

آموزش استفاده از جعبه ابزار شبکه

عصبی MATLAB با ارائه یک مثال

طبقه بندی بازار بورس با استفاده از شبکه عصبی پرسپترون چند لایه

این نوشتار جنبه آموزشی دارد و با استفاده از یک مثال چگونگی استفاده از شبکه عصبی و پیاده سازی آن در متلب را نشان می‌دهد.

تهیه و تنظیم:

اسماعیل آتش‌پز گرگری

دانشجوی کارشناسی ارشد کنترل دانشگاه تهران

Email: e.atashpaz@ece.ut.ac.ir
Atashpaz.gargari@gmail.com

Web: <http://khorshid.ut.ac.ir/~e.atashpaz>

فهرست مطالب

۱-	مقدمه.....	۳
۲-	ساختار دیتا.....	۴
۳-	شبکه عصبی.....	۵
۳-۱-	مقدمه.....	۵
۳-۲-	اجزای یک شبکه عصبی.....	۷
۳-۳-	شبکه عصبی پرسپترون چند لایه:.....	۱۰
۴-	پیاده‌سازی توسط محیط MATLAB.....	۱۱
۴-۱-	جعبه ابزار شبکه عصبی متلب.....	۱۱
۴-۲-	تشکیل ساختار شبکه عصبی پرسپترون چند لایه در متلب.....	۱۱
۴-۳-	یک مثال.....	۱۲
۴-۴-	آموزش شبکه عصبی پرسپترون چند لایه در متلب.....	۱۲
۴-۵-	ارزیابی شبکه عصبی پرسپترون چند لایه در متلب.....	۱۳
۵-	برنامه نوشته شده در متلب.....	۱۴
۶-	بررسی کارایی شبکه عصبی در طبقه بندی دیتاهای بازار بورس.....	۲۶
۷-	نتیجه گیری.....	۲۷

۱- مقدمه

با توجه به توسعه صنعت در ایران، شرکت‌ها و کارخانجات صنعتی در راستای تثبیت اقتصاد تولیدی خود و با بهره‌گیری از مشارکت عمومی اقدام به فروش سهام شرکت مورد نظر کرده تا بدین طریق سرمایه‌های سرگردان را به عرصه تولید نزدیک نمایند و با این ساز و کار باعث پیشبرد پایه‌های تکنولوژی و تولید گردند.

بازار بورس تهران اخیراً داده‌های ویژگی سهام را در سایت‌های رسمی خود ارائه می‌دهد که هر سهم در هر روز کاری داراری ویژگی‌های زیر می‌باشد.

- (۱) حجم سهم
- (۲) ارزش سهم
- (۳) تعداد سهام‌داران سهم
- (۴) بیشترین قیمت سهم
- (۵) کمترین قیمت سهم
- (۶) میانگین قیمت سهم
- (۷) قیمت ابتدای سهم
- (۸) آخرین قیمت سهم
- (۹) تغییر قیمت سهم
- (۱۰) بهترین قیمت خرید
- (۱۱) بهترین قیمت فروش
- (۱۲) تعداد خریداران سهم

در این راستا با توجه به آیت‌های ذکر شده (۱۲ ویژگی) که بطور روزانه و online ارائه می شود، شاخصی به نام درجه بندی کلاس توسط فعالان بازار بدست می آید. هدف از این کلاس بندی اسن است که آیا سهم مورد نظر برای خرید در آینده چه وضعیتی را دارا می باشد.

در این پژوهش نظرات یک عضو ساده بازار به عنوان یک معیار کلاس بندی در نظر گرفته شده است. سهام دار با توجه به داده های گذشته و ارزیابی جو بازار با توجه به موارد ذکر شده تصمیم به درجه بندی سهم برای خرید آینده می کند. در شرایط معمولی کلاس بندی به صورت زیر می باشد.

شماره کلاس	ویژگی کلاس
۱	برای خرید بسیار خوب است.
۲	برای خرید بد نیست.
۳	برای خرید ریسک ندارد و یا بی تفاوت است.
۴	برای خرید ریسک بالای دارد.
۵	برای خرید خطرناک یا نا مطمئن است.

۲- ساختار دیتا

دیتاهای مورد استفاده ۴۱۶ جفت ویژگی ها و کلاس ها می باشند و داریم:

۱۲	تعداد ویژگی ها
۵	تعداد کلاس ها
۳۱۲	تعداد داده های آموزش
۱۰۴	تعداد داده های تست

همه دیتا ها در فایل MATfile با نام BurseDATAsi قرار داده شده اند. این فایل شامل دو متغیر با نامهای Train و Test میباشد.

