دانشگاه آزاد اسلامی واحد تهران شمال

گروه مهندسی صنایع

پایان نامه برای دریافت درجه کارشناسی

در رشته مهندسی صنایع

موضوع:

پیش بینی قیمت سهام شرکت Apple با استفاده از شبکه های عصبی و مدل های سری زمانی

استاد راهنما:

دکتر شروین اسدزاده

دانشجو:

علی استادی

پاییز ۹۹

بسم الله الرحمن الرحیم

تقدیم به :

۱۷٦ پرنده که در ۱۸ دی ماه ۹۸ دیگر در آسمان پرواز نکردند ...

سپاس گزاری :

ضمن تقدیر و تشکر شایسته و در کمال افتخار و امتنان، برخود واجب می دانم مراتب سپاس بیکران خود را از زحمات استاد فرزانه و فرهیخته، جناب آقای دکتر شروین اسدزاده ابرازدارم؛ که در راه کسب علم و معرفت ما را یاری نمودند و همچنین از تمام استاید دلسوز و دانشمند خود که در طی این چند سال هیچ گاه از علم و دانش خود بر من دریغ نکردند کمال تشکر را دارم .

پروردگارا! حسن عاقبت، سلامت و سعادت را برای ایشان و همه آنان که در راه کسب دانش راهنمایمان بودند مقدر نما

چکیده

پیشبینی تغییر قیمت سهام به عنوان یک فعالیت چالش انگیز در پیش بینی سری های زمانی مالی در نظر گرفته میشود. یک پیشبینی صحیح از تغییر قیمت سهام میتواند سود زیادی را برای سرمایه گذاران به بار آورد. با توجه به پیچیدگی داده های بازار بورس، توسعه مدل های کارآمد برای پیش بینی بسیار دشوار است. .در عصر حاضر با توجه به پیشرفت فناوری در زمینه علوم کامپیوتر و فراگیر شدن آن در علوم مختلف، زمینه های استفاده از شبکه های عصبی مصنوعی با توجه به سرعت بسیار بالای پردازش درکامپیوترها به وجود آمده است. این شبکه ها با استفاده ازقابلیت یادگیری خود هر گونه تغییری در قوانین نهفته درسری های زمانی را فرا گرفته و برای پیشبینی آینده از آن استفاده می کنند.

در این پژوهش قصد داریم کارایی مدل های سری زمانی (MA_AR_ARIMA) را با کارایی مدل های شبکه ی عصبی عمیق (ANN_RNN_LSTM_GRU) بر روی داده های سهام شرکت Apple مقایسه کنیم و بهینه ترین مدل برای پیش بینی داده های سهام شرکت را انتخاب کنیم. نتایج بدست آمده در هر دو روش با معیارسنجش کارایی پیشبینی یعنی مجذور میانگین مربع خطا (RMSE)مقایسه شده است. یافته های این پژوهش نشان میدهد که هر دو روش، پیشبینی مناسبی ارائه داده اند لیکن مقایسه معیارهای سنجش کارایی بیانگر دقت بالاتر روش شبکه های عصبی عمیق نسبت به روش های سری زمانی در پیشبینی میباشد.

واژههای کلیدی: شبکه های عصبی مصنوعی عمیق _پیش بینی سهام با استفاده از شبکه های عصبی _پیش بینی سهام با استفاده از مدل های سری زمانی_مدل های سری زمانی

فهرست مطالب

"فصل اول" ۱۰

۱_۱ مقدمه ۱۱

۱_۲ طرح مسئله (تعریف و اهمیت موضوع) ۱۱

۱-۳ ضرورت انجام تحقیق ۱۲

۱-۴ سوالات تحقیق ۱۴

۱-۵هدف از انجام پژوهش ۱۴

۱-۶ نتایج مورد انتظارپس از انجام تحقیق ۱۴

۱-۷ روش تحقیق ۱۵

۱-۸ روش و ابزار گردآوری اطلاعات ۱۵

"فصل دوم" ۱۶

۲-۱ پیش نیاز های آماری ۱۷

۲-۱-۱کوواریانس(**Covariance**) ۱۷

۲-۱-۲همبستگی (**Correlation**) ۱۷

۲-۲تعریف سری زمانی ۱۸

۲-۳ چرا سری های زمانی ۱۸

۲-۴ مراحل پیش بینی با استفاده از مدل های سری زمانی ۱۹

۲-۴-۱مولفه های یک سری زمانی ۱۹
۲-۴-۱-۱ روند ۲۰
۲-۴-۱-۲ فصل (**Seasonal**) ۲۱
۲-۴-۱-۳ تغییرات نامعمول..... ۲۲
۲-۴-۲ سری زمانی ایستا و نا ایستا (**Stationary time series**) ۲۲
۲-۴-۲-۱ روند(**Trend**) ۲۳
۲-۴-۲-۲ ثابت بودن واریانس ۲۳
۲-۴-۲-۳ یکنواختی در واحد های زمانی ۲۴
۲-۴-۳ بررسی ایستایی سری های زمانی (آزمون دیکی فولر) ۲۵
۲-۴-۴ ایستا کردن سری زمانی ۲۶
۲-۴-۴-۱هموار سازی با استفاده از میانگین متحرک(**Moving Average**)..... ۲۶
۲-۴-۴-۲ هوار سازی نمایی (**Exponential Smoothing**): ۳۰
۲-۴-۴-۳ هموارسازی نمایی مضاعف (**Double Exponential Smoothing**)..... ۳۲
۲-۴-۵ تفاضل گیری (**Difference**) ۳۳
۲-۴-۶تابع خود همبستگی (**AFC**) ۳۴
۲-۴-۷ خودهمبستگی جزئی ۳۶
۲-۴-۸ مدل میانگین متحرک ۳۸
۲-۴-۹ مدل اتورگرسیو(خود همبسته) ۳۸
۲-۴-۱۰ مدل اتورگرسیو میانگین متحرک(**ARIMA**) ۴۰
۲-۴-۱۲معیار ارزیابی آکایکه (**AIC**)..... ۴۱
۲-۴-۱۳ پیش بینی و اندازه گیری خطا درآن ۴۴

"فصل سوم"..... ۴۶

۳-۱ پیدایش شبکه های عصبی مصنوعی ۴۷

۳-۲ ساختار نورون های مغز انسان ۴۹

۳-۲-۱ دندریت ها ۵۰
۳-۲-۳جسم سلولی یا هسته نورون ۵۰
۳-۲-۳ انتهای آکسون ۵۰
۳-۲-۴ ارتباط نورون ها با یکدیگر ۵۱

۳-۳ ساختار شبکه ی عصبی مصنوعی ۵۲
۳-۳-۱ نورون ها و گره ها در شبکه ی عصبی ۵۲
۳-۳-۲ وزن ها و بایاس ها در شبکه ی عصبی مصنوعی ۵۳
۳-۳-۳ تابع فعال ساز (**Activation Function**) ۵۴
۳-۳-۳-۱ تابع فعالسازی خطی ۵۵
۳-۳-۳-۲ تابع فعالسازی غیر خطی ۵۶

۳-۴ لایه های شبکه ی عصبی مصنوعی ۵۷
۳-۴-۱ لایه ی ورودی (**Input Layer**) ۵۷
۳-۴-۲ لایه ی پنهان و مفهوم **Fully Connected** ۵۷
۳-۴-۳ لایه ی خروجی (**Output Layer**) ۵۸
۳-۴-۳-۱رگرسیون ۵۸
۳-۴-۳-۲طبقه بندی دودویی ۵۸
۳-۴-۳-۳طبقه بندی چندتایی ۵۹

۳-۵ مدل های مختلف شبکه ی عصبی مصنوعی ۶۰
۳-۵-۱ مدل پرسپترون تک لایه (**SLP**) ۶۰
۳-۵-۲ مدل پرسپترون چند لایه (**MLP**) ۶۱
۳-۵-۲-۱یادگیری شبکه ی عصبی پرسپترون چندلایه ۶۲
۳-۵-۲-۱-۱ انتشار رو به جلو (**Forward Propagation**) ۶۳
۳-۵-۲-۱-۲ پس انتشار (**Back Propagation**) ۶۴
۳-۵-۲-۱-۳ الگوریتم بهینه ساز گرادیان کاهشی (**Gradient Descent**) ۶۴

۳-۵-۲-۱-۴ گرادیان کاهشی تصادفی(**Stochastic Gradient Descent**) ۶۵
۳-۵-۳ شبکه های عصبی بازگشتی (**Recurrent Neural Network**) ۶۷
۳-۵-۳-۱ تشریح پارامتر های **RNN** ۶۸
۳-۵-۳-۲تابع ضرر در **RNN** (**Loss Function**) ۷۰
۳-۵-۳-۳پس انتشار خطا در **RNN** (**Back Propagation Error**)..... ۷۰
۳-۵-۳-۴انواع مختلف ساختار **RNN** با توجه به کاربرد آن ها..... ۷۰
۳-۵-۳-۵ محو شدن و منفجر شدن گرادیان(**Vanishing and Exploding gradient**) ۷۲
۳-۵-۴ **Gated Recurrent Unit (GRU)** ۷۳
۳-۵-۵ **Long Short Term Memory(LSTM)**..... ۷۵

۳-۶ کاربرد شبکه های عصبی مصنوعی ۷۷

"فصل چهارم" ۷۸

۴-۱ مقدمه ۷۹

۴-۲ پیاده سازی الگوریتم های سری زمانی بر روی دیتا ۷۹
۴-۲-۱ بررسی ساختار سری زمانی..... ۷۹
۴-۲-۲ آزمون دیکی فولر بر روی دیتا..... ۸۰
۴-۲-۳ تفاضل گیری دیتا ۸۱
۴-۲-۳ نمودار های **ACF**و **PACF** روی دیتا..... ۸۲

۴-۳ پیاده سازی الگوریتم های شبکه ی عصبی بر روی دیتا ۸۴
۴-۳-۱ معماری شبکه های عصبی پیاده سازی شده ۸۵
۴-۳-۲ نمودار های پیش بینی شبکه های عصبی ۸۷

۴-۴ مقایسه ی عملکرد همه ی الگوریتم ها و نتیجه گیری..... ۸۹

۴-۵منابع و مآخذ ۹۲

فهرست اشکال

شکل ۱_ ۱:ده ردیف اول دیتا ۱۵

شکل ۲_ ۱ : دیاگرام عملکردی استراتژی مدلسازی به روش باکس-جنکینز(آریما) ۱۹

شکل ۲_ ۲ : سری زمانی با روند افزایشی ۲۰

شکل ۲_ ۳: سری زمانی با روند ثابت ۲۰

شکل ۲_ ۴: سری زمانی با روند کاهشی ۲۱

شکل ۲_ ۵: سری زمانی فصلی ۲۱

شکل ۲_ ۶: سری زمانی دارای تغییرات نامعمول ۲۲

شکل ۲_ ۷: سری زمانی ایستا و نا ایستا بر اساس میانگین ۲۳

شکل ۲_ ۸: سری زمانی ایستا و نا ایستا بر اساس ثابت بودن واریانس ۲۴

شکل ۲_ ۹: سری زمانی ایستا و نا ایستا بر اساس در واحد های زمانی ۲۴

شکل ۲_ ۱۰: آزمون دیکی فولر ۲۵

شکل ۲_ ۱۱: هوار سازی سری زمانی با استفاده از میانگین متحرک با۲۴ =lag ۲۷

شکل ۲_ ۱۲: هوار سازی سری زمانی با استفاده از میانگین متحرک با ۱۲ =lag ۲۸

شکل ۲_ ۱۳: هموار سازی با استفاده از میانگین متحرک مرتبه ۳ و ۵ ..... ۳۰

شکل ۲_ ۱۴:هموار سازی سری با استفاده از هموار سازی نمایی ۳۱

شکل ۲_ ۱۵:نمودار هموارسازی شده نمایی ۳۲

شکل ۲_ ۱۶: تابع خود همبستگی ۳۴

شکل ۲_ ۱۷: نمودار ACF یک سری زمانی نا ایستا ۳۵

شکل ۲_ ۱۸: نمودار خود همبستگی جزئی ۳۷

شکل ۲_ ۱۹: بررسی تعداد پارامتر بهینه با توجه به AIC ۴۳

شکل ۳- ۱: ساختار یک نورون..... ۴۹

شکل ۳- ۲: ارتباط میان نورون ها ۵۱

شکل ۳- ۳: ساختار یک نورون در شبکه ی عصبی مصنوعی ۵۲

شکل ۳- ۴: تابع فعالساز خطی ۵۵

شکل ۳- ۵: تابع فعالساز غیر خطی سیگموید. ۵۶

شکل ۳- ۶:یک عنصر پردازشی ۶۰

شکل ۳- ۷: ساختار کلی شبکه های عصبی چند لایه ی پیشخور..... ۶۱

شکل ۳- ۸:نمایش هندسی_ریاضی الگوریتم بهینه ساز گرادیان کاهشی ۶۵

شکل ۳- ۹:فرمول به روز کردن وزن ها (پارامتر ها) در گرادیان کاهشی ۶۶

شکل ۳- ۱۰: فرمول مشتق جزئی وزن ها در گرادیان کاهش ۶۶

شکل ۳- ۱۱: مکانیزم یاد گیری و بهبود پارامتر ها در شبکه ی عصبی پرسپترون چند لایه ۶۶

شکل ۳- ۱۲:مکانیزم یک شبکه عصبی برگشتی ۶۷

شکل ۳- ۱۳:نمایی از چند گام شبکه ی RNN ۶۹

شکل ۳- ۱۴: مکانیزم GRU ۷۴

شکل ۳- ۱۵: یک واحد LSTM ۷۶

شکل ۴_ ۱:نمودار سری زمانی سهام شرکت اپل ۷۹

شکل ۲۴_: نمودار سری زمانی ایستا شده ۸۱

شکل ۴_ ۳ : نمودار خود همبستگی با lag =2500..... ۸۲

شکل ۴_ ۴: نمودار خود همبستگی سری زمانی ایستا..... ۸۲

شکل ۵۴ _ : نمودار خود همبستگی جزئی سری زمانی ایستا ۸۳

شکل ۴_ ۶:مدل پیشنهادی AutoArima..... ۸۳

شکل ۴_ ۷: نمودار پیش بینی مدل ARIMA ۸۴

شکل ۴_ ۸:ساختار شبکه ی LSTM پیاده سازی شده ۸۵

شکل ۴_ ۹:ساختار MLP پیاده سازی شده ۸۶

شکل ۴_ ۱۰: نمودار پیش بینی سهام توسط MLP ۸۷

شکل ۴_ ۱۱: نمودار پیش بینی سهام توسط Simple RNN ۸۷

شکل ۴_ ۱۲: نمودار پیش بینی سهام توسط GRU..... ۸۸

شکل ۴_ ۱۳: نمودار پیش بینی سهام توسط LSTM..... ۸۸

شکل ۴_ ۱۴: مقایسه ی نمودار های همه ی الگوریتم های پیاده سازی شده ۸۹

شکل ۴_ ۱۵: نمودار ستونی مقایسه ی RMSE ۹۰

فهرست جداول

جدول ۲- ۱: تفاضل گیری سری زمانی با دو مرتبه ی ۳ و ۵ ..... ۲۹

جدول ۳- ۱:انواع مختلف RNN با توجه به کاربرد آن ها ۷۱

جدول ۴_ ۱: RMSR Score ۹۰

"فصل اول"

"**کلیات تحقیق**"

۱_۱ مقدمه

در این بخش برای آشنایی بیشتر با موضوع پروژه پایانی، توضیح مختصری در مورد عنوان کلی،ضرورتتحقیق،

سوال های تحقیق و فرآیند تحقیق داده می شود.

۱_۲ طرح مسئله (تعریف و اهمیت موضوع)

همانطور که میدانیم سرمایه و نیروی کار از ارکان اصلی تولید هستند و تأمین این عوامل و تخصیص بهینه آنها لازمه رشد اقتصادی است. این تخصیص مستلزم وجود بازار و عملکرد مطلوب نیروهای بازار است. در رابطه با سرمایه، بازار بورس میتواند این وظیفه را بر عهده داشته باشد. مهمترین وظیفه بازار بورس، جذب سرمایه های پراکنده و هدایت آنها به سوی فعالیت های سرمایه گذاری از طریق یک فرآیند تخصیص بهینه است. سرمایه گذاران با انگیزه دریافت عواید از دوکانال سود حاصل از فعالیت شرکتی که سهام آن را خریداری نموده اند و همچنین فروش مجدد سهام وارد عرصه سرمایه گذاری میشوند. نوسان قیمت سهام در تمام بازارهای بورس امری طبیعی و عادی است، اما درهر صورت میتوان با یک پیشبینی از قیمت سهام ترکیبی مطلوب از آنها را انتخاب و نوسانها را کاهش داد و از این طریق میزان اطلاعات در دسترس افراد را افزایش داد. به نظر میرسدکه افزایش اطلاعات در بازار به عملکرد بهتر آن منجر خواهد شد. پیشبینی در بازار بورس میتواند گامی درجهت افزایش و شفاف نمودن اطلاعات در بازار سرمایه باشد. پیشبینی در بازار بورس یا بازار سرمایه همواره مورد توجه مطالعات بوده است. این توجه زیاد در سالهای اخیر منجر به پیشرفت الگوهای مورد استفاده در پیشبینی شده است. عموماً روش شبکه عصبی مصنوعی درمقایسه با روشهای سری رگرسیونی عملکرد

بهتری نشان داده است. در همـین راسـتا این مطالعه با هدف پیشبینی برخی از قیمت سهام شرکت گوگل در بازار بورس آمریکا با استفاده ازالگوریتم های زمانی و شبکه های عصبی عمیق صورت گرفته است.

۱-۳ ضرورت انجام تحقیق

روشهای متداول آماری و احتمالی بر پایه روابط و فرمول های صرفاً ریاضی که به طور اخص به پیشبینی سری های زمانی میپردازد، از دیرباز مورد توجه متخصصین علوم مالی قرار گرفته است. آنها با دستمایه قراردادن این بخش از علم آمار به تحلیل، بررسی و شناخت رفتار سرمایه گذاران و بازارهای مالی می پرداختند. در تحقیقات اقتصادی و مالی به منظور تخمین و پیشبینی بیشتر از مدل های اقتصاد سنجی سری زمانی استفاده می شود . در مدلهای سری زمانی صرفا از رفتار خود متغیر برای پیش بینی استفاده میکنند. با توجه به این نکته که روند پدیده ها و متغیرهای وابسته ی دیگر نیز در طول زمان در نوسان می باشد؛ میتوان برای مدلسازی دقیقتر آنها از از مدل -های غیرخطی استفاده کرد. شناخته شده ترین و پرکاربردترین مدل در میان مدل های غیر خطی مدل شبکه های عصبی مصنوعی می باشد.

شبکه های عصبی را میتوان با اغماض زیاد، مدلهای الکترونیکی از ساختار عصبی مغز انسان نامید . مکانیسم فراگیری و آموزش مغز اساسا بر تجربه استوار است. مدلهای الکترونیکی شبکه های عصبی طبیعی نیز بر اساس همین الگو بنا شده اند. به طور کلی شبکه های عصبی زیستی از تعداد زیادی نرون و ارتباطات بین آنها تشکیل شده اند و با یک گراف همراه با گره ها و یال هایش مدل سازی می شوند. این گراف به همراه الگوریتم آموزشی آن شبکه عصبی را تبدیل به یکی از قدرتمندترین ابزارها در مدل سازی پدیده های مختلف کرده است. در واقع ماهیت و ذات تجربی و منعطف این روش باعث میشود تا در مسائلی مانند مقوله پیشبینی مقوله ای که پیچیدگی در ساختار آنها مشاهده میشود و از رفتاری غیرخطی و لجامگسیخته برخوردار هستند، به خوبی قابل استفاده باشد.

رویکرد عمده برای پیش بینی قیمت سهام، روشهای آماری مبتنی برسریهای زمانی مانند: میانگین متحرک ،هموارسازی، آریما، رگرسیون و یا ترکیبی از آنها میباشد. این روشها بنا به ماهیت خود، در محیط های با تغییرات کم با تقریب خوبی قادر به پیشبینی هستند اما در مواردی چون پیش بینی بازار سهام که شرایط محیطی همواره در حال تغییر است ممکن است نتوانند تقریب خوبی از تغییرات محیطی را تخمین بزنند. به این ترتیب نیاز به استفاده از ابزارها و مدل های نوین جهت پیش بینی این دوره ها ضروری میگردد. امروزه بهره برداری از سیستمهای هوشمند و روشهایی مانند شبکه های عصبی، الگوریتم ژنتیک، منطق فازی و ... در حوزه های مختلف علوم کاربرد فراوانی یافته است. اخیرا شبکه های عصبی به علت ماهیت غیرخطی، در پیش بینی قیمت سهام جذابیت فراوانی در بازار بورس سهام یافته است. اما به علت گستردگی روشها و ساختارهای شبکه های عصبی که از ماهیت پویای آن سرچشمه میگیرد و همچنین روند رو به گسترش این شاخه از علم، نیاز به یافتن یک روش کارآمد برای پیش بینی قیمت سهام در بازار سرمایه احساس میشود. تاثیرپذیری از عوامل مختلف چه به صورت مستقیم چه غیرمستقیم از تحوالت اقتصادی و اجتماعی که تعداد آنان در دهه اخیر کم نبوده باعث تحوالت و چرخه هایی در روند قیمت سهام در بورس اوراق بهادار شده است. بنابراین اهمیت استفاده از یک روش پویا و مناسب با شرایط حاکم که دقت پیش بینی بیشتری دارند را آشکار میکند.

منطق مقایسه ی مدلهای پیشبینی در وهله ی اول استفاده از تصریح و شیوه ی برآورد دقیق و در وهله ی دوم دستیابی به مدلی است که نتایج دقیق را در پیشبینی ها حاصل نماید.

۱-٤ سوالات تحقیق

سری زمانی چیست؟

مولفه های سری زمانی چیست؟

ایستایی در سری زمانی به چه معنی است؟

روش های ایستا کردن سری زمانی به چه صورت است؟

انواع الگوریتم های سری زمانی به صورت است؟

شبکه ی عصبی چیست؟

انواع شبکه های عصبی عمیق به چه صورت است؟

پیش بینی با استفاده از شبکه ی عصبی به چه صورت انجام می شود؟

دقت مدل های سری زمانی بیشتر است یا شبکه های عصبی ؟

۱-۵هدف از انجام پژوهش

پیش بینی قیمت سهام به وسیله ی الگوریتم های سری زمانی و شبکه های عصبی عمیق ، مقایسه ی دقت الگوریتم ها و انتخاب بهترین الگوریتم.

۱-٦ نتایج مورد انتظارپس از انجام تحقیق به دست آوردن دید کلی از پیشبینی قیمت سهام به وسیله ی الگوریتم های مختلف و شناخت ساختار این الگوریتم ها.

۱-۷ روش تحقیق

این تحقیق توصیفی و روش به کاررفته در آن پیمایشی است .داده از سایت kaggle دریافت شده و مدل سازی ها نیز طبق مقالات دیگر محققان در این زمینه با استفاده از نرم افزار python توسط بنده صورت گرفته است . همچنین از مقالات و گزارشات دیگر محققان برای توصیف مباحث تئوری استفاده شده است.

ده ردیف اول دیتای ما به صورت زیر است و ما قصد داریم که ستون Open را برای شصت روز آخر این دیتا پیش بینی کنیم و دقت الگوریتم های مختلف را باهم مقایسه کنیم .

1	Date	Close/Las	Volume	Open	High	Low
2	2020-12-31	$132.69	99116590	134.08	$134.74	$131.72
3	2020-12-30	$133.72	96452120	135.58	$135.99	$133.40
4	2020-12-29	$134.87	1.21E+08	138.05	$138.79	$134.34
5	2020-12-28	$136.69	1.24E+08	133.99	$137.34	$133.51
6	2020-12-24	$131.97	54930060	131.32	$133.46	$131.10
7	2020-12-23	$130.96	88223690	132.16	$132.43	$130.78
8	2020-12-22	$131.88	1.69E+08	131.61	$134.41	$129.65
9	2020-12-21	$128.23	1.21E+08	125.02	$128.31	$123.45
10	2020-12-18	$126.66	1.93E+08	128.96	$129.10	$126.12

شکل ۱_ ۱:ده ردیف اول دیتا

۱-۸ روش و ابزار گردآوری اطلاعات

برای جمع آوری اطلاعات از روش کتابخانه ای و مطالعه اسناد و مدارک افراد و تحقیقات و بررسی های دیگر انجام گرفته در حوزه مورد مطالعه، استفاده شده است.

"فصل دوم"

"مبانی نظری سری های زمانی و بررسی الگوریتم های آن"

۲-۱ پیش نیاز های آماری

برای داشتن پیش زمینه جهت درک مطالب عنوان شده در این مطلب و همچنین مروری بر بعضی مفاهیم آماری لازم یک سری مطالب را به طور خلاصه بیان کنیم .

