

# بهبود عملکرد طبقه‌بند شبکه عصبی چندجمله‌ای با استفاده از الگوریتم بهینه‌سازی نهنگ

مهسا معماری، عباس حریفی و عبدالله خلیلی

طبقه‌بندی و انجام تحلیل آماری است [۱]. ساختار معماری مناسب شبکه عصبی، تعداد لایه‌های پنهان و تعداد عصب‌ها در هر لایه پنهان، از عواملی هستند که می‌توانند تا حد زیادی بر دقت پیش‌بینی تأثیر بگذارند. متأسفانه هیچ روش مستقیمی وجود ندارد که بتواند برای مشخص کردن این عوامل استفاده شود؛ زیرا آنها باید بر اساس یک سری آزمایش‌های تجربی و پایه تعیین شوند [۱۰]. علاوه بر این، درک و تفسیر طبقه‌بند شبکه عصبی مشکل است، چرا که با افزایش متغیرها و لایه‌های پنهان، شبکه عصبی پیچیده‌تر می‌شود [۱۱] و [۱۲]. برای رفع نواقص ذکر شده، شبکه عصبی چندجمله‌ای<sup>۲</sup> (PNN) بر اساس روش کنترل گروهی داده‌ها پیشنهاد گردیده است [۱۳] و [۱۴].

شبکه عصبی چندجمله‌ای، یک معماری و ساختار عصبی انعطاف‌پذیر می‌باشد که توپولوژی آن به صورت از پیش تعیین شده یا ثابت مانند ANN معمولی نیست؛ اما از طریق یادگیری لایه به لایه رشد می‌کند. در این شبکه‌ها، لایه‌های متوالی با پیوندهای پیچیده با لایه قبل ایجاد می‌شوند. هر عصب در هر لایه، از طریق یک چندجمله‌ای رگرسیون درجه دو با دو ورودی از عصب‌های لایه قبل مرتبط است. اولین لایه توسط محاسبه رگرسیون متغیرهای ورودی و انتخاب بهترین‌ها برای بقا ایجاد می‌شود. لایه دوم با محاسبه رگرسیون مقادیر در لایه قبلی همراه با متغیرهای ورودی و حفظ بهترین نامزدها تشکیل می‌گردد. لایه‌های بیشتری ساخته می‌شوند تا زمانی که گسترش شبکه بر اساس معیارهای خاتمه متوقف گردد [۲].

از شبکه عصبی چندجمله‌ای برای حل مسایل مختلف از جمله جداسازی الگوهای چهره و غیر چهره [۱۵]، تعیین پارامترهای مدل رگرسیون خطی فازی [۱۶]، بررسی تشخیص سیگنال در یک سیستم دوبعدی و شناسایی آن [۱۷] استفاده شده است. در [۱۸]، زاواکا و همکارش با استفاده از گسترش یک ساختار کامل از PNN، یک شبکه عصبی چندجمله‌ای تفاضلی (D-PNN) ارائه دادند. در واقع D-PNN، یک تابع جستجو با استفاده از مشتق نسبی چندجمله‌ای در معادله دیفرانسیل دارای چند متغیر مجهول، تولید و جایگزین می‌کند. همچنین پیچیدگی مدل D-PNN رابطه مستقیم با تعداد متغیرهای ورودی دارد. شریف‌پور و همکاران [۱۹] به منظور پیش‌بینی میزان قابل هدایت حرارت و گرما در نانوسیم‌های اکسید آلومینیوم، دو مدل ارائه دادند که عبارت هستند از: ۱) یک سیستم استنتاج عصبی-فازی سازگار (ANFIS) مبتنی بر الگوریتم خوشه‌بندی میانگین C فازی (FCM) و ۲) شبکه عصبی چندجمله‌ای با الگوریتم ژنتیک (GA-PNN). در این مدل‌ها، داده‌های ورودی شامل دما، حجم غلظت نانوذرات، اندازه نانوذرات  $Al_2O_3$  و

چکیده: شبکه عصبی چندجمله‌ای (PNN) یک الگوریتم یادگیری بانظارت و از محبوب‌ترین مدل‌های مورد استفاده در کاربردهای واقعی است. هرچه شبکه عصبی چندجمله‌ای از نظر تعداد توصیفات جزئی (PDها) و لایه‌ها ساختار پیچیده‌تری داشته باشد، نیاز به زمان و حجم بیشتری برای محاسبه و ذخیره‌سازی دارد. در این تحقیق رویکرد جدیدی در زمینه بهبود کارایی طبقه‌بند شبکه عصبی چندجمله‌ای با استفاده از الگوریتم بهینه‌سازی نهنگ (WOA) به نام PNN-WOA پیشنهاد شده که علاوه بر افزایش دقت PNN، زمان و حجم محاسبات قابل تحملی دارد. در رویکرد پیشنهادی، PDها بر اساس ترکیب دوبه‌دوی ویژگی‌ها از نمونه‌های آموزشی در لایه اول تولید می‌شوند. مجموعه‌ای از PDهای تولیدشده در لایه اول، متغیرهای ورودی و بایاس، عصب‌های لایه دوم را تشکیل می‌دهند. در نهایت خروجی شبکه عصبی چندجمله‌ای، توسط مجموع وزن‌دهی شده خروجی‌های لایه دوم به دست می‌آید. با استفاده از الگوریتم بهینه‌سازی نهنگ (WOA) بهترین بردار ضرایب وزن‌دهی به گونه‌ای که شبکه PNN بیشترین دقت طبقه‌بندی را داشته باشد، به دست می‌آید. برای ارزیابی روش PNN-WOA از یازده مجموعه داده موجود در پایگاه داده UCI استفاده شد. نتایج نشان می‌دهند که PNN-WOA در مقایسه با روش‌های پیشین از قبیل PNN-MOPPSO، PNN-RCGA، RCPNN-PSO و S-TWSVM عملکرد مناسبی دارد. همچنین نتایج آزمون آماری فریدمن نشان می‌دهند که در مجموع، روش پیشنهادی PNN-WOA نسبت به سایر روش‌های مقایسه‌شده، از نظر آماری عملکرد بهتری (با مقدار  $P$  برابر  $0/039$ ) داشته است.