Train: 312*13 train data

Test: 104*13 test data

ستون اول در هر یک از ماتریسها برابر عدد کلاس مربوط به ۱۲ ستون بعدی یعنی ویژگی ها است. این عددی عددی صحیح بین ۱ تا ۵ می باشد. دیتای آموزش به صورت زیر می باشد.

Train Data :

$$\begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_{312} \end{bmatrix}_{Y_{train}: 312 \times 1} \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} & x_{1,5} & x_{1,6} & x_{1,7} & x_{1,8} & x_{1,9} & x_{1,10} & x_{1,11} & x_{1,12} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} & x_{2,5} & x_{2,6} & x_{2,7} & x_{2,8} & x_{2,9} & x_{2,10} & x_{2,11} & x_{2,12} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_{312,1} & x_{312,2} & x_{312,3} & x_{312,4} & x_{312,5} & x_{312,6} & x_{312,7} & x_{312,8} & x_{312,9} & x_{312,10} & x_{312,11} & x_{312,12} \end{bmatrix}_{X_{train}: 312 \times 12}$$

ساختار دیتای تست نیز به صورت زیر می باشد.

Test Data :

$$\begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_{104} \end{bmatrix}_{Y_{test}: 104 \times 1} \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} & x_{1,5} & x_{1,6} & x_{1,7} & x_{1,8} & x_{1,9} & x_{1,10} & x_{1,11} & x_{1,12} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} & x_{2,5} & x_{2,6} & x_{2,7} & x_{2,8} & x_{2,9} & x_{2,10} & x_{2,11} & x_{2,12} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_{104,1} & x_{104,2} & x_{104,3} & x_{104,4} & x_{104,5} & x_{104,6} & x_{104,7} & x_{104,8} & x_{104,9} & x_{104,10} & x_{104,11} & x_{104,12} \end{bmatrix}_{X_{test}: 104 \times 12}$$

هدف در این پژوهش این است که با استفاده از دیتاهای آموزش، یک شبکه عصبی مصنوعی را به گونه ای آموزش دهیم که بتواند نظم موجود در طبقه بندی دیناهای بورس را تشخیص داده و بتواند در مورد دیتاهای تست با موفقیت عمل طبقه بندی را انجام دهد.

۳- شبکه عصبی

۳-۱- مقدمه

از قرن نوزدهم به طور همزمان اما جداگانه از سویی نروفیزیولوژیستها سعی کردند سیستم یادگیری و تجزیه و تحلیل مغز را کشف کنند و از سوی دیگر ریاضیدانان تلاش کردند تا مدل ریاضی بسازند که

قابلیت فراگیری و تجزیه و تحلیل عمومی مسائل را دارا باشد. اولین کوششها در شبیه سازی با استفاده از یک مدل منطقی توسط مک کلوک و والتر پیتز انجام شد که امروز بلوک اصلی سازنده‌ی اکثر شبکه های عصبی مصنوعی است. این مدل فرضیه‌هایی در مورد عملکرد نرونها ارائه می کند. عملکرد این مدل مبتنی بر جمع ورودیها و ایجاد خروجی است. چنانچه حاصل جمع ورودی‌ها از مقدار آستانه بیشتر باشد اصطلاحاً نرون برانگیخته می‌شود. نتیجه این مدل اجرای توابع ساده منطقی مثل OR و AND بود. نه تنها نروفیزیولوژیستها بلکه روانشناسان و مهندسان نیز در پیشرفت شبیه سازی شبکه‌های عصبی تأثیر داشتند. در ۱۹۵۸ شبکه پرسپترون توسط روزنبلات معرفی گردید. این شبکه نظیر واحدهای مدل شده‌ی قبلی بود. پرسپترون دارای سه لایه به همراه یک لایه وسط که به عنوان لایه پیوند شناخته شده می‌باشد است. این سیستم می‌تواند یاد بگیرد که به ورودی داده شده خروجی تصادفی متناظر را اعمال کند. سیستم دیگر مدل خطی تطبیقی نرون یا Adalaline (Adaptive linear element) می‌باشد که در ۱۹۶۰ توسط ویدرو و هاف (دانشگاه استنفورد) بوجود آمد که اولین شبکه های عصبی بکار گرفته شده در مسائل واقعی بوده‌اند. Adalaline یک دستگاه الکترونیکی بود که از اجزای ساده‌ای تشکیل شده بود و روشی که برای آموزش استفاده می‌شد با پرسپترون فرق داشت.