۲-۱-۱کوواریانس(Covariance)

یکی از شاخص‌های مهم وابستگی بین دو متغیر تصادفی (Random Variable) در آمار، کوواریانس (Covariance)است. این مفهوم به شکلی با پراکندگی و معیار واریانس (Variance) ارتباط دارد. البته واریانس مربوط به یک متغیر است در حالیکه محاسبه کوواریانس ارتباط بین دو متغیر را بوسیله پراکندگی‌هایشان نسبت به میانگین، نشان می‌دهد. هر چه مقدار کوواریانس بین دو متغیر، بزرگتر باشد، میزان وابستگی بین آن‌ها بیشتر است و برعکس اگر میزان کوواریانس بین دو متغیر کم باشد، وابستگی خطی بین آن‌ها کم خواهد بود .هر چه مقدار کوواریانس به صفر نزدیکتر باشد، میزان وابستگی خطی بین آن‌ها کمتر خواهد بود. مقادیر مثبت نشانگر رابطه هم جهت بین دو متغیر و مقادیر منفی کوواریانس نیز بیانگر رابطه معکوس بین آن‌ها خواهد بود.

$$\mathrm{Cov}(x, y)= \frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{n-1}$$

۲-۱-۲همبستگی (Correlation)

کوواریانس به واحد اندازه‌گیری داده‌ها بستگی دارد. در نتیجه نمی‌توان بزرگی کوواریانس دو متغیر را با بزرگی کوواریانس دو متغیر دیگر بدون در نظر گرفتن واحد اندازه‌گیریشان، مقایسه کرد. ضریب همبستگی که شاخصی بدون واحد است، این مشکل را حل کرده. با توجه به نوع داده‌ها، شیوه‌های مختلفی برای

اندازه‌گیری ضریب همبستگی وجود دارد. اغلب ضریب همبستگی، رابطه بین مقدارهای میانگین دو متغیر را نشان می‌دهد. ضریب همبستگی را با ρو یا r نشان می‌دهند. در این متن به بررسی و شیوه محاسبه ضریب همبستگی پیرسون (Peasron Correlation Coefficient) ، می‌پردازیم.

$$\text{Cor}(x, y)= \frac{1}{(n-1)} \frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sigma_x * \sigma_y}$$

۲-۲ تعریف سری زمانی

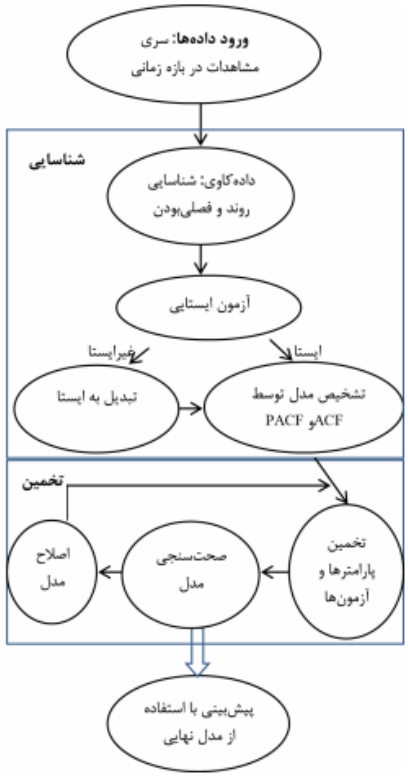
یک سری زمانی مجموعه ای از مشاهدات درباره ی یک متغیر است که در نقاط گسسته ای از زمان که معمولا فاصله های مساوی دارند ؛ اندازه گیری شده و بر حسب زمان مرتب شده اند . بنابراین یک سری زمانی از مشاهدات یک پدیده در طول زمان به دست می آید.

۲-۳ چرا سری های زمانی

تفاوت سری های زمانی و سایر روش های مدل سازی از جمله رگرسیون :

سری های زمانی با استفاده از داده های قبلی مقادیر آینده را پیش بینی می کنند ؛ در حالی که در دیگر روش های مدل سازی اغلب با استفاده از متغیر های مستقل دیگر سعی در پیش بینی متغیر مورد نظر داریم.معمولا قدرت سری های زمانی در پیش بینی کمتر است ولی به دلیل این که اطلاعات جانبی کمتری نیاز دارد تمایل به استفاده از آن زیاد می باشد .

۲-٤ مراحل پیش بینی با استفاده از مدل های سری زمانی



شکل ۲_ ۱ : دیاگرام عملکردی استراتژی مدلسازی به روش باکس-جنکینز(آریما)

همان طور که در شکل بالا مشاهده می کنید قدم اول بررسی داده و تشخیص مولفه های آن مانند روند ،تناوب ،فصلی بودن و... می باشد که در گام بعد به بررسی آن ها می پردازیم.

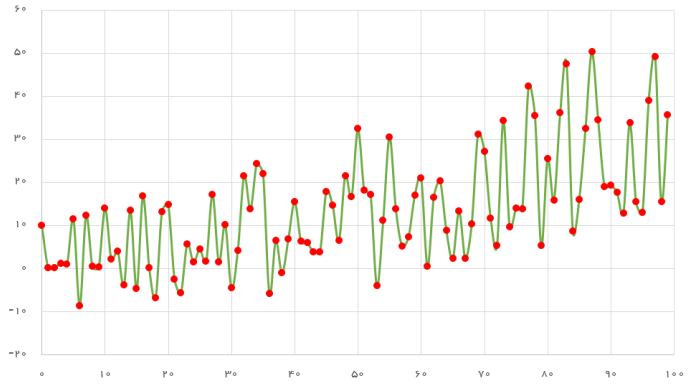
۲-٤-۱مولفه های یک سری زمانی

معمولا می‌توان الگوی رفتار یا مدل تغییرات یک سری زمانی را به چهار مولفه تفکیک کرد. روند (Trend)، تناوب(Cyclic) ، فصل (Seasonal) و تغییرات نامعمول .(Irregular) اگر نمودار مربوط به داده‌های سری زمانی را برحسب زمان ترسیم کنیم می‌توانیم این مولفه‌ها را تشخیص دهیم در نتیجه شناخت بهتری از داده‌های سری زمانی خواهیم داشت. در ادامه به معرفی و بررسی هر یک از این مولفه‌ها می‌پردازیم.

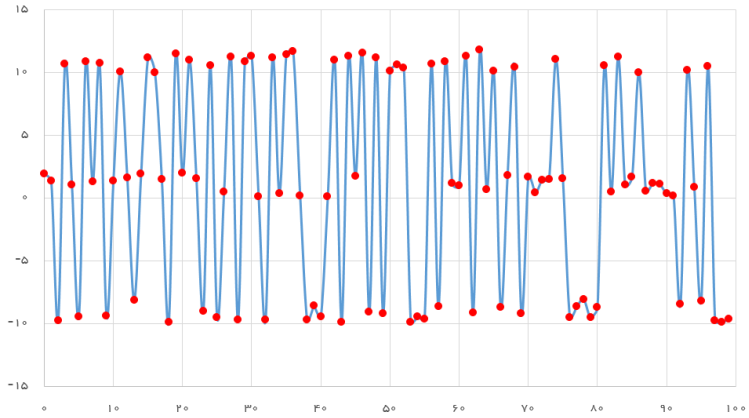
۲-٤-۱-۱ روند

تمایل سری زمانی به افزایش، کاهش یا حتی ثابت بودن، روند را تشکیل می‌دهد. در یک سری زمانی با روند افزایشی، انتظار داریم مقدارهای سری زمانی در زمان‌های $t=1$ و $t=2$ به صورت $X_{(1)} \leq X_{(2)}$ باشند. برای مثال روند برای سری زمانی مربوط به میزان جمعیت یا سرمایه در بازار بورس به صورت افزایشی، ولی روند برای میزان مرگ و میر با توجه به پیشرفت در امور پزشکی، کاهشی است.

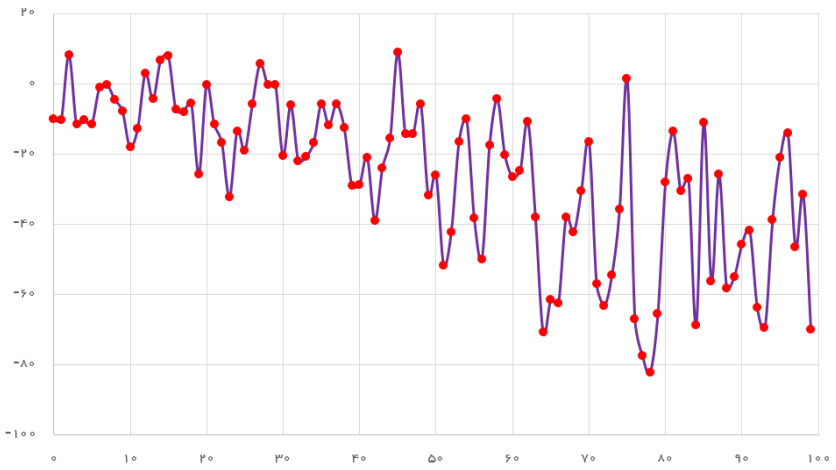
در تصویرهای زیر سه نمودار مربوط به سه سری زمانی در ۱۰۰ زمان مختلف با روندهای افزایشی، ثابت و کاهشی نشان داده شده است.



شکل ۲_ ۲ : سری زمانی با روند افزایشی



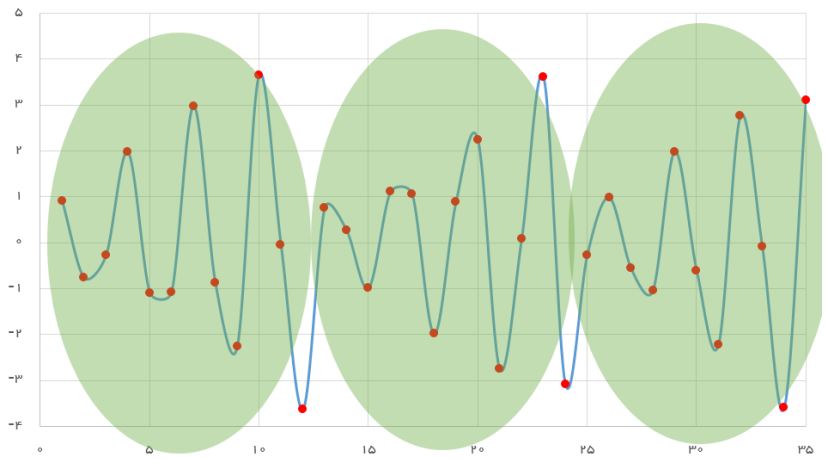
شکل ۲_ ۳: سری زمانی با روند ثابت



شکل ۲_ ۴: سری زمانی با روند کاهشی

۲-۱-٤-۲ فصل (*Seasonal*)

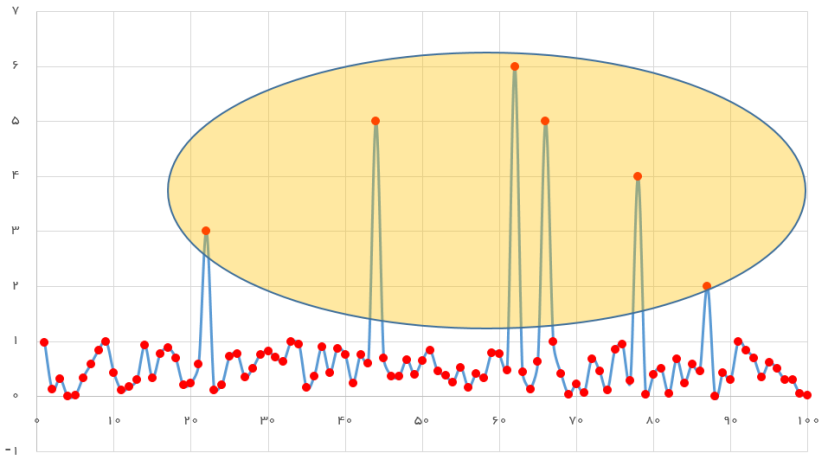
در سری زمانی، تغییراتی که در دوره‌ای کوتاه‌تر از یک تناوب به صورت تکراری رخ می‌دهد، به تغییرات فصلی معروف است. برای مثال در طول یک سال میزان فروش لباس‌های گرم در زمستان افزایش داشته و سپس در فصل‌های دیگر کاهش داشته است. این تناوب در سال‌های بعد نیز به همین شکل تکرار می‌شود. همانطور که مشخص است دوره تکرار تغییرات فصلی کوتاه‌تر از دوره تکرار برای تغییرات تناوبی است.



شکل ۲_ ۵: سری زمانی فصلی

۲-٤-۱-۳ تغییرات نامعمول

این گونه تغییرات بر اثر عوامل تصادفی و غیرقابل پیش‌بینی ایجاد می‌شوند. برای مثال زلزله یا سیل در بررسی رشد جمعیت ممکن است اثرات بزرگی داشته باشد. این مولفه بعد از شناسایی توسط نمودار ترسیم شده از سری زمانی باید حذف شود. در غیر اینصورت نتایج حاصل از تحلیل سری زمانی ممکن است گمراه کننده باشند.



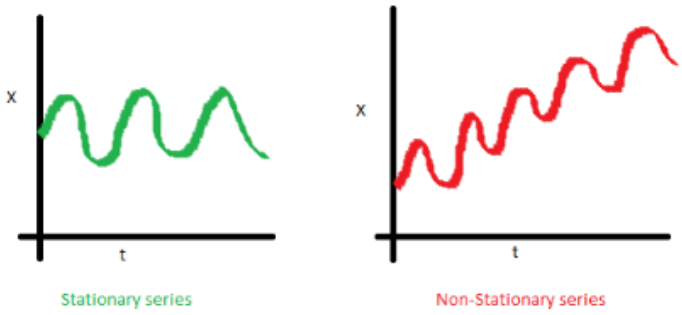
شکل ۲_۶: سری زمانی دارای تغییرات نامعمول

۲-٤-۲ سری زمانی ایستا و نا ایستا (Stationary time series)

به تصویرهای زیر توجه کنید. در همه آن‌ها محور افقی بیانگر زمان و محور عمودی متغیری است که وابسته به زمان تغییر می‌کند. این متغیر می‌تواند میزان بارش، حرارت، درآمد و غیره باشد. مطمئن هستیم که تغییرات این متغیر وابسته به زمان است و قرار است مقدار آن را برای زمان آینده پیش‌بینی کنیم. سری زمانی که با رنگ سبز نشان داده است، سری ایستا و در مقابل سری زمانی قرمز رنگ، بیانگر سری زمانی ناایستا است. در ادامه به بررسی عواملی می‌پردازیم که یک سری زمانی را ناایستا می‌کنند.

۲-٤-۲-۱ روند(*Trend*)

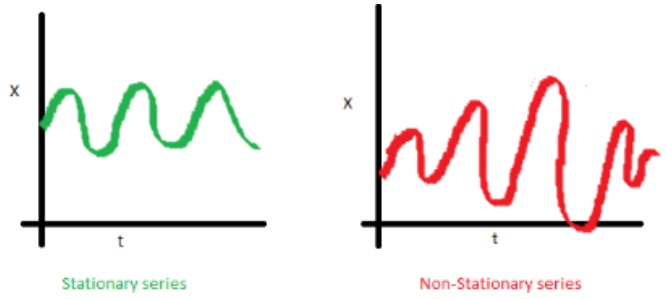
اگر میانگین سری زمانی وابسته به زمان نباشد، آن را سری زمانی ایستا (Stationary Time Series) می‌نامیم. در این صورت تغییرات میانگین سری زمانی برحسب زمان باعث ناایستایی سری زمانی (Non-Stationary)خواهد شد. تغییرات میانگین در طول دوره یا بازه زمانی سری را روند (Trend) می‌نامند. ممکن است الگوی تغییرات به صورت صعودی یا نزولی باشد.



شکل ۲_۷: سری زمانی ایستا و نا ایستا بر اساس میانگین

۲-٤-۲-۲ ثابت بودن واریانس

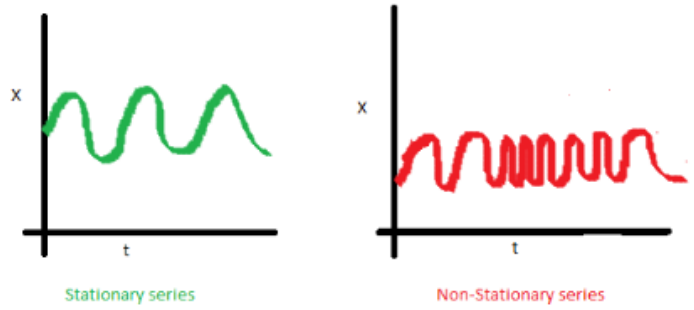
اگر واریانس پدیده سری زمانی در طول زمان ثابت نباشد، باز هم سری را ناایستا می‌نامند. در سری زمانی ایستا، پراکندگی یا واریانس نباید تابعی از زمان محسوب شود. این خاصیت را گاهی یکنواختی (Homoscedasticity)می‌نامند. در تصویر زیر به خوبی نابرابری واریانس در بازه‌های مختلف زمانی در نمودار قرمز رنگ دیده می‌شود. در حالیکه در نمودار سبز رنگ، میزان تغییرات یکسان به نظر می‌رسد. توجه داشته باشید که منظور از واریانس میانگین مربعات نوسانات نسبت به خط مرکزی روی محور عمودی است.



شکل ۲_۸: سری زمانی ایستا و نا ایستا بر اساس ثابت بودن واریانس

۲-٤-۲-۳ یکنواختی در واحد های زمانی

برای اینکه نشان دهیم میزان تغییرات سری زمانی در طول‌های مشخصی از زمان نیز مستقل از زمان است، از مفهوم کوواریانس کمک می‌گیریم. در یک سری زمانی ایستا، در بازه‌های زمانی به طول m یعنی فاصله زمانی i تا i+m داده‌های سری نباید وابسته به زمان باشند. در تصویر زیر، نمودار قرمز رنگ، نشانگر عدم چنین خاصیتی است، پس سری زمانی با توجه به اینکه دارای واریانس و میانگین ثابتی است، باز هم سری زمانی ایستا نخواهد بود.



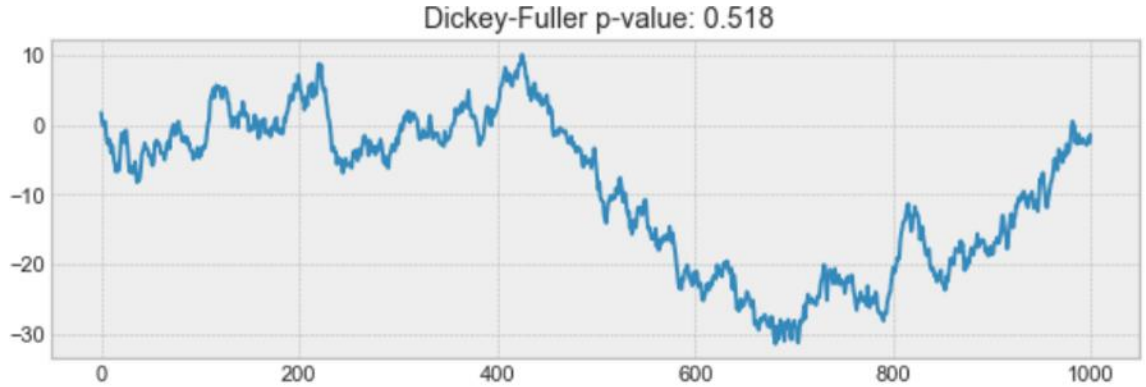
شکل ۲_۹: سری زمانی ایستا و نا ایستا بر اساس در واحد های زمانی

علت اصلی تحلیل سری زمانی روی داده‌های زمانی ایستا، سادگی انجام محاسبات است. از طرفی وجود چنین خاصیتی امکان برآورد شاخص‌های دیگر مانند میانگین و واریانس و...را فراهم می‌آورد. به علاوه روش‌های دقیق و مناسبی برای ایستا کردن سری‌های زمانی ناایستا نیز وجود دارد. بنابراین با توجه به این

موضوعات ابتدا روش‌های تحلیل سری زمانی ایستا را مورد بررسی قرار داده، سپس به سری‌های زمانی ناایستا خواهیم پرداخت

۲-٤-۳ بررسی ایستایی سری های زمانی (آزمون دیکی فولر)

چنده راه برای بررسی ایستایی یک سری زمانی وجود دارد که ما مطمئن ترین آن ها که استفاده از یک آزمون فرض آماری به نام آزمون Dickey-Fuller است را توضیح می دهیم و در پژوهش خود از آن استفاده می کینم . به کمک این آزمون آماری می‌توان ایستایی یک سری زمانی را مورد بررسی قرار داد. اگر به زبان آزمون فرض آماری در مورد این آماره صحبت کنیم، می‌توان گفت که فرض صفر در این آزمون ناایستا بودن سری زمانی است. در نتیجه اگر با توجه به مقدار p-Value یا همان مقدار احتمال، فرض صفر رد شود، رای به ایستا بودن سری زمانی خواهیم داد.با توجه به مقدار احتمال (p) می‌توان در مورد ایستایی سری زمانی تصمیم گرفت. اگر p کوچکتر از احتمال خطای نوع اول (α) باشد، فرض صفر رد شده و رای به ایستایی سری زمانی خواهیم داد. در غیر اینصورت دلیلی برای وجود ایستایی سری زمانی وجود ندارد. اطلاع دارید که معمولا مقدار αرا برابر با ۰.۰۵یا ۵٪ در نظر می‌گیرند. به تصویر زیر نگاه کنید. با توجه به شکل تغییرات سری زمانی و مقدار احتمال p −Value=0.518متوجه می‌شویم که سری زمانی ایستا نیست.



شکل ۲_ ۱۰: آزمون دیکی فولر

در اینجا از نحوه محاسبه شاخص Dickey-Fuller صرف نظر کرده‌ایم زیرا در بیشتر نرم‌افزارهای تحلیل سری زمانی محاسبات مربوط به آن قابل انجام است. از طرفی تفسیری که از آن بدست می‌آید مهم است تا بتوان ایستایی سری زمانی را تشخیص داد.

زمانی که سری زمانی ایستا باشد، قادر هستیم براساس مدل‌های مرتبط با سری زمانی، رفتار فرآیند را برحسب زمان توضیح دهیم. در غیر اینصورت تا زمانی که سری زمانی ایستا نشود، امکان استفاده از مدل‌های سری زمانی معمول وجود ندارد.

۲-٤-٤ ایستا کردن سری زمانی

فهمیدیم که برای استفاده از مدل های سری زمانی حتما باید سری زمانی ایستا باشد و در این قسمت بررسی می کنیم که اگر یک سری زمانی ایستا نبود چگونه با آن رفتار کنیم و چگونه می توان آن را ایستا کرد.

به طور کلی از دوراه برای ایستا کردن یک سری زمانی نا ایستا وجود دارد ،تفاضل گیری و هموار سازی که به بررسی آن ها می پردازیم.

۲-٤-٤-۱هموار سازی با استفاده از میانگین متحرک(*Moving Average*)

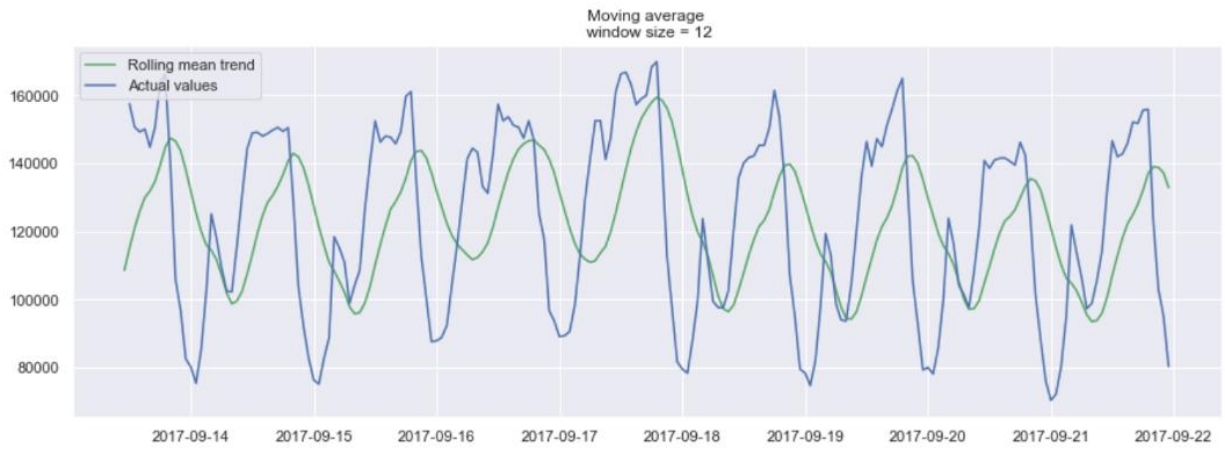
به کمک میانگین متحرک می‌توان روند یک سری زمانی را تشخیص داد. با تعریف یک پنجره (Window) از شیوه میانگین متحرک برای هموارسازی سری زمانی و تغیین نقاط تغییر روند می‌توان استفاده کرد.

نکته :منظور از پنجره در سری زمانی، بازه‌ای از سری زمانی است که به منظور تحلیل دقیق‌تر رفتار سری به کار گرفته می‌شود. در تصویر زیر با استفاده از تکنیک میانگین متحرک، روند تغییرات میانگین (Rolling mean Trend)نمایش داده شده است. اندازه پنجره در این نمودار برابر با ٢٤ است. به این ترتیب برای

بازه‌های ۲٤ ساعتی میانگین محاسبه شده است. برای مثال فاصله میانگین مصرف انرژی در بازه ساعت ۱۵ یک روز تا ساعت ۱۵ روز دیگر ملاک تعیین یک نقطه از خط سبز رنگ شده است. به همین ترتیب در کل یک روز از میانگین‌های مصرف در بازه‌های ۲٤ ساعتی استفاده شده است.

شکل ۲_ ۱۱: هوار سازی سری زمانی با استفاده از میانگین متحرک با۲۴ =lag

همانطور که دیده می‌شود، خطوط آبی رنگ مقدارهای واقعی و خطوط سبز رنگ میانگین متحرک با پنجره یا تاخیر (Lag) برابر با ۲٤ را نشان می‌دهند. به نظر می‌رسد تا تاریخ ۲۰۱۷-۰۹-۲۹روند صعودی و از آن به بعد میانگین میزان مصرف انرژی، نزولی است. از طرفی خط سبز رنگ در نمودار، نمایش هموار شده (Smoothed)سری زمانی با اندازه پنجره ۲٤ است. با تمرکز روی خطوط آبی نمودار که مربوط به سری زمانی است، مشخص است که در هر ۲٤ ساعت دو نقطه به عنوان قله (Peak) مصرف دیده می‌شوند. هر چه اندازه پنجره را کوچکتر انتخاب کنیم، میزان هموار سازی در مدل میانگین متحرک کاهش خواهد یافت. در تصویر زیر با اندازه پنجره ۱۲، تغییرات خط روند که بوسیله میانگین متحرک تولید شده بیشتر از حالت قبل با اندازه پنجره ۲٤ است. به این ترتیب هموارسازی کمتر صورت گرفته و به نظر می‌رسد سری زمانی ایستا نشده است.



