴)):

$$\text{input}: \quad X = \{(x_i, y_i)\}_{i=1}^{N} \tag{۳.۴}$$

بردارهای خروجی (توابع پایه) از آخرین لایه پنهان شبکه آموزش‌دیده بر روی داده‌های ورودی و مقادیر تابع هدف به شکل زیر هستند (معادله (۴.۴)):

$$\phi[\cdot] = [\varphi_1(\cdot), \ldots, \varphi_M(\cdot)]^T \tag{۴.۴}$$

که در آن M اندازه آخرین لایه پنهان بوده و بر روی داده‌های ورودی و مقادیر هدف

$$\{(x_n, y_n)\}_{n=1}^{N} \subset \mathbb{R}^K \times \mathbb{R}$$

آموزش داده می‌شود. حالا فرض می‌شود ماتریس خروجی مدل از داده‌ها و توابع پایه به شکل زیر باشد (معادله (۵.۴)):

$$\Phi_{ni} = \varphi_i(x_n) \tag{۵.۴}$$

با این فرض که w_{MAP} وزن‌های هر لایه از شبکه عصبی باشند، خروجی مدل به صورت معادله (۶.۴) بیان می‌شود:

$$\Phi_{ni}, \; w_{\text{MAP}} \tag{۶.۴}$$

نتایج حاصل از فاز آموزش در جدول ۱ آمده است. طبق جدول ۱ بعد از ۵۹ تکرار نتیجه مطلوب حاصل شده است. در این حالت تعداد ابرپارامترهای مدل ۱۶ تا بوده که با استفاده از روش پیشنهادی بهترین مقدار قابل محاسبه و دستیابی است. نتایج حاصل از اجرای نرم افزار پایتون به صورت زیر گزارش شده است:

Restoring model weights from the end of the best epoch: 59- loss: 0.2898 - VAL_LOSS: 0.2062 - Lr: 1.0000e-06 - 301ms/epoch - 25ms/step- Total params: 101,937-Trainable params: 101,921- non-trainable params: 16

جدول ۱: نتایج حاصل از اجرای فاز آموزش

پارامتر	شکل خروجی	لایه (نوع)
۳۵۳۲۸	(۶۴، هیچ کدام)	LSTM_X_BIDIRECTIONAL (Bidirectional)
۰	(۶۴،۱۳، هیچ کدام)	repeat_vector_1 (Repeat vector)
۶۶۰۴۸	(۶۴،۱۳، هیچ کدام)	LSTM_0_BIDIRECTIONAL (Bidirectional)
۵۲۰	(۸،۱۳، هیچ کدام)	dense_0_timedistributed (Time Distributed)
۰	(۸،۱۳، هیچ کدام)	DENSE_0_DROPOUT (Dropout)
۳۲	(۸،۱۳، هیچ کدام)	DENSE_0_BATCHNORM (Batch Normalization)
۰	(۱،۱۳، هیچ کدام)	DENSE_OUTPUT_TIMEDISTRIBUTED (Time distributed)

۲.۴. نتایج عددی و بررسی مدل پیشنهادی. پس از آموزش، لایهٔ خروجی پارامتر شده با MAP (یعنی لایهٔ خروجی و وزن‌های مرتبط با آن از w_{MAP}) با یک رگرسیون خطی بیزی جایگزین شده که عدم قطعیت در وزن‌ها را ثبت می‌کند. در این حالت داریم (معادله (۷.۴)):

$$\text{output}, y = w_{\text{regressor}}^{T} * \boldsymbol{\phi}[\cdot] + \text{noise} \tag{۷.۴}$$