در ۱۹۶۹ مینسکی و پاپرت کتابی نوشتند که محدودیتهای سیستمهای تک لایه و چند لایه پرسپترون را تشریح کردند. نتیجه این کتاب پیشداوری و قطع سرمایه گذاری برای تحقیقات در زمینه شبیه سازی شبکه‌های عصبی بود. آنها با طرح اینکه طرح پرسپترون قادر به حل هیچ مساله جالبی نمی‌باشد تحقیقات در این زمینه را برای مدت چندین سال متوقف کردند.

با وجود اینکه اشتیاق عمومی و سرمایه گذاری های موجود به حداقل خود رسیده بود برخی محققان تحقیقات خود را برای ساخت ماشینهایی که توانایی حل مسائلی از قبیل تشخیص الگو را داشته باشند ادامه دادند. از جمله گراسبرگ که شبکه‌ای تحت عنوان Avalanche را برای تشخیص صحبت پیوسته و کنترل دست ربات مطرح کرد. همچنین او با همکاری کارپنتر شبکه های (Adaptive resonance

ART (theory) را بنا نهادند که با مدل‌های طبیعی تفاوت داشت. اندرسون و کوهونن نیز از اشخاصی بود که تکنیک‌هایی برای یادگیری ایجاد کردند. ورباس در ۱۹۷۴ شیوه آموزش پس انتشار خطا یا Back Propagation را ایجاد کرد که یک شبکه پرسپترون چند لایه البته با قوانین نیرومندتر آموزشی بود. پیشرفتهایی که در ۱۹۷۰ تا ۱۹۸۰ بدست آمد برای جلب توجه به شبکه‌های عصبی بسیار مهم بود. شبکه‌های عصبی مصنوعی (Artificial Neural Network) الگویی برای پردازش اطلاعات می‌باشند که با تقلید از شبکه‌های عصبی بیولوژیکی مثل مغز انسان ساخته شده‌اند. عنصر کلیدی این الگو ساختار جدید سیستم پردازش اطلاعات آن می‌باشد و از تعداد زیادی عناصر (نرون) با ارتباطات قوی داخلی که هماهنگ با هم برای حل مسائل مخصوص کار می‌کنند تشکیل شده‌اند. انسانها از زمانهای بسیار دور سعی بر آن داشتند که بیوفیزیولوژی مغز را دریابند، چون همواره مسئله هوشمندی انسان و قابلیت یادگیری، تعمیم، خلاقیت، انعطاف‌پذیری و پردازش موازی در مغز برای بشر جالب بوده و بکارگیری این قابلیت‌ها در ماشینها بسیار مطلوب می‌نمود. روشهای الگوریتمیک برای پیاده سازی این خصایص در ماشینها مناسب نمی‌باشند. در نتیجه می‌بایست روشها مبتنی بر همان مدل‌های بیولوژیکی باشد. ANN درست مثل انسانها با استفاده از مثالها آموزش می‌بیند، همانطور که یک بچه با دیدن انواع مختلف از یک حیوان قادر به تشخیص آن می‌باشد.

۳-۲- اجزای یک شبکه عصبی

نرون عصبی انسان

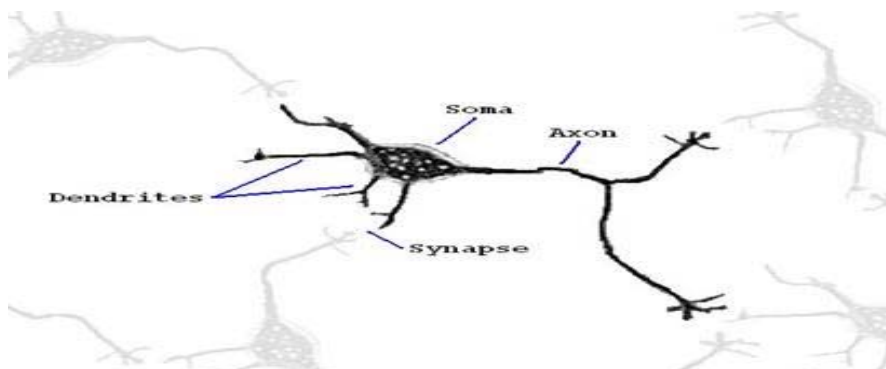
در سیستم عصبی، نرون (سلول عصبی) به عنوان اصلی ترین عنصر پردازش، شناخته شده اند. بطور کلی بدن انسان در حدود ۱۰۰ تریلیون نرون وجود دارد که تمام آن‌ها از سه قسمت اصلی تشکیل شده اند: بدنه سلول^۱، دندریت‌ها و آکسون. (شکل ۱) همان طور که در شکل مشخص است هر نرون دارای