شکل ۲_ ۱۲: هوار سازی سری زمانی با استفاده از میانگین متحرک با ۱۲ =lag

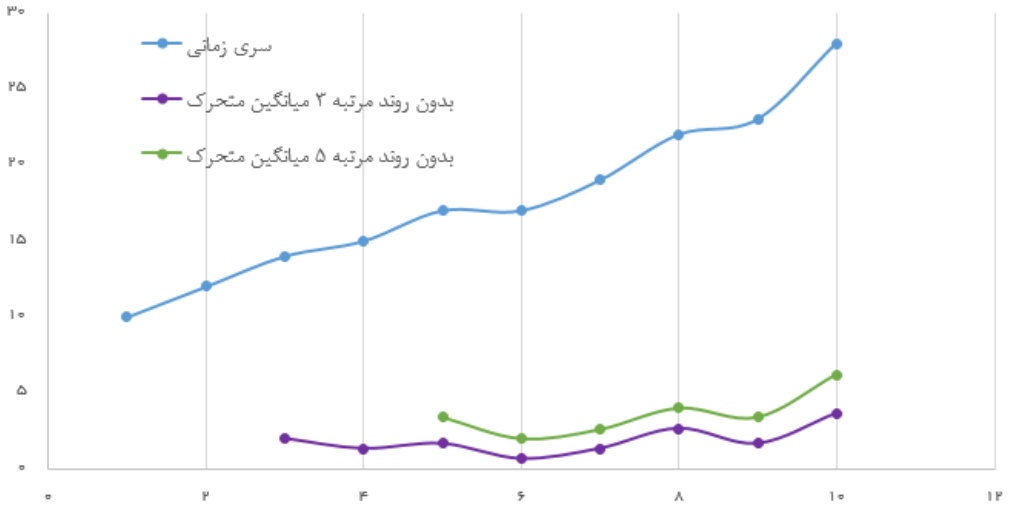
براز مثال فرض کنید یک سری زمانی طبق جدول زیر برای ۱۰ زمان مختلف ثبت شده است. اگر درجه هموار سازی میانگین متحرک را برابر با ۳ در نظر بگیریم، باید میانگین مقدار جاری و دو مقدار قبلی (مجموعه سه مقدار) را بدست آوریم و جایگزین مقدار جاری کنیم. واضح است که برای دو مقدار اول در سری زمانی، این کار امکان پذیر نیست زیرا نمی‌توان سه مقدار برای محاسبه میانگین در نظر گرفت. در نتیجه این گونه هموار سازی به کاهش مجموعه داده منجر می‌شود. هر چه درجه همواره سازی را بزرگتر انتخاب کنید، هموارسازی زودتر انجام خواهد شد ولی در عوض ممکن است خطای پیش‌بینی را افزایش دهد. حال برای آنکه سری زمانی، خاصیت ایستایی پیدا کند و روند را از آن حذف کنیم، کافی است مقدارهای سری زمانی را از میانگین متحرک کم کنیم، سری زمانی جدید بدون روند خواهد بود. پس اگر $Y(t)$سری زمانی باشد و $m(t)$ میانگین متحرک در زمان t در نظر گرفته شود، سری زمانی ایستای $X(t)$ به صورت زیر مورد محاسبه قرار می‌گیرد.

$$x(t) = y(t) - m(t)$$

سری زمانی بدون روند با میانگین متحرک مرتبه ۵	همواره سازی با مرتبه ۵	سری زمانی بدون روند با میانگین متحرک مرتبه ۳	همواره سازی بامرتبه ۳	مقدار سری زمانی	زمان
–	–	–	–	۱۰	۰
–	–	–	–	۱۲	۱
–	–	۲	۱۲	۱۴	۲
–	–	۱.۳۳	۱۳.۶۷	۱۵	۳
۳.۴	۱۳.۶	۱.۶۷	۱۵.۳۳	۱۷	۴
۲	۱۵	۱.۶۷	۱۶.۳۳	۱۷	۵
۲.۶	۱۶.۴	۱.۳۳	۱۷.۶۷	۱۹	۶
۴	۱۸	۲.۶۷	۱۹.۳۳	۲۲	۷
۳.۴	۱۹.۶	۱.۶۷	۲۱.۳۳	۲۳	۸
۶.۲	۲۱.۸	۳.۶۷	۲۴.۳۳	۲۸	۹

جدول ۲- ۱: تفاضل گیری سری زمانی با دو مرتبه ی ۳ و ۵

به تصویر زیر دقت کنید. نمودار سری زمانی ناایستا به رنگ آبی و سری‌های زمانی ایستای حاصل از عملگر میانگین متحرک مرتبه **۳** و **۵** به رنگ‌های بنفش و سبز نمایش داده شده‌اند. کاملا مشخص است که در سری‌های زمانی ایستای تولید شده، مولفه روند وجود ندارد.



شکل ۲_۱۳: هموار سازی با استفاده از میانگین متحرک مرتبه ۳ و ۵

۲-٤-٤-۲ هوار سازی نمایی (*Exponential Smoothing*):

هموارسازی نمایی نیز از همان منطق هموار سازی میانگین متحرک پیروی می‌کند. به این ترتیب می‌توان این روش را به صورت میانگین وزنی برای داده‌های سری زمانی در نظر گرفت که به داده‌های دورتر وزن کمتری در محاسبه میانگین می‌دهد. کاهش وزن مقدارهای گذشته دور در حقیقت از اهمیتشان در محاسبه و پیش‌بینی مقادیر مربوط به آینده می‌کاهد و داده‌های مربوط به حال حاضر تاثیر بیشتری خواهند داشت.

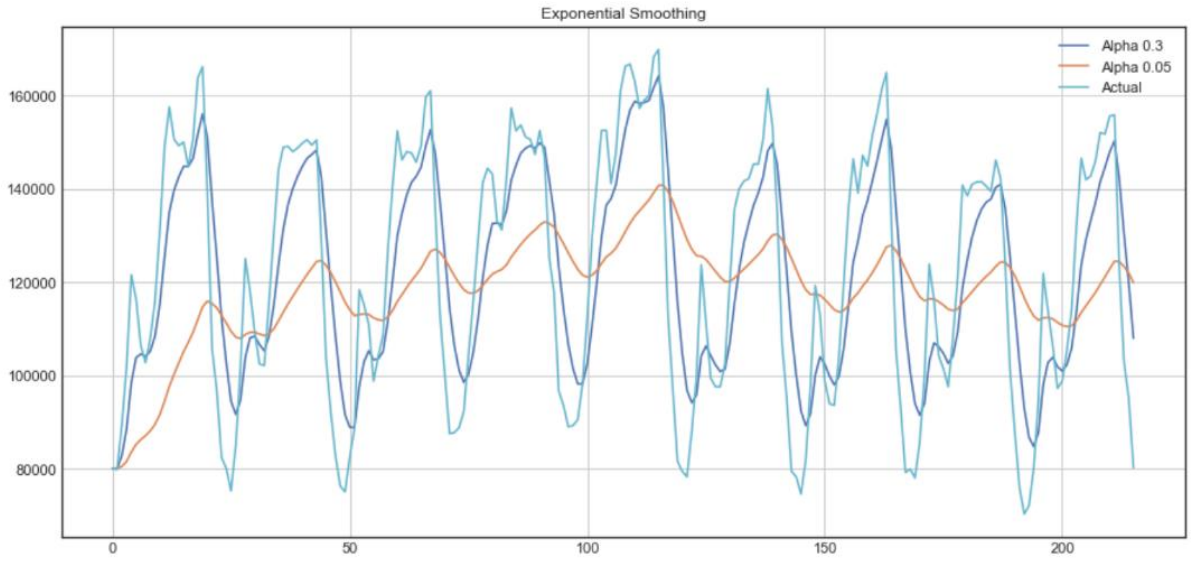
از لحاظ ریاضی، مدل هموارسازی نمایی را می‌توان به صورت زیر نمایش داد:

$$\mathrm{yt} = \alpha \mathrm{xt} + (۱ - \alpha)\mathrm{yt}$$

در اینجا منظور از xt مقدار واقعی سری در زمان t است و yt نیز بیانگر مقدار پیش‌بینی شده سری در زمان t توسط هموارسازی نمایی است. به این ترتیب میانگین وزنی بین مقدار واقعی و پیش‌بینی در زمان t برای زمان t+1 سری زمانی در نظر گرفته می‌شود.

مشخص است که در این رابطه α عامل یا فاکتور هموارسازی است. مقدار این عامل در بازه بین ۰ و ۱ قرار دارد. از طرفی با افزایش میزان این فاکتور نقش مشاهدات گذشته در محاسبه و پیش‌بینی مقدارهای آینده کاهش می‌یابد. در حقیقت فاکتور α نرخ تاثیر گذاری مقدارهای گذشته را تعیین می‌کند.

در نمودار زیر هموارسازی نمایی با استفاده از فاکتور $\alpha=0.3$ و $\alpha=0.05$ نمایش داده شده است. همانطور که مشخص است زمانی که مقدار α کم است، هموارسازی بیشتر صورت گرفته است. با انتخاب $\alpha=0$ در حقیقت هموارسازی به صورت میانگین متحرک در خواهد آمد.



شکل ۲_۱۴:هموار سازی سری با استفاده از هموار سازی نمایی

خط سبز در این نمودار مقدارهای واقعی سری زمانی است در حالیکه خط آبی و نارنجی به ترتیب با فاکتور ۰.۳ و ۰.۰۵ هموارسازی را انجام داده‌اند. استفاده از هموارسازی نمایی برای سری زمانی بدون روند مناسب به نظر می‌رسد. وجود تناوب در این حالت مشکلی ایجاد نمی‌کند.

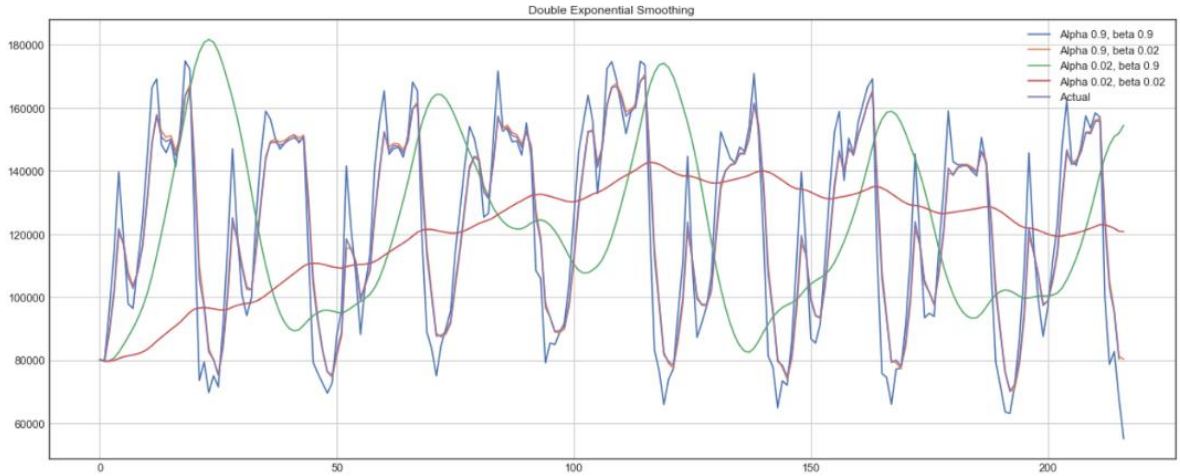
۲-٤-٤-۳ هموارسازی نمایی مضاعف (*Double Exponential Smoothing*)

اگر سری زمانی دارای روند باشند، استفاده از روش هموارسازی نمایی مضاعف نتایج بهتری را ارائه خواهد داد. به نظر می‌رسد که می‌توان هموارسازی نمایی مضاعف را به صورت دو بار استفاده از هموارسازی نمایی ساده در نظر گرفت. شکل رابطه ریاضی آن در زیر قابل مشاهده است.

$$yt = \alpha xt + (۱ - \alpha)yt\,(yt - ۱ + bt - ۱)$$

$$bt = \beta(yt - yt - 1) + (۱ - \beta)bt - ۱$$

در این رابطه، منظور از βفاکتور هموارسازی روند است که می‌تواند مانند αمقداری بین ۰ تا ۱ داشته باشد. در نمودار زیر با توجه به مقدارهای مختلف برای Alpha و Beta عمل هموارسازی صورت گرفته است.



شکل ۲ _ ۱۵:نمودار هموارسازی شده نمایی

همانطور که مشخص است با انتخاب مقدارهای کوچک برای Alpha وBeta ، هموارسازی به خوبی صورت گرفته است ولی در مقابل با انتخاب Alpha=0.9 و beta=0.02 بهترین پیش‌بینی‌ها حاصل

شده است. مشخص است با این فاکتورها پیش‌بینی مقدارهای مشاهده شده با مقدار واقعی فاصله کمی داشته ولی امکان پیش‌بینی برای زمان‌های آینده به خوبی صورت نخواهد گرفت. بنابراین استفاده از محاسبات مربوط به خط روند، ابزار بهتری برای پیش‌بینی آینده محسوب می‌شود. استفاده از هموارسازی نمایی مضاعف برای سری زمانی بدون تغییرات فصلی مناسب به نظر می‌رسد. وجود تناوب و روند در این حالت مشکلی ایجاد نمی‌کند.

۲-٤-٥ تفاضل گیری (Difference)

تفاضل گیری روشی برای تبدیل داده‌های سری زمانی نا ایستا به ایستا می باشد. می‌توان از آن برای حذف وابستگی سری به زمان استفاده کرد . این تفاوت می‌تواند به تثبیت میانگین سری‌های زمانی با حذف تغییرات در سطح یک سری زمانی , و بنابراین حذف یا کاهش روند و فصلی بودن آن کمک کند. فرمول آن به صورت زیر است:

$$d^{(1)}(t) = x(t) - x(t-1);$$

اگر با یک مرتبه تفاضل گیری موفق به ایستا کردن سری زمانی نشدیم یک بار دیگر آن را تکرار می کنیم که به این کار تفاضل گیری مرتبه دوم گفته می شود .

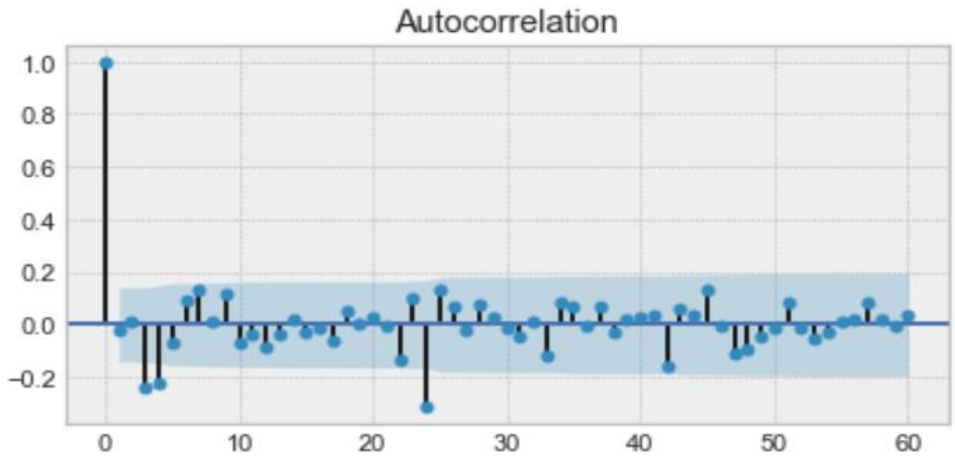
$$d^{(2)}(t) = d^{(1)}(t) - d^{(1)}(t-1).$$

حال با توجه مفهوم ایستایی سری زمانی، به معرفی تابع خودهمبستگی می‌پردازیم. از آنجایی که در محاسبه تابع خودهمبستگی از کوواریانس و همچنین واریانس سری زمانی استفاده می‌شود، اگر سری ایستا نباشد، امکان محاسبه تابع خودهمبستگی وجود نخواهد داشت.

ویژگی مهم داده های سری زمانی همبستگی بین داده ها است . لذا تحلیل اساسی در سری زمانی بررسی وابستگی داده ها به همدیگر است . بدین منظور دو تابع تعریف شده است به نام های تابع خود همبستگی (Auto Correlation Function)و تابع خود همبستگی جزئی (Partial Auto Correlation Function) .

۲-٤-٦تابع خود همبستگی (AFC)

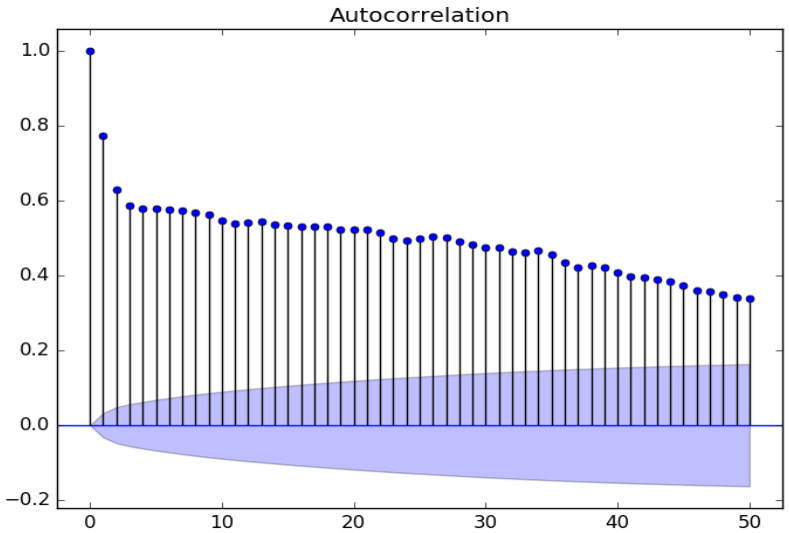
در بحث مربوط به سری زمانی، بین مشاهدات براساس زمان وابستگی وجود دارد. از آنجایی که در آمار، وابستگی را اغلب به صورت همبستگی (Correlation) بیان می‌کنند، واژه خود همبستگی (Autocorrelation)به معنی همبستگی سریالی (Serial Correlation) یا وابستگی بین مقدار دنباله‌ای برحسب زمان است. تابعی که خودهمبستگی را برحسب یک فاصله زمانی بین مشاهدات محاسبه می‌کند تابع خودهمبستگی (Autocorrelation Function) نامیده می‌شود. در بحث سری زمانی (Time Series)از «تابع خودهمبستگی» استفاده می‌کنند تا به رفتار فرآیند برحسب زمان یا مکان پی‌ببرند. در تصویر زیر نمونه‌ای از نمودار تابع خودهمبستگی را مشاهده می‌کنید.



شکل ۲_۱۶: تابع خود همبستگی

برای برآوردو تخمین پارامتر مدل میانگین متحرک(q) که جلو به آن می پردازیم از تابع و نمودار خود همبستگی استفاده می‌شود. این پارامتر بزرگترین تاخیری (lag)را نشان می‌دهد که از آن به بعد میزان خود همبستگی به صفر نزدیک می‌شود. در تصویر زیر این مقدار برابر با ٤ است. تاخیر ها از صفر شروع می شوند.

اگر نمودار ACF به صورت تدریجی نزولی و یا تدریجی نمایی بود حاکی از نا ایستا بودن مدل و نیاز به تفاضل گیری مجدد می باشد .



شکل ۲_۱۷: نمودار ACF یک سری زمانی نا ایستا

اگر دنباله یا سری زمانی را با X_t نشان دهیم، مقدارهای با تاخیر (h)را به صورت X_{t+h} خواهیم داشت. در اینجا منظور از تاخیر، فاصله‌زمانی (lag) است که بین مشاهدات در نظر گرفته‌ایم.

فرض کنید X_t مقدار سری زمانی را در زمان t نشان دهد. تابع خودهمبستگی برای این سری زمانی، ضرایب همبستگی بین مشاهدات X_t و X_{t+h} را براساس $h=1,2,3,\cdots$ نشان می‌دهد. بنابراین از لحاظ محاسباتی خودهمبستگی بین X_t و X_{t+h} به صورت زیر قابل محاسبه است.

$$AC(\mathrm{x}t, xt+h) = \frac{Cov(xt, xt+h)}{\sigma(xt).\sigma(xt+h)}$$

از آنجایی که در سری ایستا، واریانس به h بستگی ندارد می‌توان با جایگزینی مخرج کسر، عبارت بالا را به صورت ساده‌تری نیز نوشت.

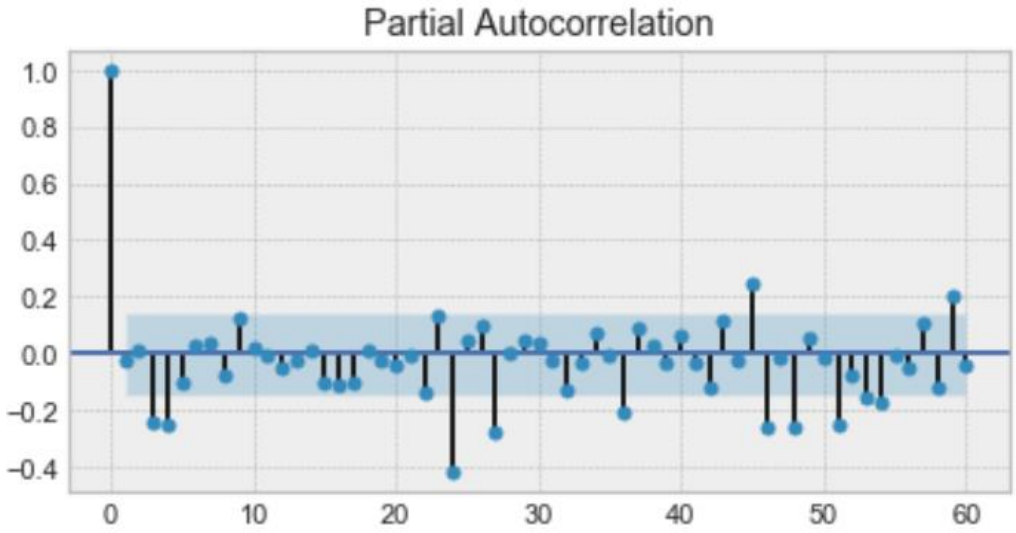
$$AC(\mathrm{x}t, xt+h) = \frac{Cov(xt, xt+h)}{Var(xt)}$$

به این ترتیب مخرج کسر ثابت بوده و از طرف دیگر می‌دانیم یکی از شرایط سری ایستا، ثابت بودن کوواریانس نیز هست. در این صورت انتظار داریم در یک سری زمانی ایستا، مقدار تابع خود همبستگی براساس تاخیر h تابعی از h باشد. این موضوع اساس تشکیل تابع خود همبستگی را نشان می‌دهد. بسیاری از سری‌های زمانی ایستا، دارای الگوی مشخصی برای تابع خود همبستگی هستند.

۲-٤-۷ خودهمبستگی جزئی

روش دیگر برای عنوان نمودن وابستگی زمانی در ساختار یک سری زمانی ، تعریف تابع خود همبستگی جزئی می باشد . عبارت است از همبستگی بین دو متغیر تصادفی X_t و X_{t+h} پس از حذف وابستگی خطی منحصر به فرد آن ها با مقادیر بینابینی X_{t+1} , X_{t+2}, X_{t+3} X_{t+4} ..... X_{t+k-1} (وابستگی بین دو بازه ی زمانی بدون در نظر گرفتن مقادیر بازه های زمانی بینشان) به عبارت دیگر $Corr(X_{t+h}, X_t | X_{t+1},....., X_{t+k-1})$ به دلیل پیچیده بودن محاسبات به جزئیات آن نمی پردازیم . اگر مقادیر خود همبستگی جزئی نزدیک صفر باشند ، مقادیر نمونه مورد نظر X_t ونمونه ی متناظر با lag X_{t+h} بهم وابسته نیستند و

بالعکس اگر این مقادیر نزدیک به -۱و +۱ باشند با هم وابسته ی شدیدند . در این نمودار یک منطقه با سایه ی آبی رنگ مشخص شده است .اگر مقادیر خودهمبستگی جزئی از این منطقه فرا تر روند می توان نتیجه گرفت که آن ها از لحاظ آماری قابل توجه می باشند .



شکل ۲_۱۸: نمودار خود همبستگی جزئی

برای تخمین مقدار مناسب برای پارامترمدل اتورگرسور(p)که جلوتر به آن می پردازیم از تابع و ضریب همبستگی جزئی استفاده می‌شود. همانطور که در نمودار زیر می‌بینید با در نظر گرفتن درجه ٤ برای تاخیرها، میزان تابع خودهمبستگی به صفر می‌رسد بنابراین می‌توان نتیجه گرفت که ٤برای پارامتر میتواند مناست باشد .

حال نوبت آن است که با مدل های سری زمانی آشنا شویم.

۲-٤-۸ مدل میانگین متحرک

یک مدل میانگین متحرک از خطاهای پیش بینی های قبلی برای به دست آوردن مقدار سری زمانی در t استفاده می کند . مدل میانگین متحرک (Moving Average) با پارامتر qمشخص می‌شود. این پارامتر نشان می‌دهد که مقدار سری زمانی در زمان tبا q مقدار خطا(اختلاف بین مقدار واقعی و مقدار پیش بینی شده) یا نویز سفید تشکیل یک مدل خطی می‌دهد که آن را با MA(q) نشان می دهند .