در معادله (۷.۴) فرض می‌شود وزن‌های رگرسیون $w_{\text{regressor}}$ دارای یک توزیع نرمال به صورت معادله (۸.۴) می‌باشند:

$$w_{\text{regressor}} \sim \mathcal{N}(۰, \alpha^{-۱} I) \tag{۸.۴}$$

با فرض توزیع معادله (۹.۴) برای تابع هدف، پیش‌بینی بر روی t به‌صورت یک توزیع نرمال با میانگین (معادله (۱۰.۴)) و واریانس (معادله (۱۱.۴)) زیر است:

$$y \sim P(t \mid w_{\text{regressor}}, \beta, X) = \mathcal{N}(w_{\text{regressor}}^{T} * \boldsymbol{\phi}[\cdot], \frac{۱}{\beta}) \tag{۹.۴}$$

$$\text{mean} = m^{T} \boldsymbol{\phi}(X) + \rho(X) \tag{۱۰.۴}$$

$$\text{variance} = \boldsymbol{\phi}(X)^{T} K^{-۱} \boldsymbol{\phi}(X) + \frac{۱}{\beta} \tag{۱۱.۴}$$

در معادلهٔ (۱۰.۴)، $\rho(X)$ تابع میانگین برای به‌دست آوردن توزیع قبلی مبنی بر اینکه تابع با یک تابع درجه دوم محدب در مرکز ناحیه تقریب زده می‌شود، می‌باشد. به‌عبارت دیگر، یک فرض معمول

می‌باشد برای حصول اطمینان از اینکه هنگام جستجوی مرز فضای جستجو، نقطهٔ بهینه در فضای داخلی فضای ورودی قرار دارد.

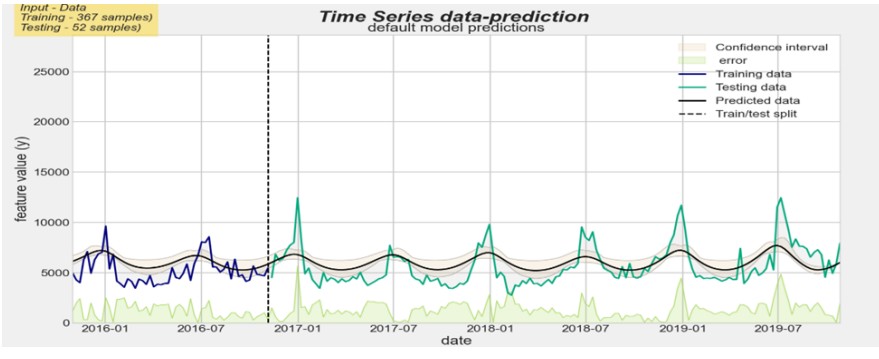
در معادلات (۱۰.۴) و (۱۱.۴) داریم:

$$m = \beta K^{-1} \phi(X)^T (y - \rho(X)) \tag{۱۲.۴}$$

و

$$K = \beta \phi(X)^T \phi(X) + \alpha I \tag{۱۳.۴}$$

مدل پیشنهادی در دو حالت مورد بررسی قرار می‌گیرد. حالت اول، خروجی شبکه بدون بهینه‌سازی ابرپارامترها می‌باشد. نتایج حالت اول در شکل‌های ۷ و ۸ آمده است.

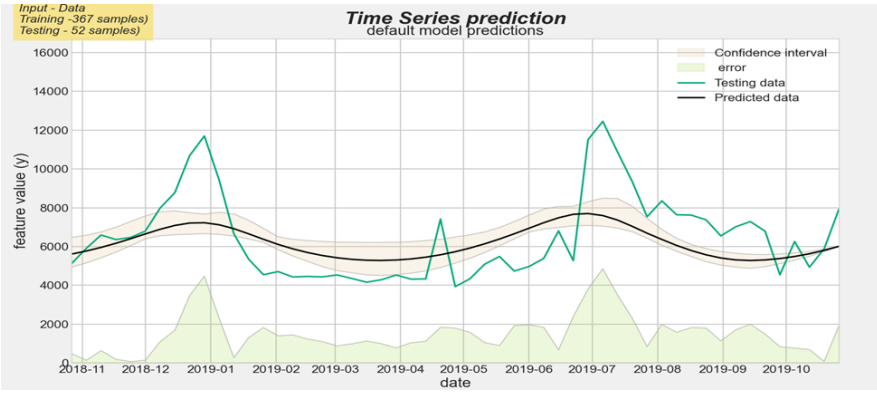


شکل ۷: نمایشی از عملکرد شبکه عصبی مورد استفاده بدون بهینه‌سازی ابرپارامترها (فاز آموزش)