^۱.cell body(soma)

تعدادی دندريت و يك آكسون است. دندريت ها به عنوان مناطق دريافت كننده سيگنال هاي الكتريكي هستند و سيگنال هاي الكتريكي را از آكسون نرون هاي ديگر به بدنه سلول مي برند. بدنه سلول انرژی لازم را برای فعالیت نرون فراهم کرده و بر روی سيگنال هاي ورودی عمل می کند (که با يك عمل جمع جبری ساده و مقایسه با يك سطح آستانه مدل می گردد). آكسون نیز سيگنال هاي الكتروشیمیایی را از بدنه سلول به دندريت ساير نرون ها منتقل می کند.

محل تلاقی يك آكسون از يك نرون به دندريت هاي ساير نرون ها را سيناپس می نامند. سيناپس ها واحدهای کوچکی هستند که ارتباط بين نرون ها را برقرار می سازد.

زمانیکه سيگنال هاي عصبی از آكسون ساير نرون ها به يك نرون می رسد، آن را تحريك می کند. نرون از هر يك از اتصالات ورودی خود يك ولتاژ كم را توسط سيگنال هاي عصبی، دريافت می کند و آن ها را با هم جمع می کند. اگر اين مقدار به مقدار آستانه برسد نرون آتش می گیرد و به آكسون خود يك ولتاژ خروجی ارسال می نماید و آكسون نیز با توجه به شدت آن، ممكن است يك سيگنال را توسط سيناپس، به دندريت نرون هاي ديگر بفرستد يا اينكه به دليل ضعيف بودن آن، هيچگونه سيگنالی را عبور ندهد. و به همین ترتیب تمامی فعالیت های مغزی انسان انجام می شود.



شكل ۱. ساختار يك سلول عصبی انسان

با این دید اجمالی از نحوه عملکرد نرون، باید سیستمی طراحی شود که دارای تعدادی ورودی باشد که با توجه به اهمیت هر یک، آن‌ها را با یکدیگر جمع ساده جبری نماید و توسط یک تابع موسوم به تابع تبدیل، آن‌ها را به نورون‌های دیگر ارسال نماید.

اجزای یک شبکه عصبی عبارتند از :

*** ورودی ها** . ورودی‌ها می‌توانند خروجی سایر لایه‌ها بوده و یا آنکه به حالت خام در اولین

لایه و به صورت‌های زیر باشد :

داده‌های عددی و رقمی

متون ادبی، فنی و ...

تصویر و یا شکل

*** وزن ها** . میزان تاثیر ورودی x_i بر خروجی y ، توسط وزن اندازه‌گیری می‌شود.

*** تابع جمع** . در شبکه‌های تک نرونی، تابع جمع در واقع خروجی مسئله را تا حدودی

مشخص می‌کنند و در شبکه‌های چند نرونی نیز تابع جمع میزان سطح فعالیت نرون z در لایه‌های درونی را مشخص می‌سازد.

*** تابع تبدیل** . بدیهی است که تابع جمع پاسخ مورد انتظار شبکه نیست. تابع تبدیل

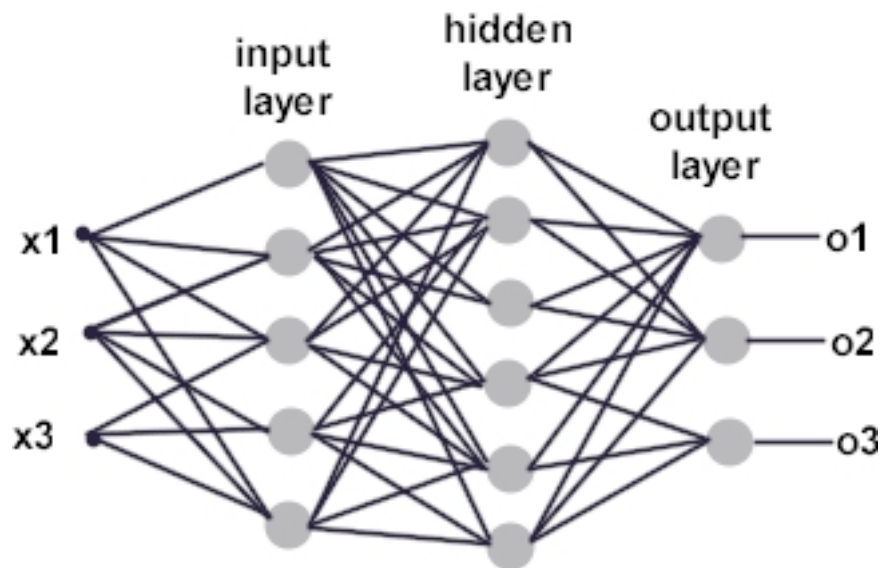
عصوی ضروری در شبکه‌های عصبی محسوب می‌گردد. انواع و اقسام متفاوتی از توابع تبدیل وجود دارد که بنا به ماهیت مسئله کاربرد دارند. این تابع توسط طراح مسأله انتخاب می‌گردد و بر اساس انتخاب الگوریتم یادگیری، پارامترهای مسأله (وزن‌ها) تنظیم می‌گردد.