در این مدل سری زمانی X(t)به صورت زیر نوشته می‌شود:

$$X(t) = \mu + \theta_1 \epsilon_{(t-1)} + \cdots + \theta_q \epsilon_{(t-q)} + \epsilon_t$$

$X(t)$: مقدار عددی سری زمانی در زمان t

μ: میانگین کلی سری زمانی تا t

θ : ضرایب معادله که توسط الگوریتم های بهینه سازی در راستای کمینه کردن میانگی مربعات خطا به دست می آید

ϵ_t : خطا یا اختلاف بین مقدار واقعی و مقدار پیش بینی شده در t

استفاده از این مدل باید با در نظر گرفتن یک شرط صورت پذیرد. شرطی که در این مدل باید رعایت شود، آن است که می‌بایست قدرمطلق ضرایب مدل کوچکتر از یک باشند.

۲-٤-۹ مدل اتورگرسیو(خود همبسته)

اگر مقدارهای سری زمانی ایستا به صورتی باشند که به مقدارهای قبلی خود بستگی داشته باشند، از مدل اتورگرسیو استفاده می‌شود. در این حالت p را تعداد مشاهدات گذشته در نظر می‌گیریم که برای پیش‌بینی

یک مقدار در نظر گرفته می‌شود. بنابراین می‌توانیم مدل اتورگرسیو را برای سری زمانی ایستا X(t)به صورت زیر بنویسیم:

$$X(t) = a_۰ + a_۱ x_{(t-۱)} + a_۲ x_{(t-۲)} + \cdots + a_p x_{(t-p)} + \epsilon_t$$

ر این رابطه $a_۰, a_۱, a_۲, \ldots \ldots, a_p$ پارامترهای مدل اتورگرسیو و ϵ_t خطای تصادفی در نظر گرفته می‌شود. چنین مدلی را به صورت $AR_{(p)}$نشان می‌دهند و p را مرتبه مدل می‌نامند. انتخاب مقدار p نیاز به بررسی سری زمانی و میزان همبستگی مقدارهای سری زمانی به یکدیگر دارد.

از آنجایی که این مدل به مانند یک مدل رگرسیون نوشته شده، به آن مدل اتورگرسیو می‌گویند، با توجه به اینکه برای X(t)رگرسیون روی مقدارهای گذشته ایجاد شده، شرط استقلال متغیرهای توصیفی حداقل برای تعداد دسته‌های کوچکتر از p وجود نخواهد داشت. همچنین مقدار حال حاضر سری زمانی فقط به pمقدار قبلی وابسته بوده و به قبل از آن ارتباطی ندارد.

معمولا برای بررسی سری زمانی ایستا از مدل اتورگرسیو مرتبه اول یا دوم استفاده می‌شود. مدل اتورگرسیو مرتبه دوم به صورت زیر است:

$$X(t) = a_۰ + a_۱ x_{(t-۱)} + a_۲ x_{(t-۲)} + \epsilon_t$$

این مدل نشان می‌دهد که مقدار متغیر در زمان t به مقدار متغیر در زمان‌های t−1و t−2 وابسته است و مدل این وابستگی نیز خطی است. از طرفی مقدار خطا یا نویز سفید ϵ_t نیز در مقدار $X(t)$ نقش دارد.

حال برای بدست آوردن ضریب کافی است که رگرسیون حاصل از مقدارهای X(t) را روی X(t−1),X(t−2)بدست آوریم. توجه داریم که در این حالت، روی سری زمانی ایستا محاسبات باید انجام

بپذیرد.شرط درست بودن این مدل نیز مانند مدل میانگین متحرک آن است که قدر مطلق ضرایب (پارامتر ها (a_p)) باید کوچکتر از یک باشد .

۲-٤-۱۰ مدل اتورگرسیو میانگین متحرک(ARIMA)

همانطور که حدس می‌زنید، این مدل به صورت ترکیبی از دو مدل ساده AR(p) و MA(q) ساخته می‌شود. به همین دلیل نیز برای نمایش این مدل از نماد ARMA(p,q)استفاده می‌شود. رابطه بین مقدارهای سری زمانی ایستا در این مدل به صورت زیر قابل مشاهده است:

$$X(t) = a_۰ + a_۱ x_{(t-۱)} + \cdots + a_p x_{(t-p)} + \epsilon_t + \theta_۱ \epsilon_{(t-۱)} + \cdots + \theta_q \epsilon_{(t-q)}$$

همان طور که گفته شد برای برآورد پارامترهای این مدل از تابع خودهمبستگی (Auto-correlation) و تابع خودهمبستگی جزئی (Partial Auto-correlation) استفاده می شود . همچنین رسم نمودارهای مربوط به این توابع همبستگی، برای تشخیص و انتخاب مدل مناسب راهگشا است.تابع خودهمبستگی بعد از چند lag صفر میشود و یا به طور نمایی کاهش مییابد. با افزایش lag،ضریب خودهمبستگی پس از چند وقفه به صفر میرسد، در غیر اینصورت به سرعت کاهش مییابد.برای یک سری زمانیMA(q) تابع خودهمبستگی آن بعد از q دوره به صفر میگراید و این یعنی مقدار متغیر در دوره فعلی صرفا تابعی از q دوره قبل متغیر است و لذا جایی که ACF به صفر میرسد می تواند تخمین مناسبی برای مرتبه MA(q)تعیین شود .

برای هر سری زمانی بعد از p دوره PACF آن به صفر میرسد و بنابراین جایی که تابع خودهمبستگی جزئی به صفر میگراید، مرتبه AR(p)تعیین میشود. اما در عمل برای تعیین کردن p و q بهینه از معیار سنجش AIC که به معیار اطلاع آیکایکی معروف است استفاده می کنیم که جلو تر به آن می گردازیم .

۲-٤-۱۱ مدل اتورگرسیو یکپارچه ی میانگین متحرک (ARIMA)

یک حالت از توسعه مدل ARMA به عنوان مدل اتورگرسیو یکپارچه میانگین متحرک ARIMA معروف است. همانطور که در قبل گفته شد، اگر سری زمانی، ایستا نباشد می‌توان به کمک تبدیل تفاضلی آن را به حالت ایستا درآورد مدل ARIMA نیز از همین خاصیت استفاده کرده و برای سری‌های زمانی ناایستا، به کمک عملگر تفاضل‌گیری، ایستایی را ایجاد و مدل مناسب را ارائه خواهیم داد. در واقع اگر از ARIMA استفاده کنیم دیگر نیاز نیست قبل از مدلسازی سری نا ایستا را ایستا کرد بلکه با تعیین مرتبه ی پارامتر q در مدل ARIMA می شود این کار را انجام داد.

سری $x_{(t)} - x_{(t-۱)}$ تفاضل مرتبه اول سری زمانی نامیده می شود . اگر تفاضل گیری مرتبه اول برای به وجود آوردن ویژگی های مدل ایستا (ایجاد تابع خود همبستگی که بعد از چند وقفه به صفر برسد یا به صورت نمایی کاهش یابد) موفق نبود ،تفاضل گیری مجدد سری زمانی (تفاضل گیری داده های تفاضل گیری شده) نیاز است که می شود تفاضل گیری مرتبه دوم . البته معمولا به بیش از یک یا دو مرتبه تفاضل گیری نیاز نیست برای رسیدن به تابع خود همبستگی با ویژگی های مذکور نیاز نیست . تعداد دفعاتی که سری زمانی باید تفاضل گیری شود را با d نشان می دهند و مرتبه ی تفاضل گیری نامیده می شود . این مدل دارای سه پارامتر P,d,q است و به صورت ARIMA(p,d,q) نمایش داده می شود و فرمول با آن با ARMA یکی است .

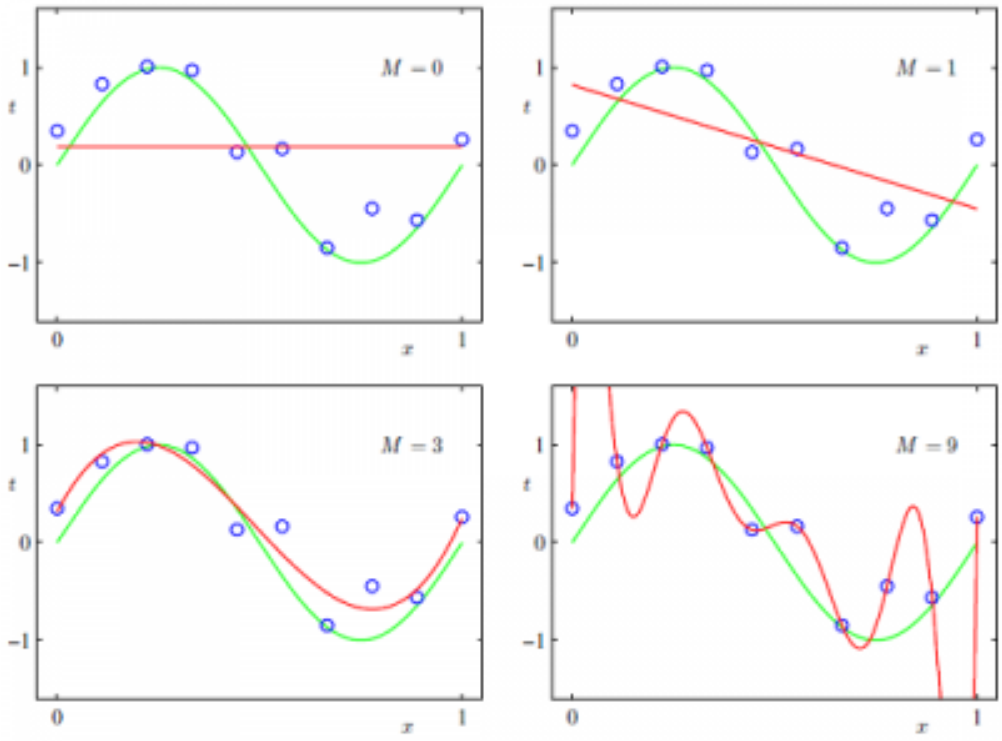
۲-٤-۱۲معیار ارزیابی آکایکه (AIC) :

همان طور که گفته شد تعیین پارامتر ها از روی نمودار های همبستگی نمی تواند بهترین راه برای تعیین پارامتر های بهینه باشد و از سال ۲۰۰۰ میلادی این روش منسوخ شده است . به منظور انتخاب بهینه ی

پارامتر ها به این صورت عمل می کنیم که مدل های مختلف را با p , q های مختلف امتحان می کنیم و آن مدلی را که AIC آن از همه کمتر است را بر می گزینیم .

اغلب به منظور ارزیابی مدلی که برای پدیده‌های تصادفی (Random Phenomena) ایجاد شده است از شاخص‌هایی استفاده می‌کنیم که براساس داده‌ها و توزیع مقدارهای آن پدیده تصادفی حاصل می‌شود. معمولا شکل همگونی داده‌ها با توزیع مورد نظر را با تابع درستنمایی (Likelihood Function) اندازه‌گیری می‌کنند. از طرفی اگر تعداد پارامترهای مدل زیاد باشد ممکن است مدل برای داده‌های موجود به خوبی برازش شده ولی برای داده‌های جدید مناسب نباشد. به این موضوع مسئله بیش‌برازش (Overfitting)گفته می‌شود. به این ترتیب با توجه به تاثیر تعداد پارامترها و تابع درستنمایی، معیار ارزیابی AIC یا معیار ارزیابی اطلاع آیکاکه (Akaike Information Criterion – AIC) یکی از این شاخص‌ها است که به هر دو معیار برای مناسب بودن مدل توجه داشته و بخصوص در تئوری اطلاع (Information Theory)نیز مورد بهره‌برداری قرار می‌گیرد. معیار ارزیابیAIC ، نمایانگر میزان اطلاعاتی است که توسط مدل از دست رفته و در نتیجه هر چه مقدار معیار ارزیابی AIC کوچکتر باشد، مدل مورد نظر نسبت به بقیه مدل‌ها، بهتر و مناسب‌تر است AIC .یک تعادل بین تعداد پارامترهای مدل (پیچیدگی مدل) و میزان برازش مدل روی داده‌ها ارائه می‌کند. با در نظر گرفتن این موضوع می‌توان گفت مدلی که توسط AIC مدل مناسب تشخیص داده شود، نه دارای بیش‌برازش (Overfitting)است و نه از کم‌برازش (Underfitting) رنج می‌برد و می‌توان آن را مدلی با برازش مناسب در نظر گرفت. معیار ارزیابی AIC را می‌توان برآوردگر کیفیت نسبی مدل آماری با توجه به داده‌های جمع‌آوری شده در نمونه تصادفی دانست. به این ترتیب به کمک معیار ارزیابی AIC می‌توان یک رابطه ترتیبی (Ordered

Relation)بین مدل‌ها، به منظور مقایسه و سنجش برتری بین آن‌ها بدست آورد. بنابراین از این شاخص به منظور انتخاب بهترین مدل از بین مدل‌های آماری موجود می‌توان استفاده کرد.



شکل ۲_۱۹: بررسی تعداد پارامتر بهینه با توجه به AIC

فرض کنید در یک مدل آماری kتعداد پارامترهای مدل باشد. اگر ^Lرا حداکثر تابع درستنمایی برای مدل در نظر بگیریم، معیار ارزیابی AIC توسط رابطه زیر قابل محاسبه است.

$$AIC = ۲k - ۲\ln(L^{\wedge})$$

بنابراین مناسب‌ترین مدل برحسب معیار اطلاع آکایکه، دارای کمترین مقدار AIC است. از آنجایی که معیار ارزیابی AIC برحسب حداکثر تابع درستنمایی محاسبه می‌شود، به شکلی میزان نیکویی برازش را اندازه‌گیری می‌کند. از طرفی نیز به منظور تعیین جریمه برای تعداد پارامترهای مدل و جلوگیری از پیچیدگی آن از k نیز کمک گرفته شده است. مشخص است که هر چه مقدار k یعنی تعداد پارامترهای مدل، بیشتر باشد، مقدار AIC نیز بزرگتر در نتیجه میزان اطلاعاتی که توسط مدل نادیده گرفته شده است،

بیشتر خواهد بود. بنابراین مدلی که بتواند کمترین میزان پیچیدگی و در عین حال بیشترین میزان برازش را داشته باشد، مدل مناسب تشخیص داده خواهد شد. همانطور که گفته شد معیارAIC ، به صورت نسبی عمل می‌کند و نمی‌توان صرفا با اندازه‌گیری معیار ارزیابی AIC برای یک مدل، تشخیص داد که بهترین مدل حاصل شده است، بلکه AIC مدل را باید با AIC مدل‌های دیگر که به نظر مناسب می‌رسند، مقایسه کرده و بهترین مدل را از بین مدل‌های موجود براساس کمترین مقدار AIC انتخاب کرد.

۲-٤-۱۳ پیش بینی و اندازه گیری خطا درآن

بیشتر تصمیمات مدیریت در تمام سطوح سازمان به طورمستقیم و یا غیرمستقیم به حالتی از پیشبینی آینده بستگی دارد. در یک تعریف کلی، پیشگویی شرایط وحوادث آینده را پیشبینی و چگونگی انجام این عمل،پیش بینی کردن نامیده میشود. از آنجا که پیش بینی وقایع آینده در فرآیند تصمیمگیری نقش عمدهای را ایفا میکند، لذا پیشبینی کردن برای بسیاری از سازمانها و نهادها حائز اهمیت است و هر سازمانی برای تصمیمگیری آگاهانه باید قادر به پیشبینی آینده باشد .برای بررسی یک مدل پیشبینی و یا انتخاب بهترین مدل از بین مدلهای مختلف برای سری زمانی به شاخصی نیاز داریم که به کمک آن تصمیم لازم در خصوص قبول یا رد مدل پیشبینی اتخاذ شود. به علاوه در تمام پیش بینی ها عدم اطمینان وجود دارد. این حقیقت از جزء غیر معمول در سری زمانی معلوم میشود. در نتیجه درکلیه روشهای پیش بینی باید انتظار خطا را داشته باشیم .به طور کلی هر چه مقدار واقعی سری y_t به مقدار پیشبینی شده آن$y\hat{}t$نزدیکتر باشد، بر صحت بیشتر مدل پیش بینی دلالت دارد. بنابراین کیفیت یک مدل با بررسی میزان خطای پیش بینی یا همان et قابل ارزیابی است.

$$et = yt - y\hat{}t$$

خطای پیشبینی ناشی از آن است کـه یـک یـا چنـد مؤلفه از مؤلفه های پیشبینی سـری زمـانی مـانند رونـد، فصـلی، دوره، به حسـاب نیامده اسـت و یـا نوسـانات بـی قاعده و نامنظم بـوده. مجمـوع کـل خطاهـا در یـک روش پیش بینی که در آن nمجموع دورههای زمـانی مـشاهده شده باشد عبارت است از:

$$AE = \text{انحراف مطلق خطا} = |e_t| = |y_t - \hat{y}_t|$$

حال برای به دست آوردن خطـا در n پریـود زمـانی از میانگین قدر مطلق خطا (MAD) استفاده مـیشـود کـه عبارت است از:

$$MAD = \frac{\sum |y_t - \hat{y}_t|}{n}$$

این خـطا را به گـونهای دیگر نیـز مـیتـوان محاسـبه نمـود و آن اسـتفاده از میـانگین مجـذور خطـا MSEمیباشد که به صورت ذیل محاسبه میشود.

$$MSE = \frac{\sum [y_t - \hat{y}_t]^۲}{n}$$

در میانگین مجذور خطا میزان خطا به دلیل به توان ۲رسیدن آن بسیار بزرگ نشان داده میشـود بـا اسـتفاده ازجذر میانگین مجذور خطاRMSE ایـن مـشکل مرتفـع میشود

$$RMSE = \sqrt{\frac{\sum [y_t - \hat{y}_t]^۲}{n}}$$

"فصل سوم"

"مبانی نظری شبکه های عصبی و بررسی الگوریتم های آن"

۳-۱ پیدایش شبکه های عصبی مصنوعی

مغز انسان، به اذعان بسیاری از دانشمندان، پیچیده ترین سیستمی است که تا کنون در کل گیتی مشاهده شده و مورد مطالعه قرار گرفته است. اما این سیستم پیچیده نه ابعادی در حد کهشکشان دارد و نه تعداد اجزای سازنده‌اش، بیشتر از پردازنده‌های ابررایانه‌های امروزی است. پیچیدگی راز آلود این سیستم بی نظیر، به اتصال‌های فراوان موجود میان اجزای آن بازمی‌گردد. این همان چیزی است که مغز ۱٤۰۰ گرمی انسان را از همه سیستم‌های دیگر متمایز می کند.

فرایندهای خودآگاه و ناخودآگاهی که در حدود جغرافیایی بدن انسان رخ می‌دهند، همگی تحت مدیریت مغز هستند. برخی از این فرایندها آن‌قدر پیچیده هستند، که هیچ رایانه یا ابررایانه‌ای در جهان امکان پردازش و انجام آن را ندارد. با این حال، تحقیقات نشان می‌دهند که واحدهای سازنده مغز انسان، از نظر سرعت عملکرد، حدود یک میلیون بار کندتر از ترانزیستورهای مورد استفاده در تراشه های سیلیکونی CPUرایانه هستند.

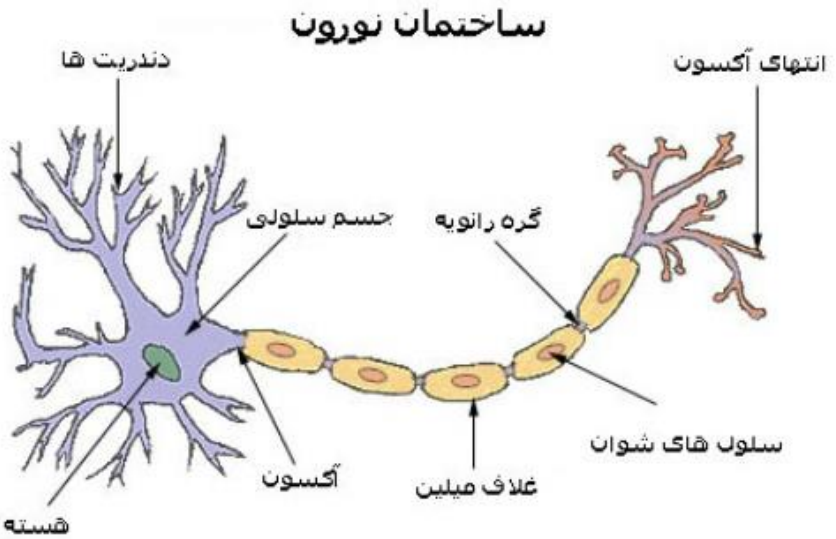
سرعت و قدرت پردازش بسیار بالای مغز انسان، به ارتباط‌های بسیار انبوهی باز می‌گردد که در میان سلول‌های سازنده مغز وجود دارد و اساساً، بدون وجود این لینک‌های ارتباطی، مغز انسان هم به یک سیستم معمولی کاهش می‌یافت و قطعاً امکانات فعلی را نداشت.

گذشته از همه این ها، عملکرد عالی مغز در حل انواع مسائل و کارایی بالای آن، باعث شده است تا شبیه سازی مغز و قابلیت های آن به مهم‌ترین آرمان معماران سخت‌افزار و نرم‌افزار تبدیل شود. در واقع اگر روزی فرا برسد (که البته ظاهرا خیلی هم دور نیست) که بتوانیم رایانه‌ای در حد و اندازه‌های مغز انسان بسازیم، قطعاً یک انقلاب بزرگ در علم، صنعت و االبته زندگی انسان‌ها، رخ خواهد داد.

از چند دهه گذشته که رایانه‌ها امکان پیاده‌سازی الگوریتم‌های محاسباتی را فراهم ساخته‌اند، در راستای شبیه‌سازی رفتار محاسباتی مغز انسان، کارهای پژوهشی بسیاری از سوی متخصصین علوم رایانه، مهندسین و همچنین ریاضی‌دان‌ها شروع شده است، که نتایج کار آن‌ها، در شاخه‌ای از علم هوش مصنوعی و در زیرشاخه هوش محاسباتی تحت عنوان موضوع «شبکه های عصبی مصنوعی» یا Artificial Neural Networksبه اختصار ANNs طبقه بندی شده است. در مبحث شبکه های عصبی مصنوعی، مدل های ریاضی و نرم‌افزاری متعددی با الهام گرفتن از مغز انسان پیشنهاد شده‌اند، که برای حل گستره وسیعی از مسائل علمی، مهندسی و کاربردی، در حوزه های مختلف کاربرد دارند.

۳-۲ ساختار نورون های مغز انسان

مغز انسان دارای تعداد بسیار و بسیار زیادی نورون (Neuron) است که اطلاعات را از نورون های دیگر دریافت می کند. مغز انسان توسط شبکه ای از نورون ها می تواند حجم بالایی از اطلاعات را پردازش کند که این اطلاعات توسط انسان حس (Senses) شده اند. به طور مثال چشم انسان محیط را می بینید و سپس اطلاعت را به مغز می فرستد و مغز تصویر دریافت شده را پردازش می کند.اطلاعات (تصویر های ورودی از چشم) به نورن های مغز ارسال می شوند و همچنین پردازش توسط نورون ها انجام می شود. در نهایت با فعال سازی یک سیگنال الکتریکی و شیمیایی، اطلاعات به نورون های دیگر فرستاده می شود، پس می بینیم که مغز انسان شامل یک شبکه پیچیده ای از نورون های متصل به هم هستند که اطلاعات میان نورون ها منتقل می کنند . شکل زیر ساختار یک نورون مغز انسان را نشان می دهد.



شکل ۳- ۱: ساختار یک نورون

۳-۲-۱ دندریت ها

دندریت (Dendrites) نقطه ورودی برای هر نورون هستند که بر اثر فعال سازی سیگنال الکترونیکی،ورودی را از نورون دیگر دریافت می کند. در واقع دندریت ها پیام عصبی (اطلاع یا داده یا ورودی) را از نورون دیگر دریافت و سپس به جسم سلولی (Cell Body) ارسال می کنند.

۳-۲-۳جسم سلولی یا هسته نورون

جسم سلولی یا هسته نورون که سوما (Soma) نیز نامیده می شود، استنتاج ها و نتیجه گیری ها (Inferences)را از روی ورودی های دندریت ها ایجاد می کند و تصمیم می گیرد که چه عکس العملی (Action) باید رخ دهد.

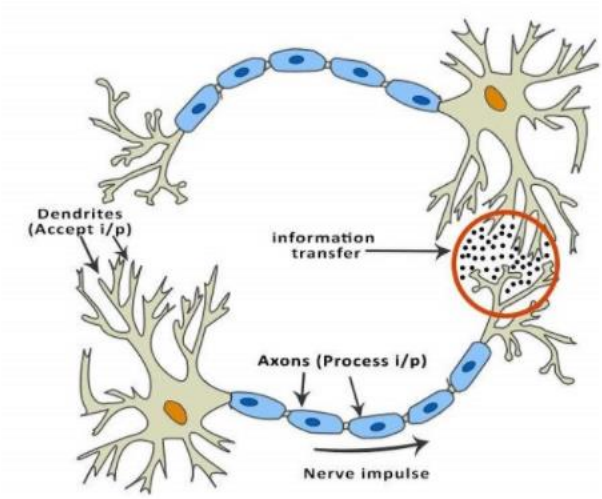
۳-۲-۳ انتهای آکسون

انتهای آکسون (Axon Terminal) خروجی ها را بر اساس یک سیگنال الکترونیکی به یک نورون دیگر انتقال می دهد.