در ادامه روش پیشنهادی جهت بهینه‌سازی ابرپارامترها اجرا شده و نتایج حاصل با نتایج مرحله قبل مقایسه می‌شود. نتایج حاصل از آموزش مدل در حالت دوم در جدول ۲ و شکل‌های ۹ و ۱۰ نشان داده شده است. در شبیه‌سازی مدل از کتابخانه‌های پایتون شامل تنسورفلو و SKLEARN و SEABRON استفاده شده است. در این مقاله یک مدل برای پیش‌بینی سری‌های زمانی پیشنهاد شد که به منظور افزایش دقت پیش‌بینی بهینه‌سازی ابرپارامترهای مدل با استفاده از فرضیاتی مانند فرض گوسی بودن توزیع در رگرسیون و تابع هدف انجام شد.

ابرپارامترهایی که مقدار بهینه آنها با روش پیشنهادی برای ماژول طبقه‌بندی‌کننده به دست آمده است عبارتند از:

(۱) تعداد واحدها در لایه پنهان



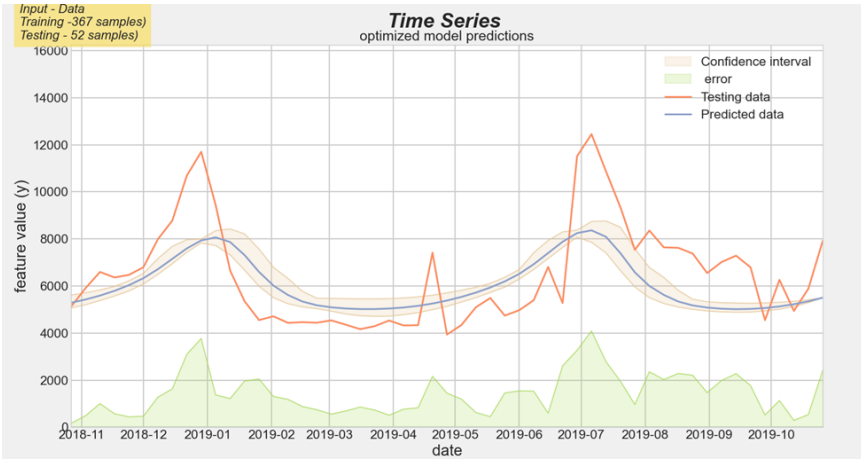
شکل ۸: نمایشی از عملکرد شبکه عصبی مورد استفاده بدون بهینه‌سازی ابرپارامترها

(۲) پارامتر پراکندگی که نشان‌دهنده میانگین فعال‌سازی مطلوب هر واحد پنهان است.
(۳) پارامتر کاهش وزن در تابع هزینه کلی
(۴) وزن دوره جریمه پراکندگی

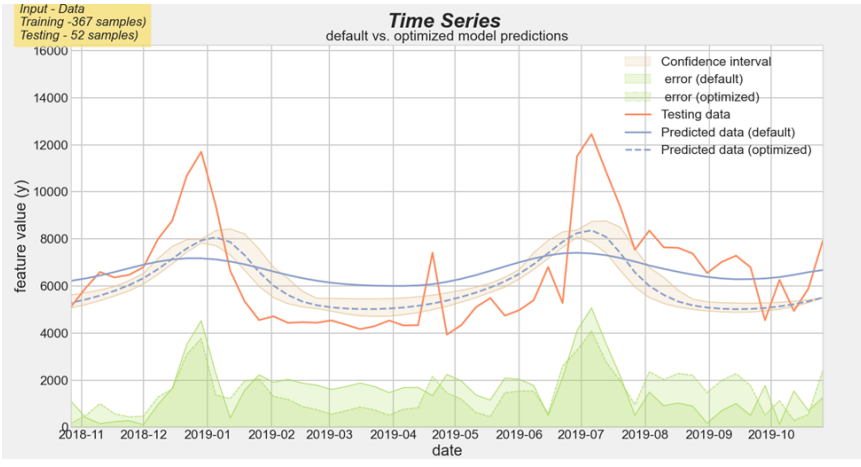
جدول ۲: مقایسه نتایج در دو حالت با استفاده از بهینه‌سازی ابرپارامترها و بدون استفاده از بهینه‌سازی ابرپارامترها در فاز آموزش