*** خروجی** . منظور از خروجی، پاسخ مسئله است.

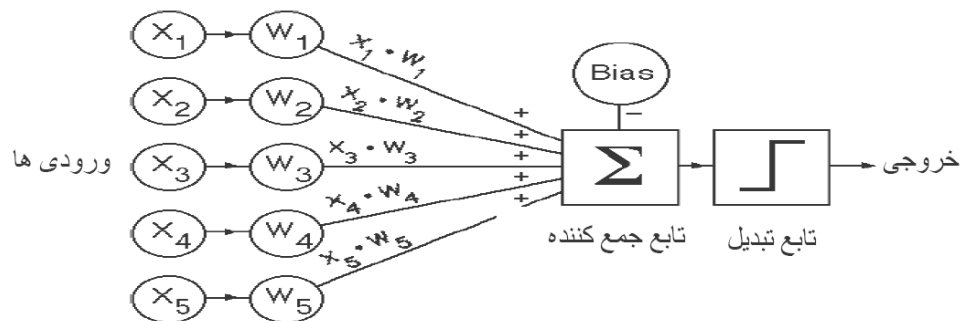
۳-۳ - شبکه عصبی پرسپترون چند لایه:

شکل زیر ساختار یک شبکه عصبی چند لایه را نشان می دهد.

این شبکه دارای یک لایه ورودی با ۵ نورون و یک لایه میانی با شش نورون و یک لایه خروجی با سه نورون می باشد.



یک نورون از این شبکه نیز در زیر نشان داده شده است.



در یک شبکه عصبی هدف از یادگیری، تعیین بهینه ضرایب وزن نورونها به گونه‌ای است که خروجی مورد انتظار (خروجی واقعی) شبکه به خروجی جاری (خروجی حاصل از اعمال ورودی به شبکه و گرفتن خروجی آن) تا حد ممکن به هم نزدیک باشند و یا به عبارت دیگر خطای میان مقدار خروجی واقعی و مقدار خروجی شبکه کمینه شود.

۴- پیاده‌سازی توسط محیط MATLAB

۴-۱- جعبه ابزار شبکه عصبی متلب

در پیاده سازی شبکه عصبی برای طبقه بندی دیتای بازار بورس از جعبه ابزار شبکه عصبی متلب استفاده شده است. منتها به جای استفاده از محیط گرافیکی آماده از بخش کد کردن آن استفاده شده است. در این بخش مروری کوتاه بر این ابزار بسیار مفید خواهیم داشت و خواهیم دید که چگونه می توان از آن برای تشکیل ساختار و یادگیری یک شبکه عصبی استفاده کرد.

۴-۲- تشکیل ساختار شبکه عصبی پرسپترون چند لایه در متلب

برای تشکیل ساختار شبکه عصبی پرسپترون چند لایه در متلب و با استفاده از جعبه ابزار شبکه عصبی از دستور newff استفاده می کنیم. فرمت این دستور به صورت زیر است.

$net = newff(PR, [S1 \ S2 \dots \ SNI], \{TF1 \ TF2 \dots \ TFN\}, BTF, BLF, PF)$

که در آن داریم.

PR	یک ماتریس $R \times 2$ از مقادیر مینیما و ماکزیمم R تا ورودی شبکه است.
Si	تعداد نرون ها در لایه نام است.
TFi	تابع انتقال لایه نام است. تابع انتقال پیش فرض مورد استفاده توسط متلب تابع 'tansig' می باشد.
BTF	تابع یادگیری شبکه پس انتشار خطا است. تابع پیش فرض 'trainlm' می باشد.
BLF	تابع یادگیری وزن ها و بایاس در روش پس انتشار خطا است. تابع پیش فرض 'learngdm' می باشد.
PF	تابع عملکرد یا ارزیابی شبکه است. شبکه به گونه ای آموزش خواهد دید تا