توجه کنید که توضیح بیشتر در مورد اجزی نورون مربوط به حوزه پزشکی و خارج از این پژوهش است ولی به عنوان خلاصه باید بگوییم که در شبکه های عصبی مصنوعی (Artificial Neural Network یا به اختصار ANN) یک سری ورودی ها داریم که در مغز از طریق دندریت ها وارد می شوند و سپس یک پردازش روی ورودی ها انجام می شود که منجر به ایجاد استنتاج و نتیجه گیری هایی می شود و در نهایت خروجی ها تولید می شوند. بنابراین یک شبکه عصبی حداقل از سه لایه، لایه ورودی (Input Layer)، لایه پردازش (Hidden Layer) و لایه خروجی (Output Layer) تشکیل شده است. در مباحث یادگیری ماشین از مفهوم شبکه های عصبی برای ایجاد الگوریتم های یادگیری استفاده می شود.

۳-۲-٤ ارتباط نورون ها با یکدیگر

همانطور که توضیح داده شد نورون های مغز (یا به عبارت کاملتر نورون های زیستی Biological Neuran) و چگونگی ارائیاط میان آنها، انگیزه ای برای ایجاد شبکه های عصبی مصنوعی یا AAN شده است. بنابراین در مغز انسان یک شبکه ی عصبی (Network Neural) وجود دارد که از ارتباط میان نورون های مختلف پدید آمده است. به عبارت دیگر در مغز انسان هر نورون به نورون دیگر می تواند پیام ارسال کند. شکل زیر و در ناحیه قرمز رنگ می توانید ارتباط میان دو نورون را ببینید.



شکل ۳- ۲: ارتباط میان نورون ها

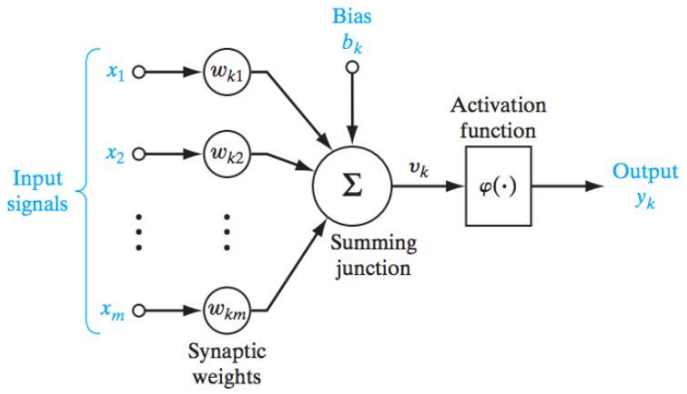
هر نورون از طریق پایانه آکسون خودش به دندریته های نورون دیگر پیام (یا اصطلاحا سیگنال) ارسال می کند. بنابراین سیگنال های ورودی از طریق دندریت های نورون با وزن سیناپسی (synaptic Weights)ضرب می شوند و سپس به جسم سلولی (Cell Body) ارسال و در آنجا انباشته (Accumulated)می شوند. اگر قدرت (Strength) سیگنال حاصل از آن بالاتر از یک آستانه (Threshold) مشحض باشد، نورون پیام را به آکسون (Axon) می فرستند. با توجه به عملکر نورون های مغز، تابع فعال سازی تصمیم می گیرد که آیا یک سیگنال به آکسون منتقل شود یا نه؟ زمانی که چیز جدید را یاد می گیریم، آستانه و وزن سیناپسی برخی از نورون ها تغییر می کنند و این باعث یک ارتباط

جدید میان نورون ها می شود (ناحیه قرمز رنگ شکل بالا) که در واقع به معنی یادگیری چیزهای جدید در مغز ما انسان ها است.

۳-۳ ساختار شبکه ی عصبی مصنوعی

۳-۳-۱ نورون ها و گره ها در شبکه ی عصبی

در شباهت و مقایسه با نورون های مغز که وظیفه پردازش پیام های عصبی را دارند، در شبکه های عصبی هر نورون معادل با یک واحد پردازش (Process Unit) است که یک سری ورودی را دریافت (لایه ورودی - Input Layer) و سپس یک سری پردازش ها را از طریق عملگرها و رابطه های ریاضی در لایه Hidden Layer انجام می دهد و در نهایت یک سری خروجی (Output Layer) را ایجاد می کند.بنابراین در یک شبکه عصبی (Neural Network) کوچکترین واحد پردازش یک نورون است که آنرا یک گره (Node) یا واحد (Unit) نیز می نامند. هر گره یک سری ورودی ها را از گره های دیگر یا از منابع خارجی (External Sources) دریافت می کند. به هر ورودی یک وزن (Weight) انتساب داده می شود. در مفاهیم شبکه های عصبی مصنوعی و یادگیری ماشین (Machine Learning) به این ورودی ها اصطلاحا ویژگی ها (Features) گفته می شود و به خروجی ایجاد شده برچسب (Label) گفته می شود.



شکل ۳- ۳: ساختار یک نورون در شبکه ی عصبی مصنوعی

۳-۳-۲ وزن ها و بایاس ها در شبکه ی عصبی مصنوعی

هر اتصال (Connection) دارای یک وزن (Weight) ویژه خودش است که در آغاز به گونه تصادفی برگزیده می شود. همانطور که داده شد ، ورودی ها در این وزن ها ضرب می شوند. سپس تمامی این حاصل ضرب های ورودی در وزن به تابعی ارسال می شود که آنرا اصطلاحا تابع زیگما ∑ گفته می شود .

اگر با مدل رگرسیون خطی (Linear Regression) آشنا باشید، وزن ها در شبکه عصبی، مفهومی همسان به ضریب ها در معادله رگرسیون خطی و ورودی های شبکه نورون نیز، مفهومی همسان با متغیرهای معادله خطی دارند. همچنین مفهوم دیگری به نام بایاس است که این مفهوم در شبکه عصبی، همسان با مفهوم شیب از مبدا (عرض از مبدا) در معادله خطی است.

ولی بایاس (Bias) چیست و چه کاربردی دارد؟ همانگونه که گفتیم، وزن ها در آغاز تصادفی برگزیده می شوند، پس اگر بایاس نباشد، چون همه وزن ها نیز صفر هستند، پس حاصل ضرب هر وزن در ورودی اتصال خودش برابر با صفر و از این رو مجموع آنها صفر است، پس برای پیش گیری از این، یک ورودی به نام بایاس می دهند که شماره نا صفر (غیر صفر) است. بنابراین با توجه به آن چه که گفتیم، اگر گمان کنیم که نورون دارای ۱۰ ورودی است، پس به چند وزن برای آن نیاز است؟ پاسخ ۱۱ وزن است که ۱۰ وزن برای هر ورودی نورون و یک وزن نیز که همان بایاس است. وزن ها به گونه تصادفی و با مقداری کوچک برای نمونه میان صفر تا ۰.۳ تعیین می شوند.

$$Y = \sum (weight * input) + bias$$

۳-۳-۳ تابع فعال ساز (Activation Function)

در شکل ۳-۱ تابع زیگما (∑) معمولا یک تابع غیر خطی (None Linear Function) است که اصطلاحا آنرا تابع فعال سازی (Activation Function) یا تابع انتقال (Transfer Function) می نامند. تابع فعال سازی برای محدود کردن دامنه خروجی یک نورون استفاده می شود.وظیفه تابع فعال سازی این است که آیا نورون باید فعال (اصطلاحا Fire) شود یا نه؟ در واقع تابع فعال سازی با بررسی مقدار Y و مقایسه آن با مقدار آستانه (Threshold) نورون تصمیم به فعال سازی نورون می گیرد.

هر تابع فعال سازی یک عدد را دریافت می کند و سپس عملیات ریاضی را بر روی آن انجام می دهد تا نهایتا یک عدد بین بازه صفر تا ۱+ یا بین بازه ۱- تا ۱+ را بر می گرداند. بنابراین به بیان ساده می توانیم عملکرد شبکه های عصبی (Neural Network) را محاسبه خروجی تمامی نورون های (گره های) درون شبکه عصبی در نظر بگیریم. توجه کنید که تابع فعال سازی مبتنی بر مفاهیم و عملیات های ریاضی است.

در شکل ۳-۱ می بینید که مجموع حاصل ضرب وزن ها در ورودی ها به تابع زیگما فرستاده شده است و سپس نتیجه آن به عنوان ورودی به تابع فعال سازی فرستاده می شود و در نهایت تابع فعال سازی است که نتیجه خروجی را مشخص می کند. همچنین در فرمول بالا ، Y همان تابع زیگما است که شامل مجموع حاصل ضرب وزن ها در ورودی ها بعلاوه بایاس است.

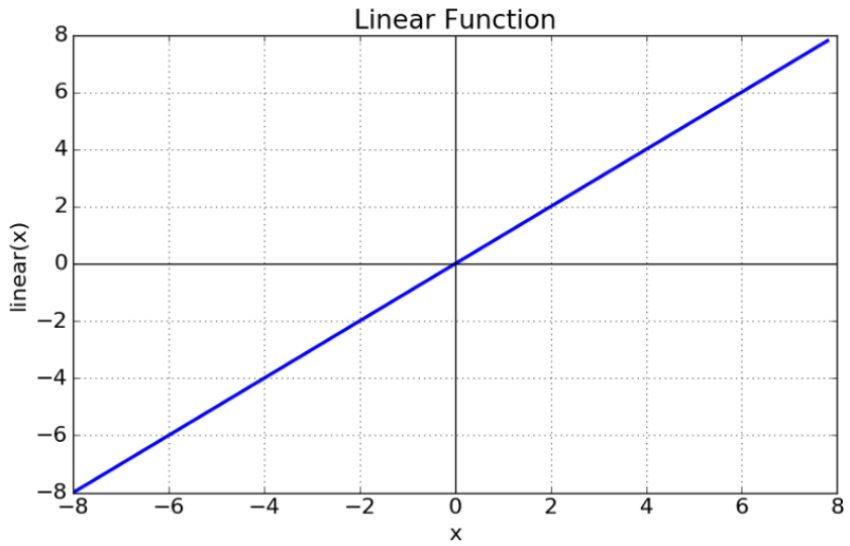
بنابراین مطابق با آنچه که در نورون های مغز اتفاق می افتد، هر نورون ورودی ها را جمع می کند تا به یک حد آستانه برسد. سپس یک سیگنال فعال می شود که باعث متصل شدن نورون ها به نورون دیگر می شود . در واقع خروجی یک نورون به ورودی نورون دیگر فرستاده می شود.

اصولا توابع فعال سازی به صورت غیر خطی (Non - Linear) هستند ولی به طور کلی می توانیم توابع فعال سازی را به دو دسته زیر تقسیم کنیم:

توابع فعال سازی خطی یا Linear Activation Function و توابع فعال سازی غیر خطی یا Non - linear Activation Function

۳-۳-۳-۱ تابع فعالسازی خطی

ساده ترین شکل توابع فعال سازی، از نوع تابع خطی F(x)=mx+b است. که در آن m شیب خط یا ضریب زاویه ای خط می گویند.b را عرض از مبدا می گویند که محل تقاطع خط با محور عمودی با محور yها را مشخص می کند. نمودار توابع خطی به صورت یک خط است.شکل زیر نمودار تابع خطی را نشان می دهد. چون در آن ۰ = b است پس محل تقاطع خط با محور عمودی در نقطه صفر است. توجه کنید در معادله خطی درجه X برابر با یک است. خروجی تابع بر اساس ورودی تابع قابل تشخیص است و میان بازه منفی بی نهایت تا مثبت بی نهایت خواهد بود.

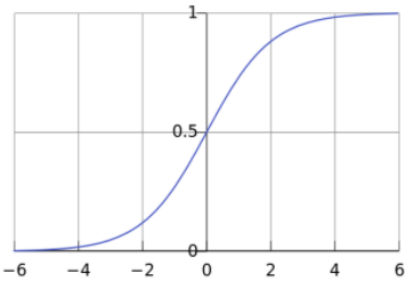


شکل ۳- ۴: تابع فعالساز خطی

۳-۳-۳-۲ تابع فعالسازی غیر خطی

تابع سیگموئید، که همچنین به نام تابع لجستیک (Logistic Function) نامیده می شود، منحنی به شکل S دارد که می تواند هر مقدار را به یک مقدار میان بازه ۰ و ۱ نگاشت دهد. اگر منحنی به سمت مثبت بی نهایت برود، پیش بینی به عدد ۱ نگاشت داده می شود و اگر منحنی به سمت منفی بی نهایت برود، پیش بینی به عدد ۰ نگاشت داده خواهد شد. اگر خروجی عملکرد sigmoid بیش از یک آستانه عددی مانند ۰.۵باشد، می توان نتیجه را به صورت ۱ یا YES طبقه بندی کرد، و اگر آن کمتر از این استانه عددی مثلا همان عدد ۰.۵باشد، می توان آن را به صورت ۰ یا نه مثلا خروجیNO: اگر خروجی ۰.۷۵ باشد، ما می توانیم از نظر احتمال: بله ۷۵ درصد احتمال دارد که بیمار از سرطان رنج می برد.

$$F(t) = \frac{1}{1+e^{-t}}$$



شکل ۳- ۵: تابع فعالساز غیر خطی سیگموید.

توابع غیر خطی انواع مختلفی دارند که ما فقط یک نوع از آن (سیگموید) را بررسی کردیم .

توجه کنید در مفاهیم شبکه های عصبی و یادگیری ماشین ابتدا یک سری داده های ورودی را موجود است که به آنها نمونه (Example) می گویند و سپس توسط این ورودی ها و داده هایی که بعدا وارد می شوند، می توان نتایج با خروجی ها را پیش بینی کرد. به عبارت دیگر توسط شبکه های عصبی و پیروی و تقلید یا شبیه سازی عملکرد مغز انسان و شبکه نورون های درون آن و استفاده از عملیات و مفاهیم ریاضی

می خواهیم این قابلیت را به کامپیوترها و ماشین ها بدهیم که بتوانند از یک سری ورودی ها و داده هایی که بعدا وارد می شوند، بتوانند پیش بینی هایی (Prediction) را انجام دهند.

۳-٤ لایه های شبکه ی عصبی مصنوعی

شبکه های عصبی در برگیرنده شبکه ای از نورون ها هستند که هر سطر از این نورون ها، یک لایه (Layer) نامیده می شوند که هر شبکه عصبی می تواند چندین لایه داشته باشد که به آنها شبکه های عصبی چند لایه ای (Multi Layers) گفته می شود. همچنین ساختار (یا معماری) نورون های درون یک شبکه عصبی را، توپولوژی آن شبکه می گویند. در هر شبکه عصبی، سه لایه ورودی (Input Layer)، لایه پنهان (Hidden Layer) و لایه خروجی (Output Layer) وجود دارد.

۳-٤-۱ لایه ی ورودی (Input Layer)

اولین لایه از شبکه ی عصبی را لایه ی ورودی می گویند و این همان لایه ای است که ورودی ها را از مجموعه داده ها دریافت می کند. در کشیدن و نمایش دادن یک شبکه عصبی، در لایه ی وردی به ازای هر ورودی یا هر ستون از مجموعه داده ها، یک نورون در آن هست. بنابراین اگر یک دیتا شامل دو ستون باشد در نتیجه دو نورون (گره) در لایه ورودی می باشد ، پس این شبکه دو ورودی X_1 و X_2را دریافت می کند.

۳-٤-۲ لایه ی پنهان و مفهوم Fully Connected

پس از لایه ورودی، لایه پنهان هست که بر پایه نام آن، این لایه از دید ورودی پنهان است. خروجی لایه بیرونی به لایه پنهان پیوند خورده است. همانند شبکه عصبی که از هر نورون لایه ورودی، یک خروجی (یال) به همه گره های لایه پنهان، پیوند خورده باشد، آن را شبکه Full Connected می گویند.

همچنین یادگیری ژرف با یادگیری عمیق (Deep Learning) بدین مفهوم است که چندین لایه پنهان در شبکه عصبی باشد.

۳-٤-۳ لایه ی خروجی (Output Layer)

واپسین لایه، لایه خروجی است که یک مقدار یا یک بردار (Vector) را برگشت می دهد که این خروجی مطابق با مسئله ای است که آن را مدل کرده اید. برگزیدن (انتخاب) تابع فعال سازی در لایه خروجی، بسیار از مسئله ای که آن را مدل می کنید، تاثیر می گیرد. در زیر برخی از این مسئله ها را فهرست کرده ایم.

۳-٤-۳-۱رگرسیون

یک مسئله رگرسیون، تک نورون خروجی داشته باشد و همچنین نورون لایه خروجی،باید تابع فعال ساز خطی داشته باشد.

۳-٤-۳-۲طبقه بندی دودویی

در مسئله های طبقه بندی دودویی (Binary Classification)، باید تک نورون خروجی داشته باشیم که تابع فعال ساز غیر خطی مانند Sigmoid را به کار ببرد. تابع Sigmoid، ورودی را دریافت و سپس عددی میان صفر یا یک را برگشت می دهد. بنابراین در مسئله های طبقه بندی دودویی، تابع فعال ساز Sigmoid، برای پیش بینی شانس (احتمال) یکی از دو کلاس مسئله به کار می رود. در اینجا مفهوم دیگری به نام آستانه (Threshold) هست که مقدار آن را مثلا ۰.۵ در نظر می گیریم و اگرمقدار تابع Sigmoid کوچکتر از مقدار آستانه باشد، پس پاسخ برای نمونه، Class A و اگر بزرگتر یا برابر ۰ . ۵ باشد، پس پاسخ Class B است.

۳-٤-۳-۳طبقه بندی چندتایی

در مسئله های چند طبقه بندی (Multiple Classification)، در لایه خروجی چندین نورون هست که هر نورون، برای یکی از کلاس درون مسئله است. در طبقه بندی های چندتایی، باید تابع غیر خطی دیگری به نام Softmax را به کار ببریم. در اینجا نیز به هر کدام از نورون های لایه ی آخر یک کلاس داده می شود و پاسخ یا همان کلاسی است که بیشترین مقدار را داشته باشد یعنی در واقع بیشترین احتمال وقوع را داشته باشد.

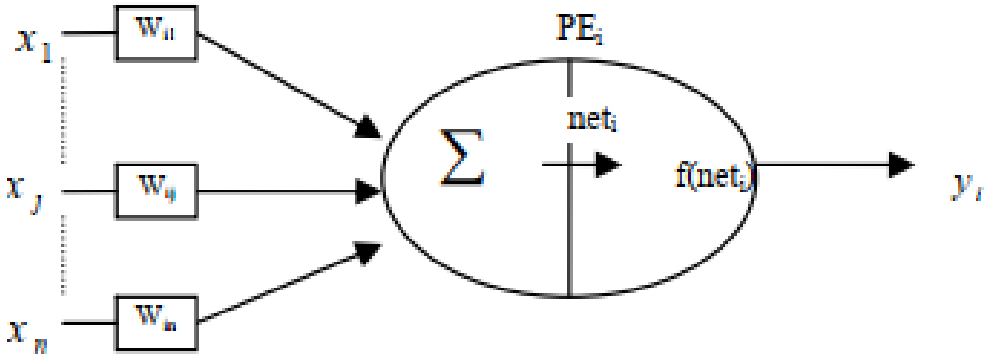
۳-۵ مدل های مختلف شبکه ی عصبی مصنوعی

شبکه های عصبی مصنوعی انواع مختلفی دارند که به بررسی چندتا از آن ها می پردازیم .

۳-۵-۱ مدل پرسپترون تک لایه (SLP)

ساده ترین مدل شبکه های عصبی مصنوعی، مدل پرسپترون تک لایه (Single Layer Perceptron) یا SLP است و این مدل، پایه ای برای مدل های پیچیده تری است که در شبکه های عصبی توسعه داده می شوند. معمولا شکبه عصبی SLP برای مسئله های طبقه بندی دودویی (Binary Classification) به کار می رود، جایی که داده ها بر پایه ویژگی هایشان متعلق به یکی از دو کلاس هستند.

توجه کنید بر پایه شکل زیر، شبکه های SLP بدون لایه پنهان (Hidden Layer) هستند و از این رو ورودی های شبکه، پس ضرب شدن در وزن ها (Weight) و تک مقدار بایاس (Bias)، سر راست (مستقیم) به لایه خروجی متصل هستند. در شکل زیر می بنید که مجموع حاصل ضرب های ورودی ها در وزن ها به تابع فعال سازی درون لایه خروجی فرستاده می شود و سپس کلاس (یا برچسب) ورودی ها پیش بینی خواهد شد.



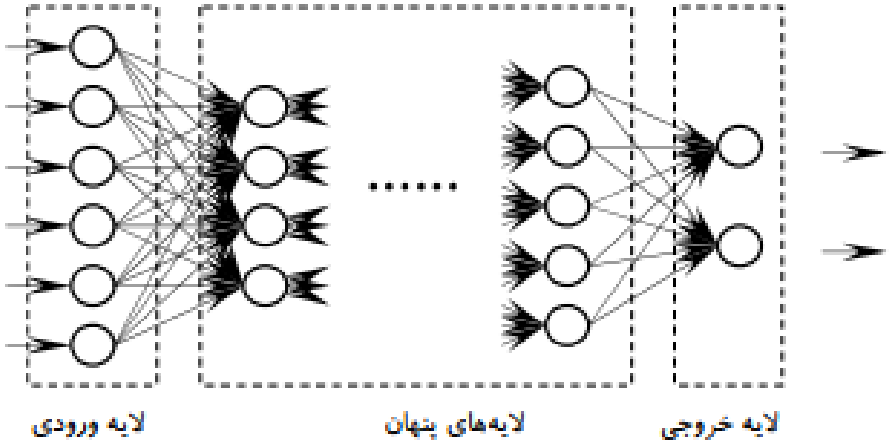
شکل ۳- ۶:یک عنصر پردازشی

Wها وزنهای اختصاص یافته به هر ورودی،net تـابع زیگما ، fتابع فعالسازی، x هـا ورودیهـای نـرون و y هـا خروجی های نرون می باشند.

۳-۵-۲ مدل پرسپترون چند لایه (MLP)

مدل پرسپترون چند لایه ای (Multi Layer Perceptron) یا MLP بسیار همانند SLP است با این تفاوت که این مدل دارای لایه های پنهان است که همه سه لایه ورودی، لایه (یا لایه های) پنهان و لایه خروجی به یکدیگر متصل شده اند و این مفهوم شبکه عصبی رو به جلو (Feed Forward) را می سازند.

بر پایه شکل زیر هر نورون در یک لایه سر راست (مستقیما به نرون های درون لایه پسین پیوند خورده است. یکی از کلیدی ترین جنبه هایی که مدل MLP را از SLP متمایز می کند، الگوریتم Back Propagationاست که شیوه ای رایج برای یادگیری (Training) شبکه عصبی است. انتشار برگشت (Back - Propagation) خطای محاسبه شده از لایه خروجی را به لایه ورودی منتقل می کند به گونه ای که می توانیم سهم هر لایه را در خطا مشاهده کنیم و شبکه را بر این اساس تغییر (بهبود) دهیم.



شکل ۳- ۷: ساختار کلی شبکه های عصبی چند لایه ی پیشخور

۳-۵-۲-۱یادگیری شبکه ی عصبی پرسپترون چندلایه

یادگیری شبکه عصبی پرسپترون با مقدار دهی اولیه وزن ها انجام می شود. این مقداردهی اولیه وزن ها به صورت تصادفی از مقدار هایی از یک توزیع نرمال (Normal Distribution) انجام می شود. همچنین می توانیم گرادیان کاهشی (Gradient Descent) برای بهبود یادگیری شبکه عصبی به کار ببریم. در واقع گرادیان کاهشی برای کمینه کردن تابع خطا (Error Function) است .یک شبکه عصبی مصنوعی که به خوبی آموزش دیده باشد، دارای وزن هایی است که سیگنال را تقویت و نویزها را کم می کند. ورودی های که در وزن های بزرگتر ضرب شوند، بر تفسیر شبکه بر داده ها بیشتر تاثیر می گذارند تا ورودی هایی که در وزن های کوچکتر ضرب شده اند. این به مدل ما کمک می کند تا یاد بگیرد که پیش بینی کننده ها (یا ویژگی ها – Features) به کدام خروجی ها گره خورده اند و وزن ها و بایاس را مطابق با آن تنظیم می کند.

فرایند یادگیری شبکه عصبی ، در برگیرنده فرایند تنظیم دوباره وزن ها و بایاس است، به گونه ای که ارزش (مقدار) برخی از وزن ها بزرگتر شده و ارزش برخی کوچکتر خواهد شد. بدین گونه به قسمت های ویژه ای از اطلاعات ارزشمند تر شده و برخی دیگر کم ارزش تر می شوند.

در بیشتر مجموعه داده ها، ویژگی های خاصی به طور مشخص با برچسب های خاص در ارتباط هستند. برای نمونه مساحت یا اندازه خانه برحسب متر مربع بر قیمت نهایی خانه تاثیر گذار است. شبکه های عصبی این رابطه ها را با انجام حدس هایی بر پایه ورودی ها و وزن هابه صورت کورکورانه می آموزند و سپس تخمین می زنند که تا چه اندازه برآیندها (نتیجه ها) دقیق هستند.

اما هنوز یک پرسش بزرگ مانده است: دقیقا چگونه می توان فهمید که بردارهای پارامتر (Parameter Vectors) که همان وزن همه اتصال ها در شبکه عصبی هستند، چه باید باشند؟ به یاد داشته باشید که

وزن ها پارامترهای شبکه عصبی هستند. این کار با فرآیندی انجام می شود که معمولا به آن آموزش یا یادگیری (Training) گفته میشود.

در حین آموزش ، تعداد زیادی نمونه آموزشی (Training Data) را به شبکه عصبی می دهیم و به طور تکراری وزن ها را تغییر می دهیم یا بهتر است بگوییم بروز می کنیم تا خطاهایی را که در نمونه های آموزش ایجاد می شود را به حداقل برسانیم. پس از نمونه های کافی ، انتظار داریم شبکه عصبی ما در حل کارهایی که برای انجام آن آموزش دیده اند کاملا مؤثر باشد.