مرحله	دقت (%)	اندازه لایه پنهان	پارامتر پراکندگی	پارامتر کاهش وزن	جریمه پراکندگی
اولیه	۹۷٫۱۰	۱۰۰	۰٫۱۲	۰٫۰۰۱۱	۱
بعد از ۲۰ تکرار	۹۷٫۳۶۵۴	۲۰۶	۰٫۳۳۸۷۷	۰٫۰۰۸۱۹۲	۳
بعد از ۴۰ تکرار	۹۷٫۷۷۷۳	۳۰۰	۰٫۱۱	۰٫۰۰۳۱	۳
با تنظیم ابرپارامترها	۹۷٫۶۷۸	۲۰۰	۰٫۳۱	۰٫۰۰۵۵۲	۳

۳.۴. تأثیر نویز و پراکندگی داده‌ها بر بهینه‌سازی بیزی. یکی از چالش‌های مهم در بهینه‌سازی بیزی وجود داده‌های نویزی یا داده‌های دارای پراکندگی زیاد است که ممکن است باعث کاهش دقت پیش‌بینی و سخت‌تر شدن آموزش مدل شود. نویز در داده‌ها باعث می‌شود مدل نتواند به خوبی ساختار اصلی تابع هدف را یاد بگیرد و احتمالاً جواب‌ها یا پیش‌بینی‌های غیرقابل اعتماد ارائه



شکل ۹: نمایشی از عملکرد شبکه عصبی مورد استفاده به همراه مدل بهینه‌سازی ابرپارامترها



شکل ۱۰: نمایشی از عملکرد و مقایسه شبکه عصبی مورد استفاده در دو حالت بدون استفاده از بهینه‌سازی ابرپارامترها و همراه با مدل بهینه‌سازی ابرپارامترها

دهد. برای کاهش اثرات نویز می‌توان تنظیمات خاصی روی مدل فرآیند گاوسی مثل انتخاب توابع هسته‌ای که در برابر نویز مقاوم‌تر هستند یا اضافه کردن روش‌های منظم‌سازی که از پیچیدگی بیش از حد مدل جلوگیری کنند، انجام داد. این کارها به حفظ تعادل بین انعطاف‌پذیری و پایداری مدل کمک می‌کنند. پیش‌پردازش داده‌ها مثل صاف‌سازی یا فیلتر کردن نویز، و همچنین استفاده از

توزیع‌های احتمال به جای مقادیر نقطه‌ای نیز می‌تواند کمک زیادی به کاهش اثرات نویز کند. در نهایت جمع‌آوری داده‌های با کیفیت بالاتر و استفاده از داده‌های مکمل نیز از جمله راهکارهای مفید برای بهبود عملکرد مدل در شرایط نویزی است. با توجه به نکات گفته شده، روش پیشنهادی قادر است در شرایطی که داده‌ها نویزی یا دارای پراکندگی زیاد هستند نیز عملکرد قابل قبولی ارائه دهد و کارایی خود را حفظ کند.

۵. نتیجه‌گیری

در این مقاله یک مدل برای پیش‌بینی سری زمانی پیشنهاد شد که در آن از بهینه‌سازی بیزی برای تنظیم ابرپارامتر با مدل فرآیند گاوسی استفاده شده است. از مزایای روش بهینه‌سازی بیزی کارایی بالا، انعطاف‌پذیری آن در مواجه شدن با توابع هدف غیرخطی و حتی پیچیده، قابلیت تنظیم و تطبیق با انواع مسائل و مدیریت عدم قطعیت داده‌ها و مدل می‌باشد. همچنین در روش بیزی به منظور افزایش سرعت همگرایی در فرآیند بهینه‌سازی از اطلاعات پیشین استفاده می‌شود. برای اینکه مدل پیشنهادی مزیت بزرگ صرفه‌جویی در زمان را داشته باشد، به صورت خطی با تعداد مشاهدات مقیاس‌بندی شده است. ابرپارامترهای تقریبی به دست آمده با بهینه‌سازی بیزی از کارایی خوب و مناسبی برخوردار است. بهینه‌سازی بیزی نوعی راه حل «یک برای همه» ارائه می‌دهد که برای مسائل مختلف بهینه‌سازی و یادگیری ماشین قابل استفاده است. بهینه‌سازی بیزی در مسائلی که تابع هدف پیچیده و یا ابعاد ویژگی‌ها بسیار بالا باشد ممکن است از کارایی لازم برخوردار نباشد. در مواردی نیز ممکن است نیاز به توسعه مدل‌های خاص برای مساله مورد نظر باشد و کاربر باید برخی از باورهای قبلی را داشته باشد تا در مدل گنجانده شود. در این حالت دقت روش بهینه‌سازی بیزی به انتخاب مدل جایگزین بستگی دارد. با این وجود، بهینه‌سازی بیزی با ارائه تقریب خوب و دقیق برای پارامترهای تنظیم شده به کاربر کمک می‌کند و به طور قابل‌توجهی در زمان مورد نیاز برای جستجوی بیشتر و انتخاب مقادیر مناسب صرفه‌جویی می‌کند. روش بهینه‌سازی بیزی به دلیل توانایی در مدیریت عدم قطعیت و تنظیم خودکار مدل‌ها در بسیاری از مسائل کاربردی می‌تواند مورد استفاده قرار گیرد. به عنوان مثال در زمینه‌ی پیش‌بینی بازارهای مالی، این روش می‌تواند به بهبود دقت مدل‌های یادگیری ماشین به‌ویژه در شرایطی که داده‌ها دارای نوسان زیاد و غیرقابل پیش‌بینی هستند، کمک کند. در حوزه‌ی مدیریت انرژی، استفاده از این روش برای تنظیم پارامترهای سیستم‌های کنترل مصرف، می‌تواند موجب بهینه‌سازی منابع و کاهش هزینه‌ها شود. در حوزه سلامت نیز، برای مدلسازی داده‌های مربوط به بیماران و تشخیص زودهنگام بیماری‌ها، کاربردهای گسترده‌ای دارد. البته همانطور که گفته شد در برخی مسائل خاص مانند مدل‌هایی با ابعاد

بسیار زیاد یا توابع هدف دارای نویز زیاد ممکن است کارایی این روش کاهش یابد. در این موارد استفاده ترکیبی از بهینه‌سازی بیزی با سایر روش‌ها می‌تواند راه‌گشا باشد. در کارهای بعدی می‌توان از روش‌های بهینه‌سازی ابتکاری مانند الگوریتم بهینه‌سازی گروه ذرات و الگوریتم جستجوی گرانشی و غیره برای تنظیم ابرپارامترها استفاده کرد [۱۰، ۱۱]. می‌توان نتایج حاصل از این روش‌ها را با روش پیشنهادی در این مقاله مورد مقایسه و ارزیابی قرار داد.

مراجع

[۱] طالبی خارزنی، م. و بنی عامریان، م. (۱۴۰۲) پیش‌بینی بازار فارکس با استفاده از شبکه عصبی ترکیبی و یادگیری انتقال. پنجمین کنفرانس ملی مدیریت و صنعت گردشگری، تهران. قابل بازیابی از: https://civilica.com/doc/1745697

[۲] غفاری رزین، س.ر.، هوشنگی، ن. و وثوقی، ب. (۱۴۰۲) ارزیابی کارایی مدل شبکه عصبی حافظه کوتاه‌مدت طولانی در پیش‌بینی سری زمانی یونوسفر و مقایسه آن با مدل‌های GRNN، GIM و NeQuick. فصلنامه علمی–پژوهشی اطلاعات جغرافیایی «سپهر»، سال ۳۲، شماره ۱۲۶، صص ۱۱۵–۱۲۹.
Doi: 10.22131/sepehr.2023.547749.2839