۳-۵-۲-۱-۱ انتشار رو به جلو (Forward Propagation)

گام Forward Propagation زمانی رخ می دهد که داده های ورودی به شبکه وارد شده و تا پایان شبکه پیشمی روند تا در لایه خروجی پیش بینی شود. داده ها از لایه ورودی وارد شبکه شده و هر نورون مقداری

را محاسبه کرده و به نورون های لایه دیگر (لایه پسین) می فرستند تا اینکه در لایه خروجی برآیند شبکه که

همان برچسب است، بدست بیاید.سپس از Loss Function استفاده می کنیم تا خطای شبکه را برآورد. به گفته دیگر، تابع Loss یاتابع خطا، برای بررسی و هم سنجش (مقایسه) این است که تا چه اندازه پیش بینی بدست آمده به مقدار درست و راستین (واقعی) نزدیک است. در زمینه یادگیری ماشین، نشانه $\hat{y}$ مقداری است که مدل یا شبکه عصبی بدست آورده است ولی پاسخ درست که باید باشد، با نشانه y نمایش داده می شود. تفاوت میان این دو، اندازه نادرستی (خطا) را نشان می دهد.

۳-۵-۲-۱-۲ پس انتشار (Back Propagation)

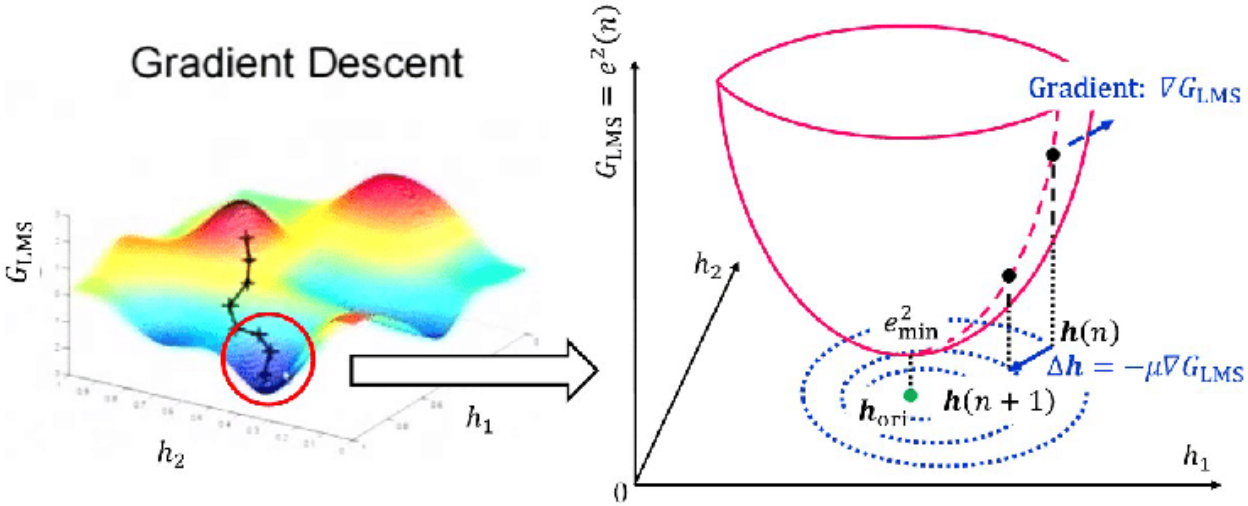
پس از آنکه ارزش نادرستی (خطای) شبکه بدست آمد، این ارزشها (یا مقدار) نادرستی، پسگرد Backward می شود و از این رو واژه را Back – Propagation برای آن به کار می بریم. اگر خروجی مطابق با یک برچسب یا کلاس یا Target درون مسئله بود، پس به پاسخ دلخواه رسیده ایم ولی اگر خروجی مطابق با برچسب نباشد، بنابراین وزن های چسبانده شده به پیوندهای (اتصال های) شبکه را باید دوباره تنظیم کنیم. برای این کار از الگوریتم گرادیان کاهشی استفاده می کنیم و وزن ها رو به روز می کنیم . بنابراین با بازگشت از پایان شبکه به آغاز شبکه Back – propagation برای بهبود کارایی شبکه و ساخت مدل درست، وزن ها بروز می شوند.

۳-۵-۲-۱-۳ الگوریتم بهینه ساز گرادیان کاهشی (Gradient Descent)

گرادیان کاهشی یکی از الگوریتم یادگیری ماشین برای بهینه سازی (Optimization) است در واقع گرادیان یک ابزار سودمند و کارآمد برای تنظیم پارامترهای مدل با انگیزه (هدف) کمینه کردن هزینه است، به ویژه اگر اطلاعات زیادی داشته باشید. به یاد داشته باشید، در شبکه های عصبی مصنوعی ، وزن های نورون ها و بایاس پارامترهای شبکه هستند. بنابراین انگیزه از گرادیان کاهشی، پیدا کردن مقدارهایی برای پارامترهای شبکه عصبی است، به گونه ای که اندازه هزینه کاهش پیدا کند. یعنی یافتن پارامترها با کمترین هزینه انگیزه گرادیان کاهشی است. گرادیان کاهشی مشتق مرتبه اول تابعی است که می خواهیم کمینه محلی تابع در نقطه محلی را پیدا کنیم. گردایان کاهشی گام هایی در راستای منفی گرادیان (یا شیب تابع) است.

۳-۵-۲-۱-٤ گرادیان کاهشی تصادفی(Stochastic Gradient Descent)

گرادیان کاهشی تصادفی (Stochastic Gradient Descent) روشی مبتنی بر تکرار برای بهینه سازی یک تابع مشتق پذیر به نام تابع هدف (تابع هزینه یا Cost Function) است که یک تقریب تصادفی از روش گرادیان کاهشی می باشد. در حقیقت گرادیان کاهشی تصادفی الگوریتمی در اختیار ما قرار می دهد که طی چند حلقه تکرار مقدار کمینه یک تابع و مقادیری را که به ازای آنها تابع کمینه (Minimum) مقدار خود را می گیرد، بدست بیاوریم. تفاوت گرادیان کاهشی تصادفی با گرادیان کاهشی استاندارد در این است که برخلاف گرادیان کاهشی استاندارد که برای بهینه سازی تابع هدف از تمام دادههای آموزشی استفاده می کند، گرادیان کاهشی تصادفی از گروهی از داده های آموزشی که به طور تصادفی انتخاب می شود برای بهینه سازی استفاده می کند. این روش در مسائل آماری و یادگیری ماشین کاربرد فراوانی دارد.



شکل ۳- ۸:نمایش هندسی_ریاضی الگوریتم بهینه ساز گرادیان کاهشی

Update rules:

$$\theta_0 := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

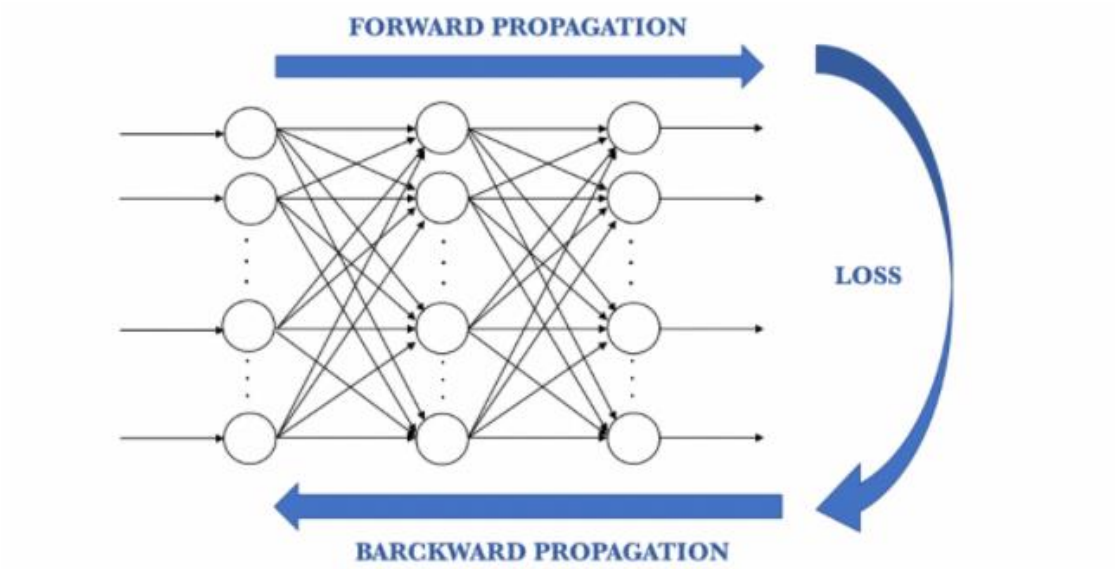
شکل ۳- ۹:فرمول به روز کردن وزن ها (پارامتر ها) در گرادیان کاهشی

Derivatives:

$$\frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)$$

$$\frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right) \cdot x^{(i)}$$

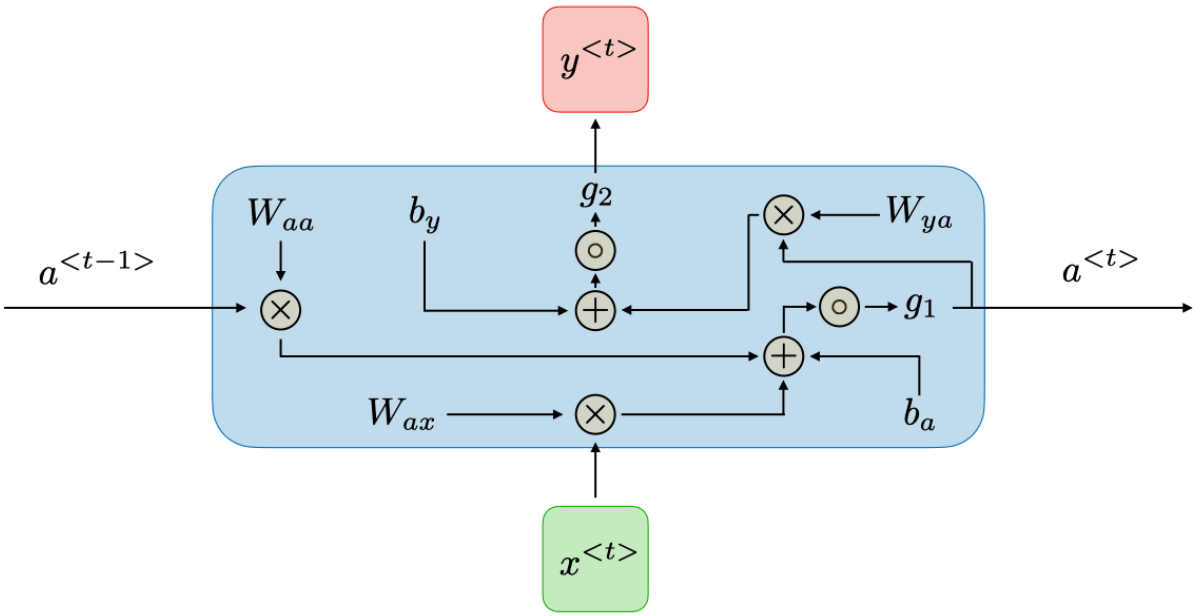
شکل ۳- ۱۰: فرمول مشتق جزئی وزن ها در گرادیان کاهش



شکل ۳- ۱۱: مکانیزم یاد گیری و بهبود پارامتر ها در شبکه ی عصبی پرسپترون چند لایه

۳-۵-۳ شبکه های عصبی بازگشتی (Recurrent Neural Network)

این گونه از شبکه ها به طور خاص برای پردازش داده های سری یا دنباله دار مفید هستند و در آن ها هر نورون قادر به حفظ حالت داخلی یا همان حافظه به منظور حفظ اطلاعات مرتبط با ورودی قبلی می باشد . این ویژگی حفظ حالت درونی یا همان حالت حافظه به شبکه کمک می کند تا قادر به فهم وکشف ارتباط بین گام های مختلف زمانی در دنباله های طولانی باشد . نام شبکه ی عصبی بازگشتی از این واقعیت استنتاج شده است که این نوع شبکه ها به صورت بازگشتی عمل می کنند ، یعنی پردازش برای تک تک المان های یک دنباله (کلمه ، صوت، جمله ، عدد و ..) انجام می پذیرد و خروجی آن وابسته به ورودی فعلی و عملیات های قبل است . این عمل از طریق تکرار یک خروجی در زمان t با ورودی شبکه در زمان t+۱ انجام می شود یعنی خروجی از مرحله ی قبل با ورودی تازه در مرحله ی جدید ترکیب می شود . این چرخه ها اجازه ی ورود اطلاعات از یک گام به گام زمانی بعدی را موجب می شوند .



شکل ۳- ۱۲:مکانیزم یک شبکه عصبی برگشتی

۳-۵-۳-۱ تشریح پارامتر های *RNN*

a<t> : به آن حالت مخفی یا hidden state گفته می شود برداری است که اندازه ی آن را ما مشخص می کنیم .این اندازاه یک هایپر پارامتر است .

ماتریس وزن Wax : ماتریس وزن مختص ارتباط بین ورودی و حالت مخفی. ماتریسی به ابعاد اندازه ی داده ی ورودی و اندازه ی a<t> .

ماتریس وزن Waa : ماتریس وزن مختص ارتباط حالت پنهان قبلی (a<t-۱>) و حالت پنهان فعلی (a<t>) که دارای ابعاد (a<t-۱>,a<t>) می باشد .

ماتریس وزن Wya : ماتریس وزن مختص ارتباط بین حالت مخفی و خروجی (y^) می باشد و داراری ابعاد (a<t>,$\hat{y}$) می باشد .

بردار ba : بایاس مربوط به محاسبه ی a<t> .این بردار اندازه ای برابر با اندازه ی a<t> دارد .

بردار by : بایاس مربوط به پیش بینی گام های زمانی ($\hat{y}$) .این بردار اندازه ای برار با خروجی ($\hat{y}$) دارد .

انتشار رو به جلو در رRNN (Forward Propagation)

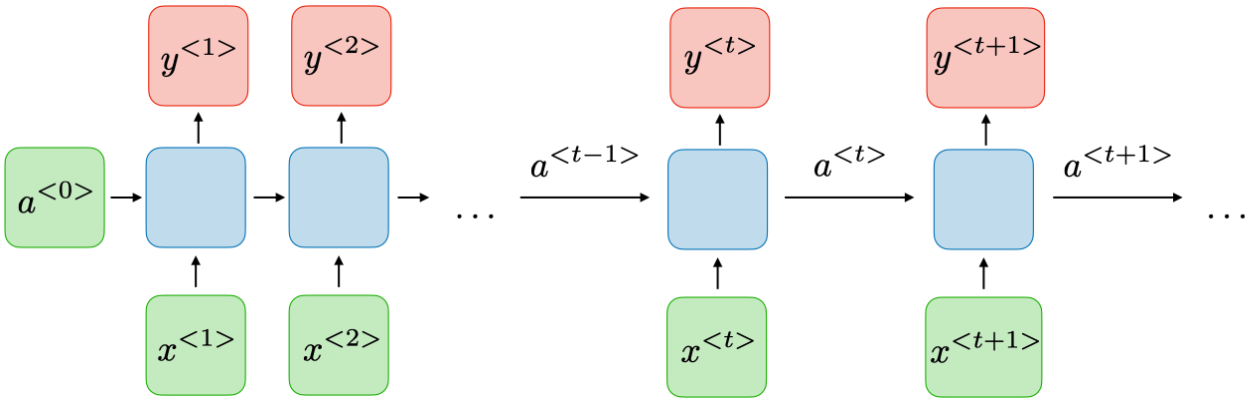
در هر گام زمانی (timestep) ، a<t> و y<t> به صورت زیر محاسبه می شوند :

$$a<t>=g۱(Waa<t-۱>+Waxx<t>+ba)$$

$$y<t>=g۲(Wya<t>+by)$$

که Waa, Wax, Way, Way, ba, by شامل ضرایب می شوند که در طول گام های زمانی به اشتراک گذاشته می شوند . g۱ و g۲ هم به ترتیب توابع فعالساز سیگموید و سافتمکس هستند .

ما در هر گام زمانی به دنبال تقریب زدن یک توزیع احتمالی بر روی تمامی رخدادهای ممکن بعدی به ازای اطلاعات دریافت شده از گام های زمانی قبلی هستیم .



شکل ۳- ۱۳:نمایی از چند گام شبکه ی RNN

بر خلاف شبکه ی عصبی که تا الان خواندیم که از پارامتر های مختلف در هر لایه بهره می برند ، در یک شبکه ی RNN از پارامتر های یکسان در تمامی گام ها بهره مند می شود . یعنی هر لایه ی RNN تنها دارای تعداد معینی پارامتر است که در تمامی گام های زمانی از آن ها به صورت اشتراکی استفاده می شود . ما در هرگام تنها ورودی متفاوت داشته و عملیات یکسان بر روی ورودی های مختلف صورت می گیرد . در نتیجه این مسئله باعث کاهش قابل ملاحظه ی تعداد پارامتر های مورد نیاز جهت یادگیری می شود .

۳-۵-۳-۲ تابع ضرر در RNN (Loss Function)

تابع ضرر RNN به صورت زیر محاسبه می شود :

$$\ell^{<t>}(y^{\wedge}, y) = -y^{<T>} log y^{\wedge<T>} - (۱ - y^{<t>}) \log (۱ - y^{\wedge<t>})$$

$$\ell(y^{\wedge}, y) = \sum_{i=1}^{Tx} \ell^{<t>}(y^{\wedge<t>}, y^{<t>})$$

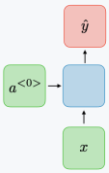
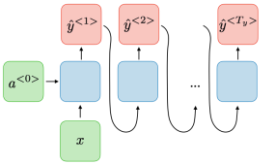
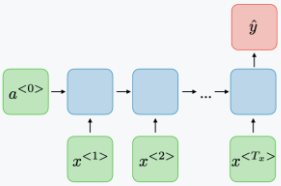
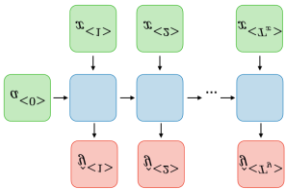
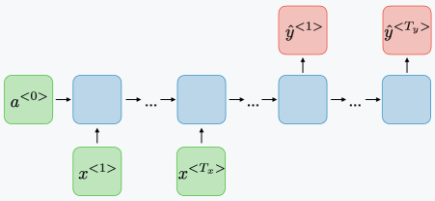
۳-۵-۳-۳ پس انتشار خطا در RNN (Back Propagation Error)

برای انجام پس انتشار ما از گام زمانی اخر شروع کرده خطا را محاسبه میکنیم و سپس در یک حلقه محاسبات را به ازای تمامی گام های زمانی ادامه میدهیم. برای محاسبه ی گرادیان ها به صورت زیر عمل می کنیم :

- محاسبه ی خطا در هر گام زمانی ($\ell^{<t>}(y^{\wedge}, y)$)
- محاسبه ی گرادیان عبارت خروجی در هر گام زمانی (y^<t>)
- محاسبه ی گرادیان عبارت حالت مخفی جدید در هر گام زمانی (a<t>)

۳-۵-۳-٤ انواع مختلف ساختار RNN با توجه به کاربرد آن ها

ممکن است تعداد خروجی ها (y^) با تعداد ورودی های (x) در یک شبکه ی عصبی برابر نباشد. این مسئله مربوط به کاربرد مورد نظر ما می باشد . به عنوان مثال زمانی که قصد تحلیل احساسات بر روی یک جمله را داشته باشیم ما تنها به یک خروجی نیاز داریم و نیازی به بودن خروجی بد از هر ورودی نیست به همین ترتیب ممکن است ما نیاز به خروجی در همه ی گام ها نداشته باشیم . در جول زیر انواع ساختار های RNN را می توانیم مشاهده کنیم .

Type of RNN	Illustration	Example
One-to-one Tx =Ty		Traditional neural network
One-to-many Tx=١ , Ty=١		Music generation
Many-to-one Tx>١, Ty=		Sentiment classification
Many-to-many *Tx=Ty*		Name entity recognition
Many-to-many *Tx≠Ty*		Machine translation

جدول ۳- ۱:انواع مختلف RNN با توجه به کاربرد آن ها

۳-۵-۳-۵ محو شدن و منفجر شدن گرادیان(*Vanishing and Exploding gradient*)

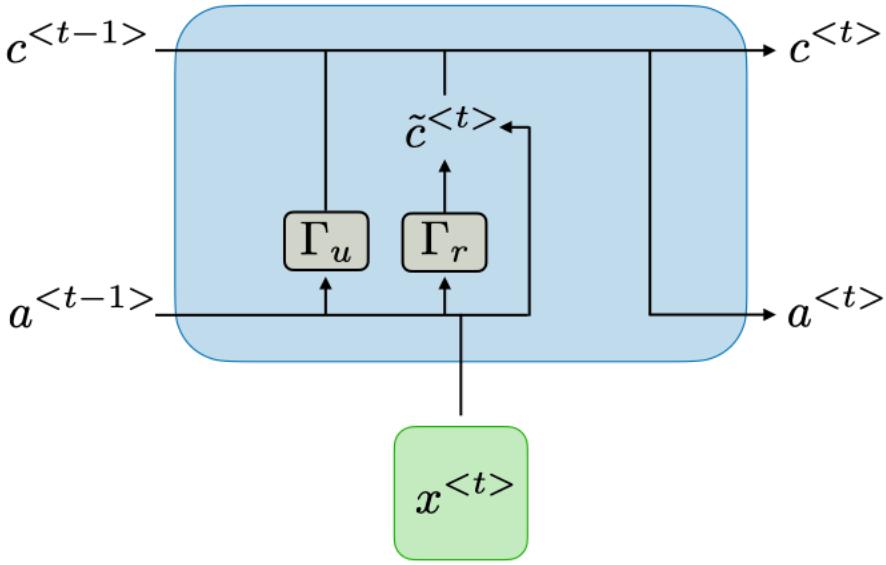
الگوریتم RNN ساده که تا الان خواندیم برای نمونه هایی که وابستگی های طولانی (long dependencies) دارند مناست نیست . در شبکه های عصبی عمیق ،هنگام پس انتشار به دلیل عمق زیاد ، گرادیان از خروجی انتشار بسیار سختی برای تاثیر گذاری بر وزن های لایه های اولیه دارد تا بر محاسبات موجود در لایه های اولیه اثر بگذارد ، برای شبکه های RNN نیز به همین منوال است . شبکه های RNN ساده دارای نفوذ محلی هستند (local influence) به این معنی که خروجی فقط از ورودی هایی که در نزدیکی ان هستند تاثیر می پذیرد و برای گام های زمانی جلوتر بسیار سخت است که از گام های زمانی عقب تر تاثیر بگیرند . و ممکن است گرادیان کاهشی در محاسبات دچار خطا شود و در پس انتشار (Back Propagation) به وزن های نورون های لایه های اول مقادیر بسیار کوچکی که به صفر میل می کند تخصیص دهد و این عمل به خاطر تکرار شدن گرادیان در طول تعداد زیادی از لایه های شبکه رخ می دهد که به آن محو شدگی گرادیان (vanishing) گفته می شود و هرچه تعداد لایه های شبکه بیشتر باشد در نتیجه احتمال محو شدگی نیز بیشتر است .

همان گونه که vanishing برای RNN صدق می کند exploding نیزهم صادق است و باعث می شود مقادیر پارامتر ها خیلی بزرگ شوند و الگوریتم دچار اشتباه شود . تناوب های نامنظم مقادیر تابع ضرر حاکی از exploding می باشد .برای حل مشکلexploding و vanishing معمول ترین راه حل استفاده از روش gradient clipping می باشد که به این صورت عمل می کند که بردار های گرادیان را در نظر می گیرد و اگر ان ها از یک آستانه ای بیشتر بودند آن ها را rescale می کند . استفاده از مقدار دهی اولیه ی مناسب وزن ها می تواند در جلوگیری از محو شدگی گرادیان تا حدودی موثر باشد اما

کافی نیست به منظور فائق امدن بر این مشکل از دیگر الگوریتم های RNN نظر GRU و LSTM اسفاده می شود که در ادامه به شرح ان ها می پردازیم .

۳-۵-٤ Gated Recurrent Unit (GRU)

یکی از راه های جلوگیری از محو شدن گرادیان استفاده از الگوریتم GRU می باشد .این نوع شبکه از مفهوم هایی به نام دروازه ی به روزرسانی (update gate) و دروازه ی بازنشانی (reset gate) استفاده می کند .این دو گیت در واقع دو بردار هستند که بااستفاده از ان ها تصمیم گرفته می شود چه اطلاعاتی به خروجی منتقل شده و چه اطلاعاتی منتقل نشود .نکته ی خاص در مورد این گیت ها این است که می توان طوری ان ها را اموزش داد تا اطلاعات مربوط به گام های زمانی بسیار قبل را بدون این که در طی گام های زمانی دستخوش تغییر شود ، حفظ کند . یکی از مشکلات شبکه های عصبی بازگشتی ساده عدم توانایی ان ها در استفاده از دنباله های طولانی است. به عنوان مثال در جمله ی " دانش اموزان منتخب ایران به مرحله ی نهایی المپیاد ریاضی رسیده و به مقام نخست دست یافتند " شبکه برای این که بتواند تسخیص صحیحی در انتخاب فعل نهایی جمله داشته باشد نیاز مند به خاطر سپردن دانش اموزان در ابتدای جمله می باشد وشبکه ی RNN ساده مکانیزمی برای انجام این مهم ندارد و در بهترین حالت چند گام محدود محلی دارای قدرت تاثیر گذاری در انتخاب های شبکه اند چراکه در RNN ورودی گام فعلی همیشه با یک مقدار جدید از از ورودی گام قبل جایگزین می شود .



شکل ۳- ۱۴: مکانیزم GRU

$$c^{\sim<t>} = \tanh\left(Wc[\Gamma r * c^{<t-۱>}, x^{<t>}] + bc\right)$$

$$\Gamma u = \sigma(Wu[c^{<t-۱>}, x^{<t>}] + bu)$$

$$\Gamma r = \sigma(Wr[c^{<t-۱>}, x^{<t>}] + br)$$

$$c^{<t>} = \Gamma u * c^{\sim<t>} + (۱ - \Gamma u) * c^{<t-۱>}$$

$$a^{<t>} = c^{\sim<t>}$$

در GRU برای حل این مشکل از update gate که به گاما Γu نشان داده می شود ، استفاده می شود. اگر به فرمول Γu نگاهی بیندازیم متوجه تابع سیگموید در آن خواهید شد این به این معناست که مقادیر آن بین صفر و یک می باشند (تقریبا صفر و تقریبا یک) .

این دروازه اصطلاحا سوییچی است که مشخص می کند در یک گام زمانی حالت قبلی مورد استفاده قرار گیرد ($c^{<t-۱>}$) و یا ورودی که به گام زمانی داده شده است مورد محاسبه قرار گیرد $c^{\sim<t>}$ و یا ترکیبی از هر دو . با استفاده از این قابلیت شبکه قادر است در دنباله های طولانی به راحتی یک حالت از چندین

گام زمان قبل را در چندین گام زمانی بعد اثر بدهد . به طور دقیق تر شبکه قادر است با میل دادن Wu به سمت صفر، Γu را هم به صفر میل دهد و این گونه $c^{\sim<t>}$ را بی اثر نموده و از $c^{<t-۱>}$ برای محاسبه ی حالت فعلی گام زمانی $c^{<t>}$ و پیش بینی نتیجه ی گام فعلی($y^{\wedge<t>}$) استفاده کند .

در این شبکه همچنین مکانیزمی برای کنترل میزان اطلاعات از گام قبل تعبیه شده است که با استفاده دروازه ی بازنشانی (reset gate) این عمل را نجام می دهیم که آن را با Γr نشان می دهند که مقدار ان عددی است بین ۰ و ۱ .دروازه ی بازنشانی یا همان reset gate در اصل مانند سوییچی عمل می کند که شبکه با کمک ان می تواند مشخص کند چه میزان از اطلاعات گذشته $c^{<t-۱t>}$ در گام فعلی $c^{<t>}$ مورد نیاز است . به طور دقیق تر با صفر بودن($\Gamma r = ۰$) این سوییچ ، از گام قبلی هم اطلاعاتی وارد نمی شود و در محاسبات دخالت داده نمی شود و به این صورت شبکه را قادر به فراموش کردن حالت محاسبه شده ی قبلی می کند . و هر چه بیشتر به سمت$\Gamma r = ۱$ پیش رویم سهم بیشتری اطلاعات حالت قبل در محاسبات حالت جدید دخالت داده می شوند .به کمک این دو دروازه الگوریتم GRU به راحتی می تواند نسبت به ذخیره سازی و یا فیلتر کردن اطلاعات از گام های زمانی قبلی اقدام کرده و از دنباله های طولانی که شبکه های عصبی بازگشتی عادی در برخورد با آن ها دچار مشکل می شوند بیشترین استفاده را ببرد .

اگر به فرمول ها نگاه کنیم متوجه می شویم $c^{\sim<t>}$ همان $a^{<t>}$ در RNN ساده است و به عنوان یک جایگزین برای $c^{<t>}$ تلقی می شود .

۳-۵-۵ Long Short Term Memory(LSTM)

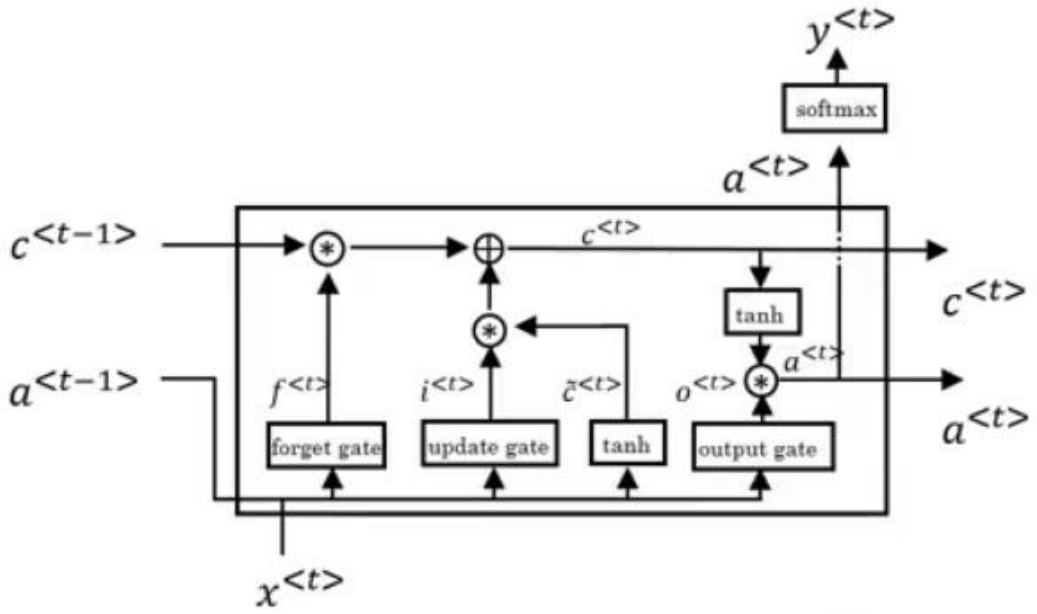
برخلاف شبکه های عصبی بازگشتی سنتی که در آن محتوا در هر گام زمانی از نو بازنویسی می شد و LSTM شبکه قادر است نسبت به حفظ حافظه ی فعلی از طریق دروازه های معرفی شده تصمیم گیری

کند . به طور شهودی اگر LSTM ویژگی مهمی را در دنباله ی ورودی در گام های ابتدایی تشخیص دهد به سادگی می تواند این اطلاعات را طی مسیر طولانی منتقل کند.

LSTM نسبت به GRU از ساختار پیچیده تری بر خوردار است .در این شبکه سه گیت وجود دارد که از طریق آن شبکه نسبت به کنترل جریان اطلاعات درون خود اقدام می کند.

علاوه بر این سه گیت یک سلول حافظه نیز وجود دارد (memory cell) یا به اختصار C .

همچنین شبکه دارای یک ورودی از حافظه ی پنهان $a^{<t-1>}$ و یک ورودی x می باشد و سه خروجی تولید می کند $a^{<t>}$ و $c^{<t>}$ و $y^{\wedge<t>}$. تمامی گیت ها دارای ابعاد یکسان با بردار حافظه ی مخفی (حالت مخفی)$a^{<t>}$ می باشند و همان طور که معلوم است بر خلاف GRU، $a^{<t>}$ برابر $c^{<t>}$ نمی باشد و کاربرد های متفاوتی دارند .



شکل ۳- ۱۵: یک واحد LSTM

$$c^{\sim<t>} = \tanh\,(Wc[a^{<t-۱>}, x^{<t>}] + bc)$$

$$\Gamma u = \sigma(Wu[a^{<t-۱>}, x^{<t>}] + bu)$$

$$\Gamma f = \sigma(Wf[a^{<t-۱>}, x^{<t>}] + bf)$$

$$\Gamma o = \sigma(Wo[a^{<t-۱>}, x^{<t>}] + bo)$$

$$c^{<t>} = \Gamma u * c^{\sim<t>} + \Gamma f * c^{<t-۱>}$$

$$a^{<t>} = \Gamma o * \tanh\,(c^{<t>})$$

۳-٦ کاربرد شبکه های عصبی مصنوعی

امروز به قدری استفاده از سیستم‌های هوشمند و به ویژه شبکه عصبی مصنوعی گسترده شده است که می توان این ابزارها را در ردیف عملیات پایه ریاضی و به عنوان ابزارهای عمومی و مشترک، طبقه‌بندی کرد. چرا که کمتر رشته دانشگاهی است که نیازی به تحلیل، تصمیم‌گیری، تخمین، پیش‌بینی، طراحی و ساخت داشته باشد و در آن از موضوع شبکه‌های عصبی استفاده نشده باشد.شبکه های عصبی مصنوعی در طبقه یندی اسناد و اطلاعات در شبکه های کامپیوتری ،طراحی و بهینه سازی سیستم های فنی مهندسی ،تصمیم گیری بهینه در پروژه های مهندسی ،مدل سازی پدیده های فیزیکی پیچیده ،پیش بینی عمل های جراحی و نتایج درمان ،پیش بینی قیمت سهام و شاخص بورس ،تحلیل و ارزیابی ریسک،هدف گیری و تعقیب در سلاح های موشکی و ... کاربرد دارند .

"فصل چهارم"

"اجرای الگوریتم ها روی دیتا ،مقایسه ی الگوریتم ها و نتیجه گیری "

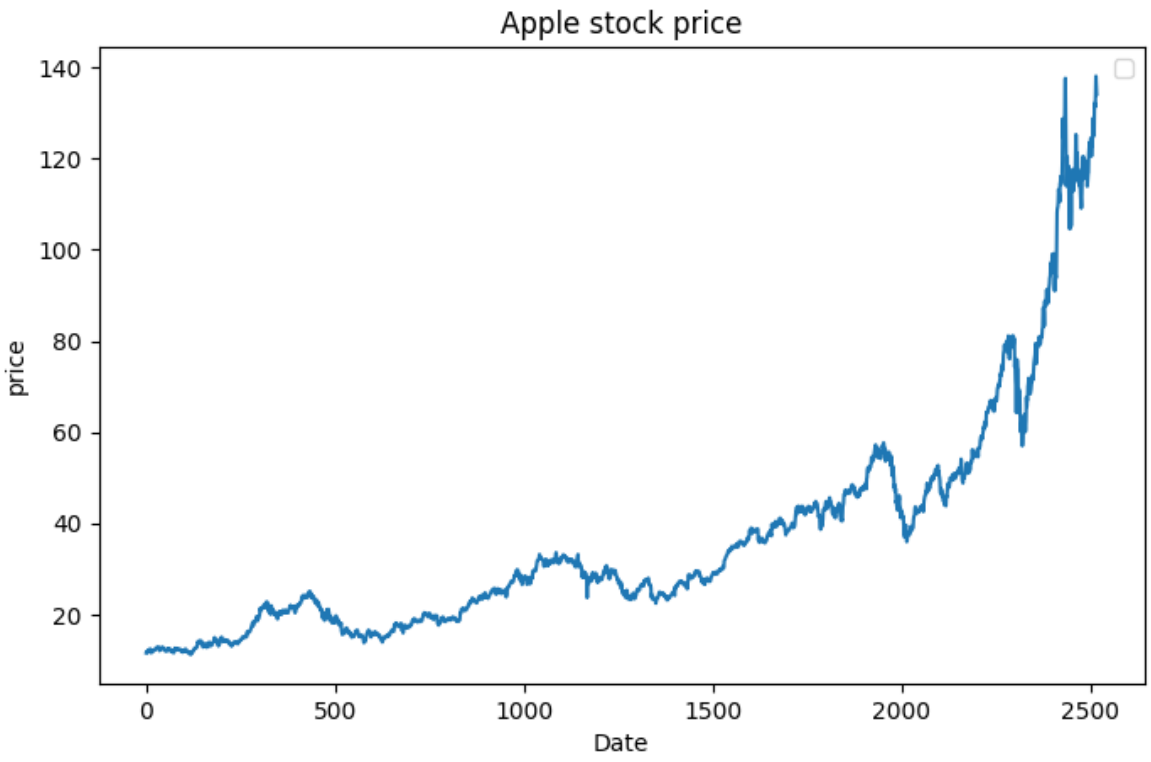
٤-١ مقدمه

در این فصل هر کدام از الگوریتم های معرفی شده در فصل ۲و۳ را روی دیتای سهام شرکت اپل آموزش می دهیم و بعد از ساختن شصت روز اخر دیتا را به وسیله ی الگوریتم های مختلف پیش بینی می کنیم . برای سنجش دقت الگوریتم هم از معیار مجذور میانگین مربعات خطا RMSE استفاده می کنیم .

٤-۲ پیاده سازی الگوریتم های سری زمانی بر روی دیتا

٤-۲-۱ بررسی ساختار سری زمانی

اولین قدم بررسی ساختار دیتای مورد نظر است تا یک دید کلی از سری زمانی به دست آوریم .برای این منظور نمودار سری زمانی مورد نظر را ترسیم می کنیم .



شکل ۴_ ۱:نمودار سری زمانی سهام شرکت اپل

با یک نگاه اجمالی متوجه می شویم که سری زمانی ما دارای روند صعودی می باشد و ایستا نیست و نیاز به روند زدایی دارد . نا ایستا بودن این سری زمانی را جلو تر با **استفاده** از آزمون دیکی فولر ثابت خواهیم کرد .

٤-٢-٢ آزمون دیکی فولر بر روی دیتا

حال با استفاده از آزمون دیکی فولر ایستایی سری زمانی مورد نظر را بررسی می کنیم .

خروجی آزمون به صورت زیر خواهد بود :

```
Augmented Dickey-Fuller Test on " Open"
-----------------------------------------------
Null Hypothesis: Data has unit root. Non-Stationary.
Significance Level    = ٠,٠٥
Test Statistic        = ٢,٩٩٢١
No. Lags Chosen       = ٢٦
Critical value ١٪     = -٣,٤٣٣
Critical value ٥٪     = -٢,٨٦٣
Critical value ١٠٪    = -٢,٥٦٧
=> P-Value = ١,٠. Weak evidence to reject the Null Hypothesis.
=> Series is Non-Stationary.
```

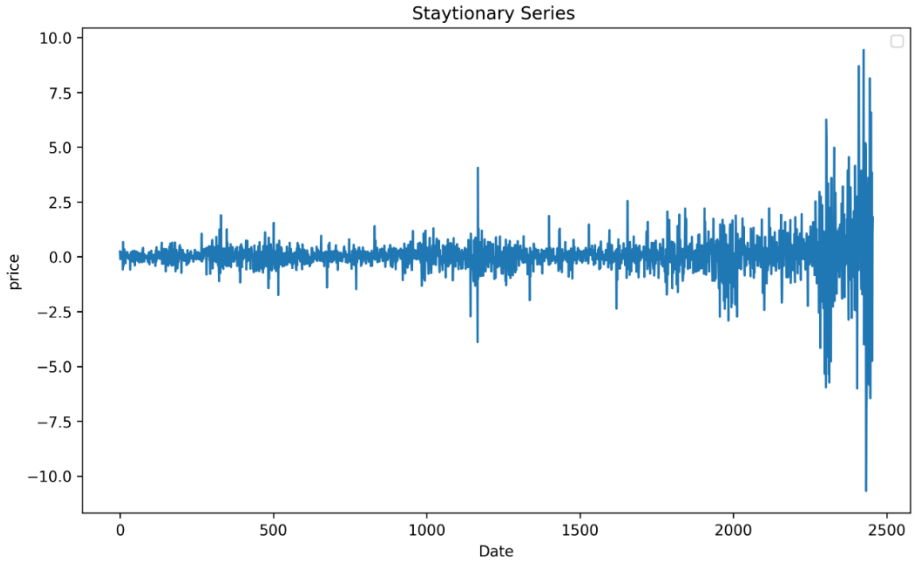
همان طور که مشاهده می کنید فرض صفر ما رد نشد پس سری زمانی ما ایستا نیست و ما نیاز داریم که با استفاده از تفاضل گیری سری زمانی را ایستا کنیم .

٤-٢-٣ تفاضل گیری دیتا

حال با استفاده از نرم افزار پایتون تفاضل گیری را روی دیتا انجام میدهیم و یک بار دیگر با استفاده از آزمون دیکی فولر ایستایی دیتا را بررسی میکنیم و در آخر نگاهی به نمودار دیتای تفاضل گیری شده می اندازیم .

```
Augmented Dickey-Fuller Test on "٠"
-----------------------------------------------
Null Hypothesis: Data has unit root. Non-Stationary.
Significance Level    = ٠,٠٥
Test Statistic        = -٩,٧٩٦٥
No. Lags Chosen       = ٢٥
Critical value ١%     = -٣,٤٣٣
Critical value ٥%     = -٢,٨٦٣
Critical value ١٠%    = -٢,٥٦٧
=> P-Value = ٠,٠. Rejecting Null Hypothesis.
=> Series is Stationary.
```

همان طور که ملاحظه می کنید بعد از یک مرتبه تفاضل گیری سری زمانی به صورت ایستا در می آید.

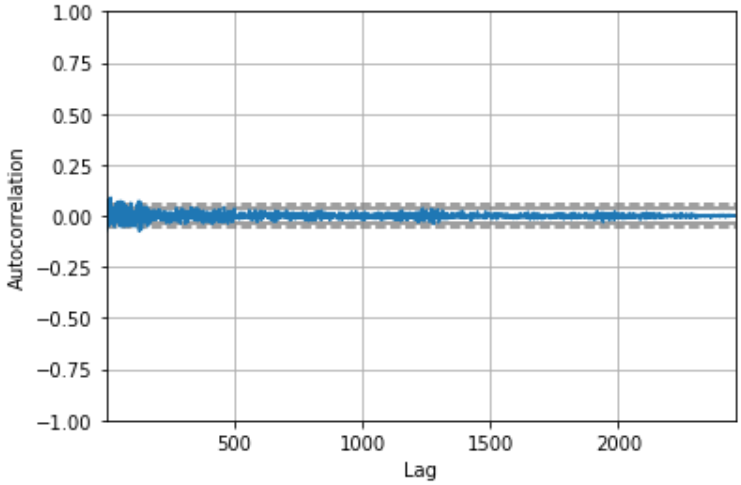


شکل ٤ 2_: نمودار سری زمانی ایستا شده

همان طور که مشهود است پس از تفاضل گیری روند صعودری در دیتا حذف شد و سری زمانی ما ایستا شد .

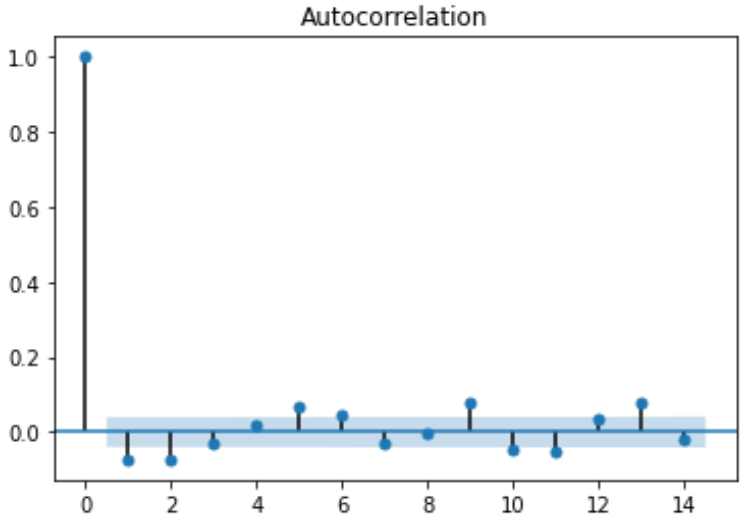
٤-٢-٣ نمودار های ACFو PACF روی دیتا

حال که سری زمانی ما ایستا شد می توانیم با استفاده از نمودار های خودهمبستگی و خود همبستگی جزئی حدود پارامتر های p, q را تخمین بزنیم . همچنین می شود با استفاده از نمودار خود همبستگی به ایستایی سری زمانی پی برد .

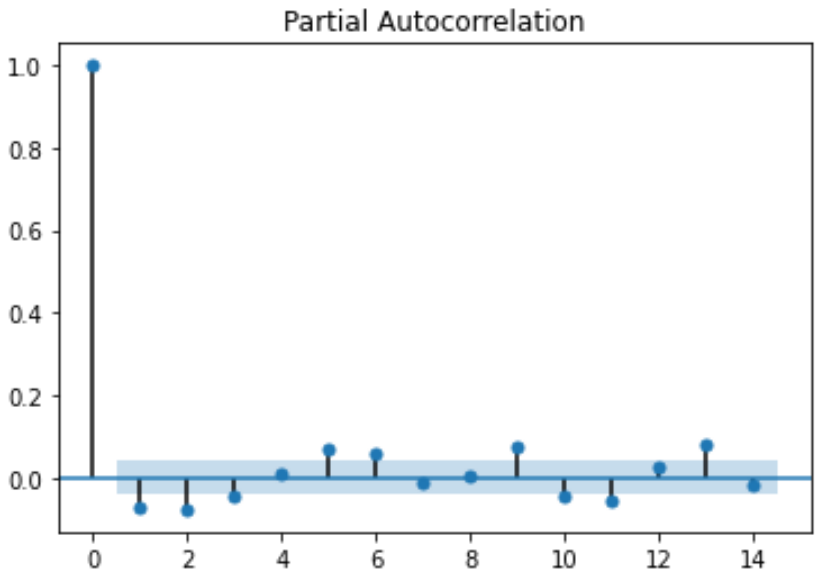


شکل ۴_۳ : نمودار خود همبستگی با lag =2500

همان طور که مشاهده می کنید نمودار سری زمانی ما بین دو خط فاصله ی اطمینان قرار گرفته که حاکی از ایستا بودن مدل ما می باشد .



شکل ۴_۴: نمودار خود همبستگی سری زمانی ایستا



شکل ۵۴ _ : نمودار خود همبستگی جزئی سری زمانی ایستا

با توجه به نمودار خود همبستگی و خود همبستگی جزئی تخمین زده میشود که مقدا ٤ برای p, و ۱ برای q می تواند متناسب باشد . حال که پارامتر هارا تخمین زدیم می توانیم مدل رو روی داده هایمان آموزش دهیم و سپس شصت روز آخر را پیش بینی کنیم . ما در نرم افزاز پایتون از تابع autoArima برای مدل سازی استفاده می کنیم که به صورت خودکار با استفاده از معیار aic پارامتر های بهینه ی مدل آریما رو تشخیص میدهد.

SARIMAX Results

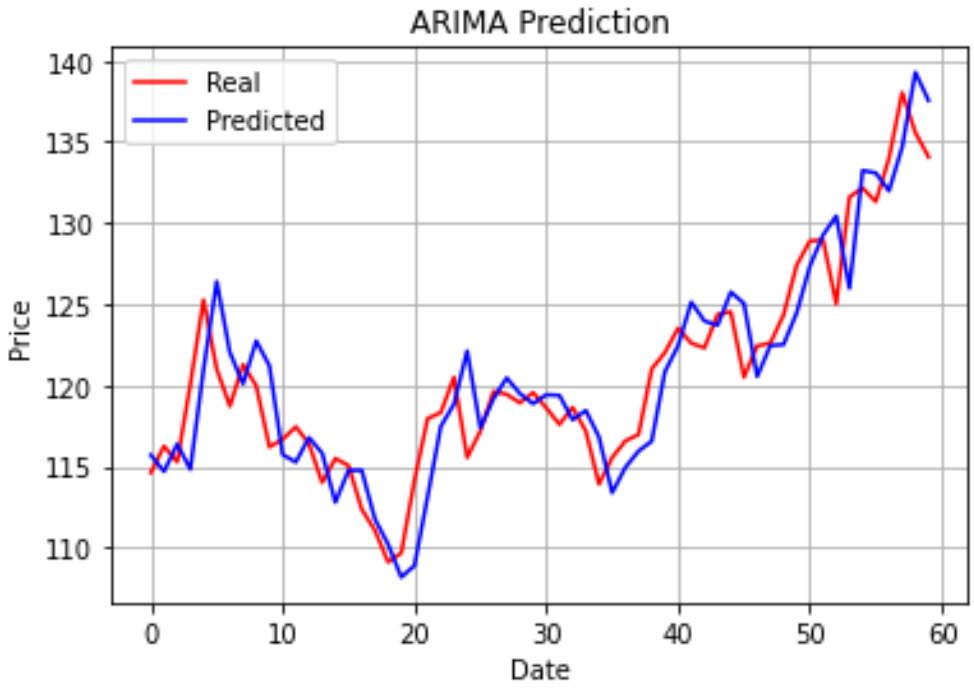
Dep. Variable:	y	**No. Observations:**	2457
Model:	SARIMAX(5, 2, 0)	**Log Likelihood**	-3452.295
Date:	Tue, 05 Jan 2021	**AIC**	6916.590
Time:	21:37:51	**BIC**	6951.426
Sample:	0	**HQIC**	6929.249
	- 2457		
Covariance Type:	opg		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
ar.L1	-0.9321	0.005	-181.359	0.000	-0.942	-0.922
ar.L2	-0.8328	0.008	-109.558	0.000	-0.848	-0.818
ar.L3	-0.6766	0.008	-80.042	0.000	-0.693	-0.660
ar.L4	-0.4655	0.008	-59.825	0.000	-0.481	-0.450
ar.L5	-0.2160	0.007	-31.499	0.000	-0.229	-0.203
sigma2	0.9743	0.007	136.966	0.000	0.960	0.988

Ljung-Box (L1) (Q):	1.57	**Jarque-Bera (JB):**	95470.96
Prob(Q):	0.21	**Prob(JB):**	0.00
Heteroskedasticity (H):	20.53	**Skew:**	0.29
Prob(H) (two-sided):	0.00	**Kurtosis:**	33.54

شکل ۴_۶:مدل پیشنهادی AutoArima

همان طور که مشاهده می کنید مدل پیشنهادی توسط نرم افزار ARIMA با پارامتر های p=5,d=2, و q=0 می باشد . اگر مدل ARIMA(5,2,0) را رو داده هایمان آموزش دهیم و سپس شصت روز آخر را پیش بینی کنیم نمودار زیر حاصل می شود.