[۳] فرزاد، ع.، دهقان منشادی، ه. و دشتی رحمت‌آبادی، م. (۱۴۰۲) پیش‌بینی مسائل مربوط به زمان‌بندی پروژه‌های عمرانی با استفاده از شبکه عصبی LSTM (حافظه طولانی کوتاه‌مدت). نشریه مهندسی عمران امیرکبیر، سال ۵۵، شماره ۹، صص ۱۷۵۳–۱۷۶۴.
Doi: 10.22060/ceej.2023.21383.7701

[۴] یراقی، م. و ربیعی، ع. (۱۳۹۹) مرور و مقایسه الگوریتم‌های شبکه عصبی بازگشتی عمیق LSTM و GRU در مدل‌سازی داده‌های سری زمانی نرخ ارز. در: علوم رایانشی، صص ۴۰–۵۰.

[5] Abbasimehr, H., & Paki, R. (2021) Improving time series forecasting using LSTM and attention models. Journal of Ambient Intelligence and Humanized Computing, 13, 673–691. Available: https://api.semanticscholar.org/CorpusID:234311110

[6] Abbasimehr, H., Shabani, M., & Yousefi, M. (2020) An optimized model using LSTM network for demand forecasting. Computers & Industrial Engineering, 143, 106435. https://doi.org/10.1016/j.cie.2020.106435

[7] Baek, Y., & Kim, H. Y. (2018) ModAugNet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module and a prediction LSTM module. Expert Systems with Applications, 113, 457–480. https://doi.org/10.1016/j.eswa.2018.07.019

[8] Eid, M. M., Khodadadi, N., & Mirjalili, S. (2022) Meta-heuristic optimization of LSTM-based deep network for boosting the prediction of monkeypox cases. [Online], pp. 1–20.

[9] ElSaid, A., El Jamiy, F., Higgins, J., Wild, B., & Desell, T. (2018) Optimizing long short-term memory recurrent neural networks using ant colony optimization to predict turbine engine vibration. Applied Soft Computing, 73, 969–991. https://doi.org/10.1016/j.asoc.2018.09.013

[10] Ghanbari, N., Zahiri, H., & Shahraki, H. (2022) Clustering of fuzzy data sets based on particle swarm optimization with fuzzy cluster centers. International Journal of Industrial Engineering & Production Research, 33(2), 1–12.

[11] Ghanbari, N., Zahiri, S. H., & Shahraki, H. (2024) Clustering of triangular fuzzy data based on heuristic methods. Journal of Electrical and Computer Engineering Innovations, 12(1), 1–14. https://doi.org/10.22061/jecei.2023.9641.645

[12] Hajizadeh Tahan, M., Ghasemzadeh, M., & Rezaeian, M. (2021) An evolutionary attention-based deep long short-term memory for time series prediction. Computational Intelligence and Electrical Engineering, 11(4), 15–28.

[13] Hochreiter, S., & Schmidhuber, J. (1997) Long short-term memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[14] Izadi, M. (2013) Forecasting inflation rate in Kermanshah province with LSTM neural network.

[15] Karevan, Z., & Suykens, J. A. K. (2020) Transductive LSTM for time-series prediction: An application to weather forecasting. Neural Networks, 125, 1–9. https://doi.org/10.1016/j.neunet.2019.12.030

[16] Ma, R., Zhu, X., & Yan, L. (2020) A hybrid approach for clustering uncertain time series. Journal of Computing and Information Technology, 28(4), 255–267. https://doi.org/10.20532/cit.2020.1004802

[17] Moghar, A. (2020) A LSTM recurrent stock market prediction using neural network. Procedia Computer Science, 170, 1168–1173. https://doi.org/10.1016/j.procs.2020.03.049

[18] Sagheer, A., & Kotb, M. (2019) Time series forecasting of petroleum production using deep LSTM recurrent networks. Neurocomputing, 323, 203–213. https://doi.org/10.1016/j.neucom.2018.09.082

[19] Science, N., Phenomena, C., Kumar, V., Chimmula, R., & Zhang, L. (2020) Time series forecasting of COVID-19 transmission in Canada using LSTM networks. Chaos, Solitons and Fractals, 135.

[20] Surakhi, O., et al. (2021) Time-lag selection for time-series forecasting using neural network and heuristic algorithm. pp. 1–22.