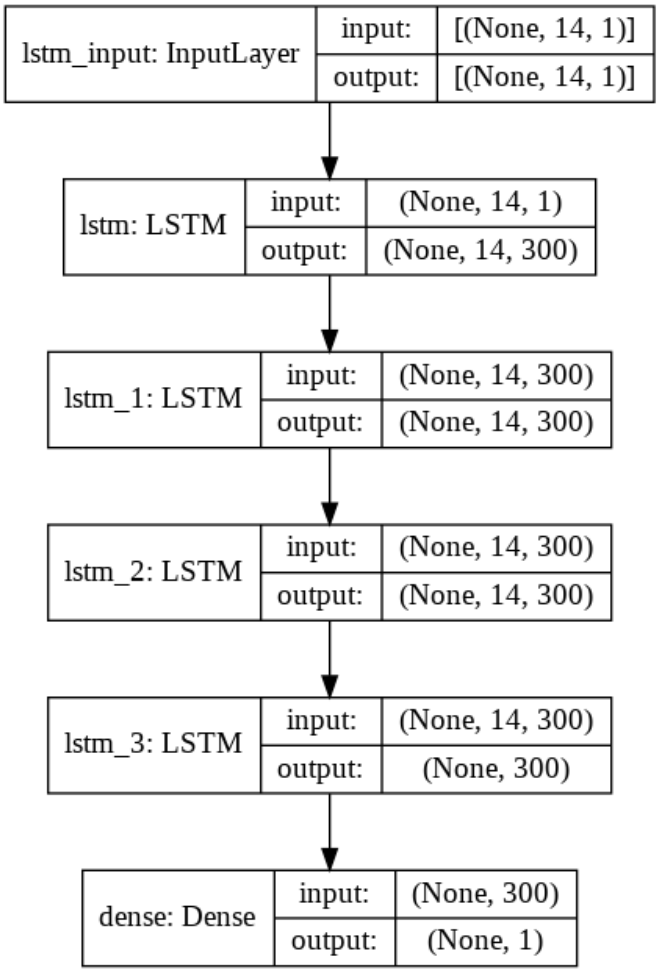
شکل ۴_۷: نمودار پیش بینی مدل ARIMA

٤-۳ پیاده سازی الگوریتم های شبکه ی عصبی بر روی دیتا

در این تحقیق ما چهار الگوریتم MLP, Simple RNN,GRU و LSTM را روی دیتا مون آموزش می دهیم و با استفاده از آن به پیش بینی شصت روز آخر قیمت سهام شرکت اپل می پردازیم.

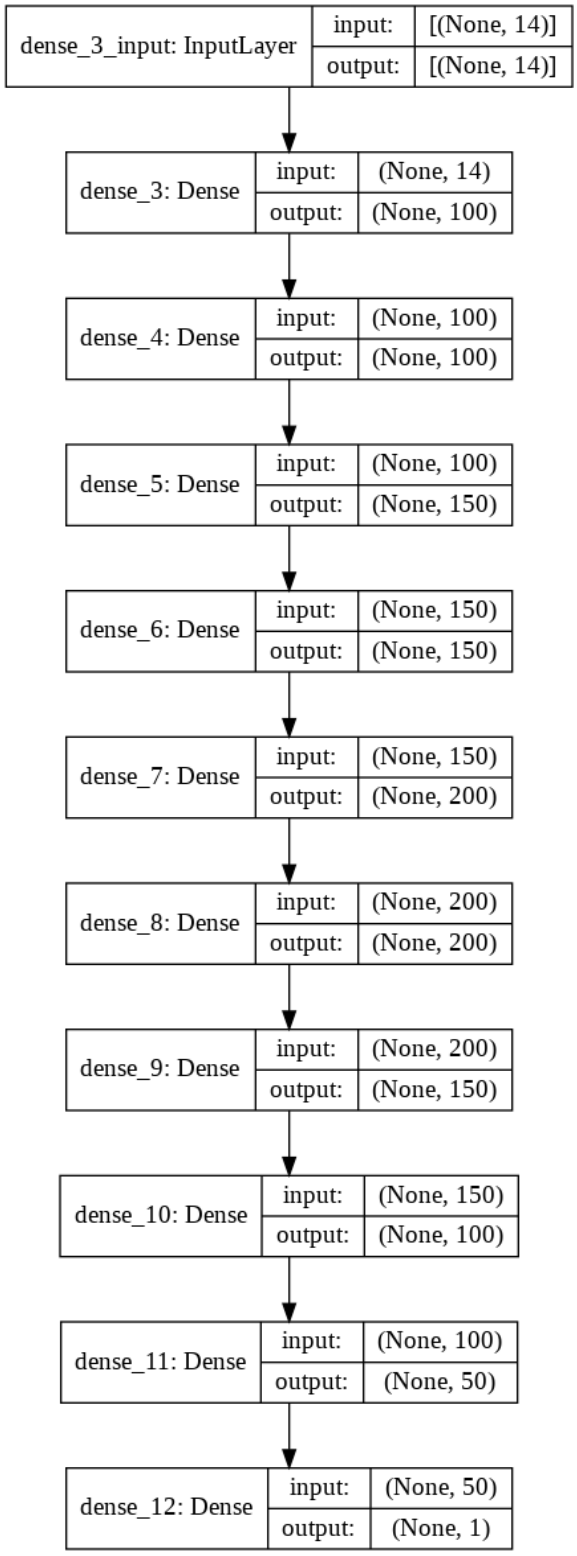
٤-٣-١ معماری شبکه های عصبی پیاده سازی شده

ما از یک ساختار یکسان برای شبکه های عصبی Simple RNN ,GRU و LSTM استفاده کرده ایم . به این صورت که هر شبکه ی عصبی شامل یک لایه ی ورودی (input layer) ،چهار لایه ی پنهان (hidden layer) که هر کدام شامل سیصد عدد unit (نورون) هستند و تابع فعالساز به کار رفته در لایه های پنهان تانژانت هایپربولیک (Tanh) می باشد . لایه ی آخر نیز(Output layer) یک یونیت دارد به این علت که مسئله سری زمانی است و ما می خواهیم که یک عدد را پیش بینی کنیم و تابع فعالساز لایه ی آخر نیز linear می باشد . به عنوان مثال شماتیک ساختار LSTM را در شکل زیر می توانید مشاهده کنید .



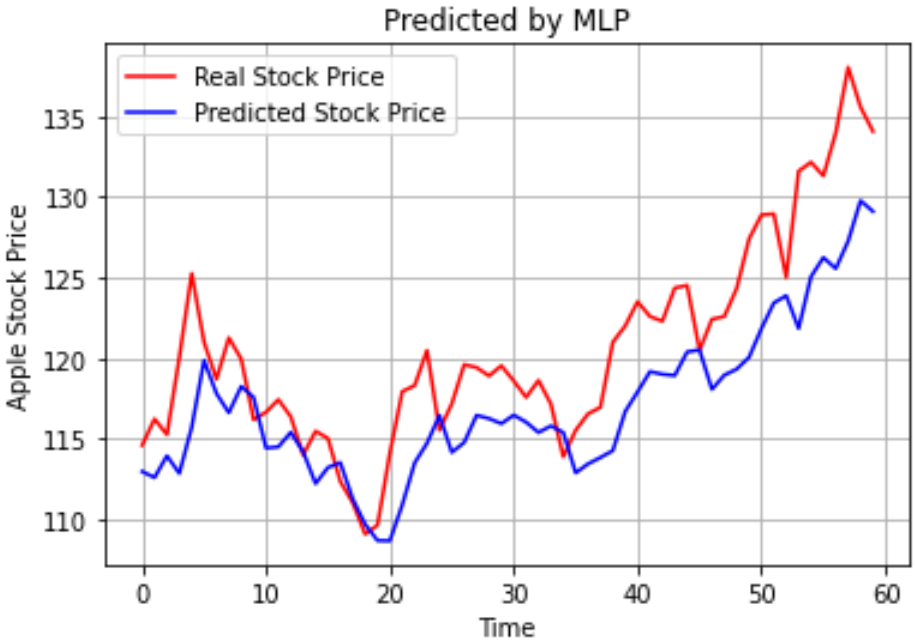
شکل ۴_۸:ساختار شبکه ی LSTM پیاده سازی شده

و همچنین MLP درای نه لایه ی پنهان می باشد که تعداد نورون ها در هر لایه متفاوت است که در شکل زیر قابل مشاهده است .

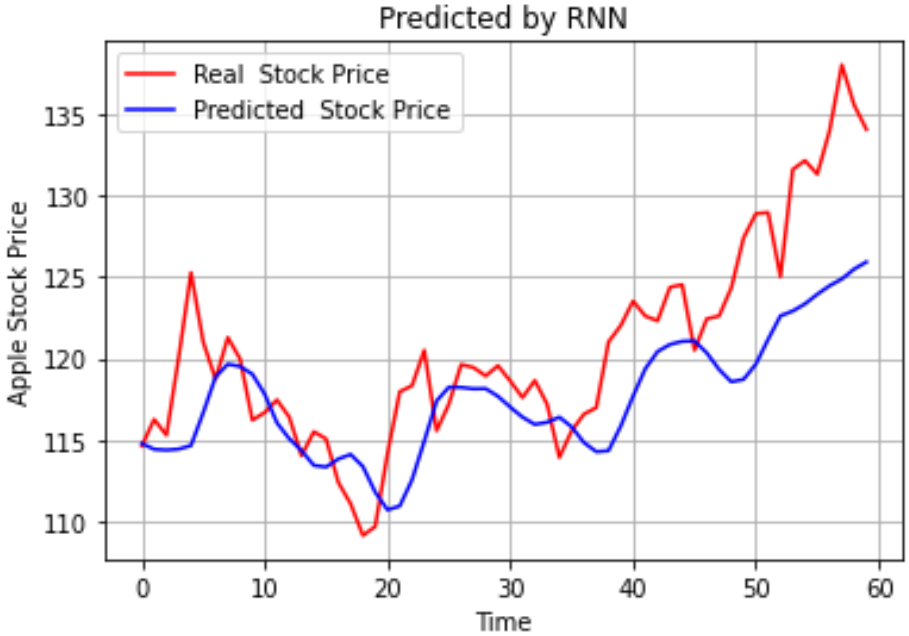


شکل ۴_۹:ساختار MLP پیاده سازی شده

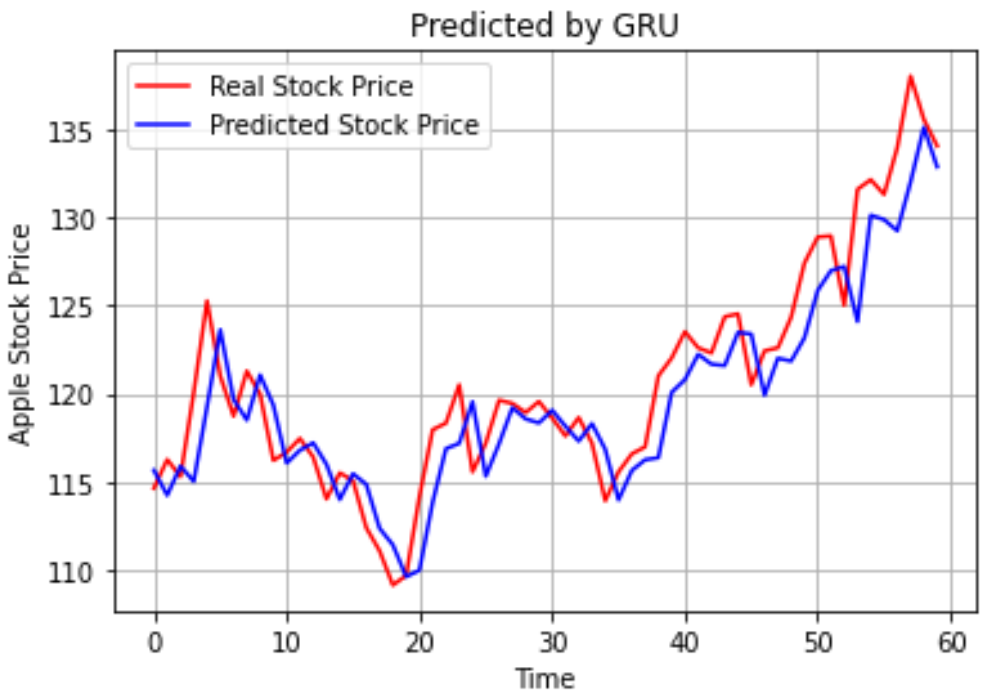
٤-٣-٢ نمودار های پیش بینی شبکه های عصبی



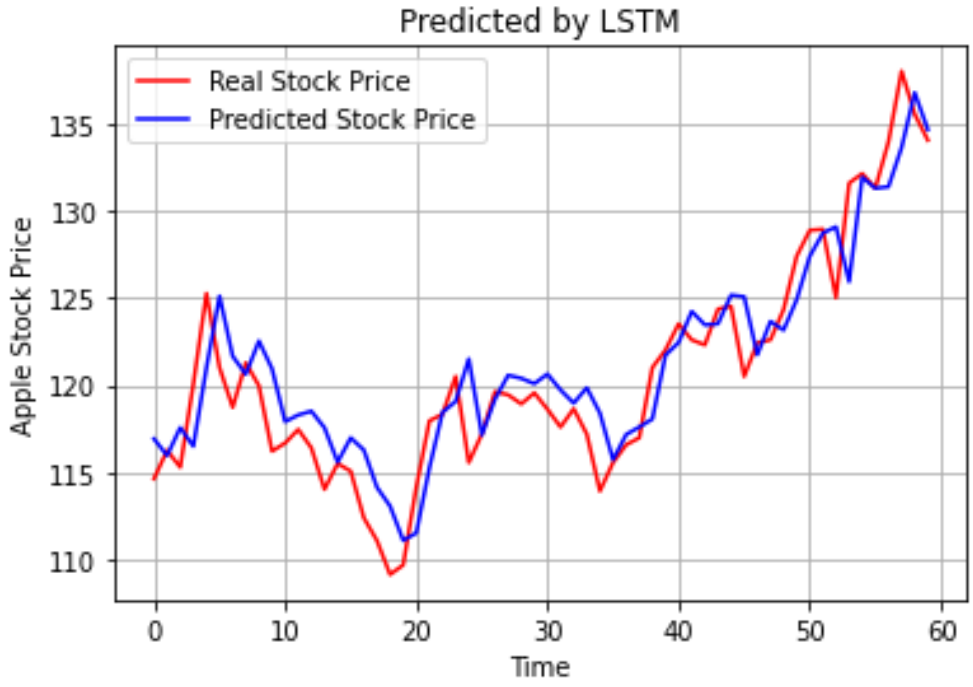
شکل ۴_ ۱۰: نمودار پیش بینی سهام توسط MLP



شکل ۴_ ۱۱: نمودار پیش بینی سهام توسط Simple RNN



شکل ۴_۱۲: نمودار پیش بینی سهام توسط GRU

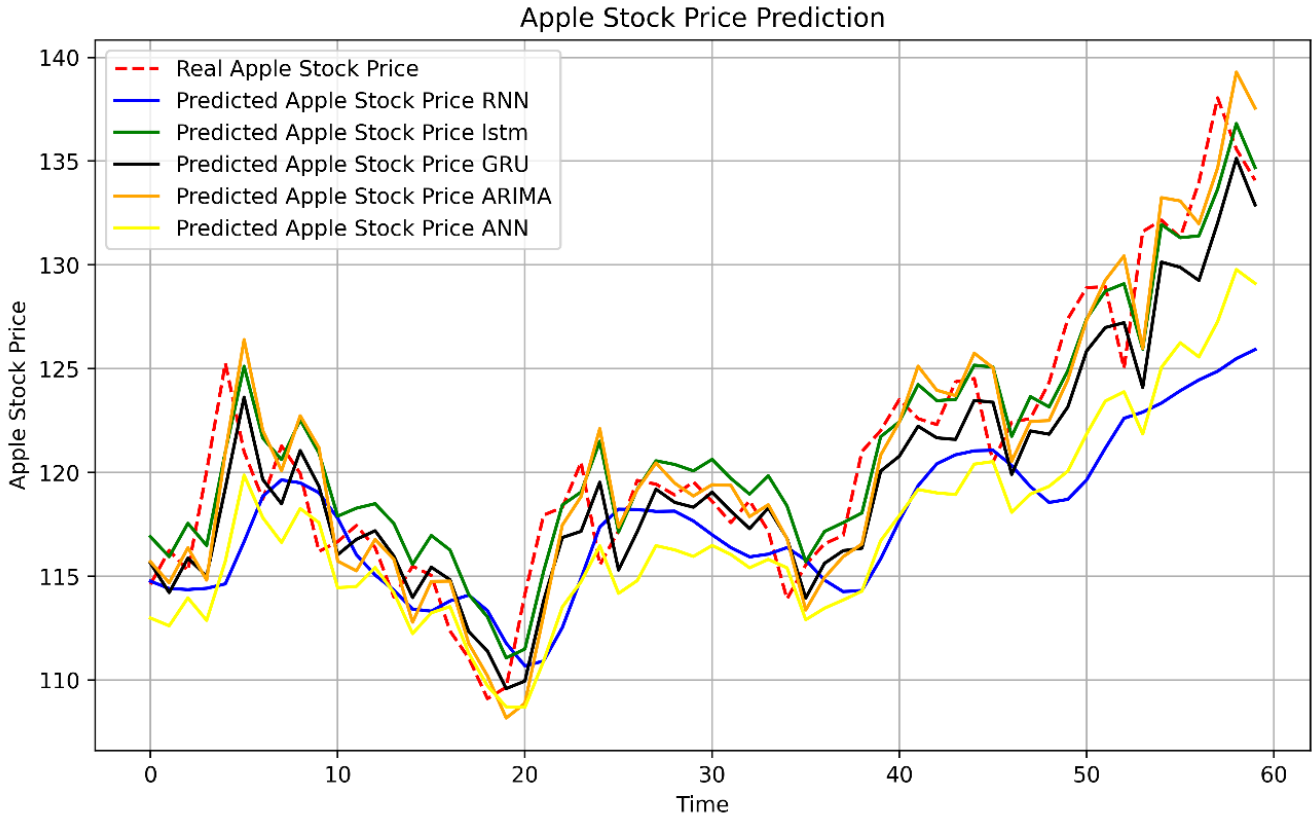


شکل ۴_۱۳: نمودار پیش بینی سهام توسط LSTM

با یک نگاه اجمالی به نمودار ها متوجه خواهیم شد که که LSTM و GRU در مقایسه با Simple RNN و MLP ، قیمت سهام را طی شصت روز آخر با دقت بالاتری پیش بینی کرده اند و همچنین MLP نسبت به Simple RNN دارای دقت بیشتری می باشد .

٤-٤ مقایسه ی عملکرد همه ی الگوریتم ها و نتیجه گیری

حال نوبت آن رسیده است که نتایج به دست آمده را باهم مقایسه کنیم. برای این کار از دو ابزار مقایسه ی نمودار های پیش بینی و مقایسه ی معیار RMSE استفاده می کنیم



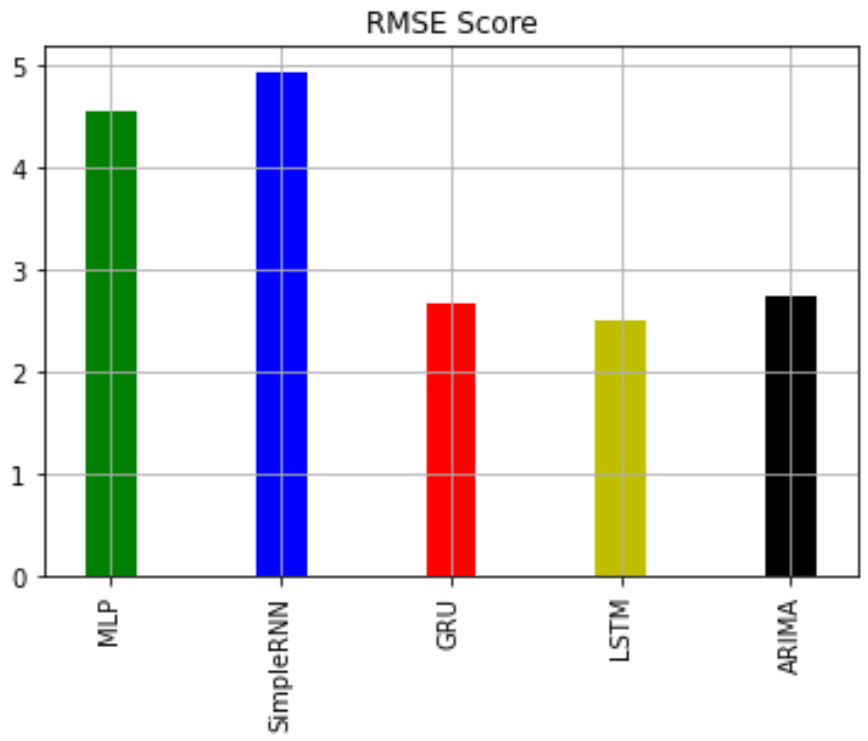
شکل ۴_۱۴: مقایسه ی نمودار های همه ی الگوریتم های پیاده سازی شده

با توجه به شکل می توان دریافت که دو الگوریتم Simple RNN و ANN دارای عملکرد و دقت ضعیف تری نسبت به سایر الگوریتم ها می باشند و همچنین می توان دریافت سه الگوریتم GRU و

LSTM و ARIMA دارای دقت خوبی بوده و با هم رقابت تنگاتنگی دارند . برای پی بردن به این که کدام الگوریتم بیشترین دقت را در پیش بینی دارد از معیار RMSE کمک می گیریم.

	RMSE Score
MLP	4.5583
SimpleRNN	4.9412
GRU	2.6707
LSTM	2.5108
ARIMA	2.7449

جدول ۴_ ۱: RMSR Score



شکل ۴_ ۱۵: نمودار ستونی مقایسه ی RMSE

با بررسی RMSE Score متوجه خواهیم شد که LSTM بیشترین دقت را داشته است و بعد از آن به ترتیب GRU و رARIMA قرار می گیرند . همان طور که از در شکل بالا هم به وضوح مشخص است Simple RNN و یعد از آن هم MLP کمترین دقت را داشته اند . در نتیجه همان طور که در ابتدای مقاله گفته شد شبکه های عصبی برگشتی از نوع LSTM و GRU می توانند در مقایسه با مدل های سری زمانی بر روی دیتا های سری زمانی عملکرد بهتری داشته باشند . به علاوه در بازار مالی امروزه قیمت سهام علاوه بر خودش به بسیاری از متغیر های دیگر همچون طلا، ارز ،شاخص کل، مسکن ،نفت ،فلزات و .... وابسته است که می توان این دسته از مسائل را در قالب Regression Classification با استفاده از انواع شیکه های عصبی حل نمود که می توان آن را پیشنهادی پیرامون پژوهش های آینده در نظر گرفت .

٤-٥ منابع و مآخذ

١. آر, بیل, و جکسون تی. ٨٦. *آشنایی با شبکههای عصبی*. تهران: موسسه انتشارات علمی دانشگاه صنعتی شریف.

٢. اعتمادی, حسین, علی اصغر انواری رستمی , و وحید احمدیان . ٩٤. "ارزیابی توان پیشبینی سود فصلی هر سهم با استفاده از مدلهای سری زمانی." *مجله مهندسی مالی و مدیریت اوراق بهادار*.

٣. البرزی, محمود, احمد یعقوب نژاد , و حسین مقصود. ٨٧. "کاربرد شبکههای عصبی مصنوعی در پیشبینی شاخص بازده نقدی و قیمت سهام." *فصلنامه مطالعات تجربی حسابداری مالی*.

٤. تقوا, محمدرضا, محمد علی حسین زاده یزدی , ابراهیم پور رجب, و سارا بابا احمدی. ٩٣. "یشبینی قیمت سهام شرکتهای بورس اوراق بهادار تهران با استفاده از شبکه های عصبی مصنوعی." *پژوهش های نوین در حسابداری*.

٥. حیدری زارع, بهزاد, و حمیدرضا کردلوئی . ٨٩. "پیشبینی قیمت سهام با استفاده از شبکه عصبی مصنوع." *فصلنامه ی مدیریت*.

٦. کیا , محسن, اصغر عارفی , و محمود باغجری . ٩٤. "قایسه پیش بینی شاخص بورس اوراق بهادار تهران با دو روش آریما و لژاندر." *دانشکده مدیریت و حسابداری دانشگاه شهید بهشتی*.

٧. لوران, فاست. ٨٨. *مبانی شبکههای عصبی مصنوعی: ساختارها، الگوریتم ها و کاربردها*. تهران: نشر نص.

٨. مشیری, سعید, و حبیب مروت . ٨٧. "پیشبینی شاخص کل بازدهی سهام تهران با استفاده از مدلهـای خطـی و غیر خطی." *فصلنامه پژوهشنامه بازرگانی*.

٩. همایون, اسدالله, حمید محمدی , و رسول کشتکار. ٨٩. "فصلنامه پژوهشها و سیاستهای اقتصادی." *ارزیابی مدلهای پیشبینی شاخصهای بازار بورس ایران*.

١٠. یار محمدی, مسعود, مهدی کلانتری, و رحیم محمودوند . ٩٥. "مقایسه ی تجربی مدل های باکس جینکینز،شبکه های عصبی مصنوعی،و تحلیل مجموعه ی مقادیر تکین در پیش بینی سری های زمانی ." *مجله ی مدل سازی پیشرفته ی ریاضی* .

ISLAMIC AZAD UNIVERSITY
North Tehran Branch

" B.SC " Thesis
Industrial Engineering

Research Title
Apple Stock Price Forecasts using Neural Networks and Timeseries Models

Advisor
Dr.Shervin Asadzadeh

By
Ali Ostadi

Autumn 2020