کلیدواژه: شبکه عصبی چندجمله‌ای، الگوریتم بهینه‌سازی نهنگ، توصیف جزئی، طبقه‌بندی.

## ۱- مقدمه

طبقه‌بندی، یکی از مهم‌ترین وظایف چالش‌برانگیز [۱] و [۲]، در داده‌کاوی [۳] و [۴]، شناخت الگو [۵]، وب‌کاوی [۶]، بیوانفورماتیک [۷] و پیش‌بینی مالی [۸] و [۹] می‌باشد. شبکه عصبی به عنوان یک ابزار مهم برای طبقه‌بندی ظهور کرده است. شبکه‌های عصبی مصنوعی<sup>۱</sup> (ANN) قادر به تخمین احتمالات پیشین می‌باشند که مبنایی برای ایجاد قانون

این مقاله در تاریخ ۲۸ تیر ماه ۱۳۹۹ دریافت و در تاریخ ۱۰ بهمن ماه ۱۴۰۰ بازنگری شد.

مهسا معماری، گروه مهندسی برق و کامپیوتر، دانشگاه هرمزگان، بندرعباس، ایران، (email: memari.stu@hormozgan.ac.ir).

عباس حریفی (نویسنده مسئول)، گروه مهندسی برق و کامپیوتر، دانشگاه هرمزگان، بندرعباس، ایران، (email: harifi@hormozgan.ac.ir).

عبدالله خلیلی، گروه مهندسی برق و کامپیوتر، دانشگاه هرمزگان، بندرعباس، ایران، (email: khalili@hormozgan.ac.ir).

کروموزوم‌های GA اعداد حقیقی می‌باشند و این روش موسوم به PNN-RCGA است.

چالش اصلی PNN، آموزش به صورت لایه به لایه می‌باشد که موجب ایجاد یک ساختار بزرگ با تعداد قابل توجهی از گره‌ها می‌گردد. این امر شامل محاسبات زیادی شده و در نتیجه از نظر زمان و حافظه پرهزینه است. در این تحقیق برای جلوگیری از پیچیدگی شبکه عصبی چندجمله‌ای، روش PNN-WOA پیشنهاد می‌شود که در این روش، PNN فقط تا لایه دوم گسترش می‌یابد. در گام اول ماتریس ضرایب به بهترین وجه مناسب برای PDهای اولین لایه با استفاده از روش حداقل مربعات به دست می‌آید. در مرحله دوم، PDهای حاصل از لایه اول با تمام ویژگی‌های الگوی ورودی و بایاس به عنوان گره‌های لایه دوم در نظر گرفته می‌شوند. در اینجا با استفاده از الگوریتم بهینه‌سازی نهنگ، بهترین بردار ضرایب برای عصب‌های لایه دوم به دست آمده و در نهایت بیشترین دقت و عملکرد شبکه عصبی چندجمله‌ای حاصل می‌شود. این کار تا زمانی که بهترین بردار ضرایب به دست آید تکرار می‌شود. هنگامی که الگوریتم بهینه‌سازی نهنگ پایان یافت، بهترین بردار ضرایب برای داده‌های آزمایش ذخیره می‌گردند.

سایر بخش‌های مقاله بدین ترتیب هستند: در بخش دوم به معرفی ساختار و روابط ریاضی شبکه عصبی چندجمله‌ای پرداخته می‌شود. توضیح روش پیشنهادی در بخش سوم آمده و نیز در این بخش به چگونگی بهینه‌سازی ضرایب شبکه PNN با استفاده از الگوریتم WOA پرداخته می‌شود. در بخش چهارم، نتایج به دست آمده از PNN-WOA بر روی مجموعه داده‌های موجود در پایگاه داده UCI با روش‌های مشابه مقایسه شده و نهایتاً در قسمت پنجم، نتیجه‌گیری عنوان می‌گردد.

## ۲- ساختار شبکه عصبی چندجمله‌ای

الگوریتم PNN مبتنی بر روش GMDH می‌باشد و از انواع متفاوت چندجمله‌ای‌ها مانند خطی، درجه دوم، مکعبی و غیره استفاده می‌کند [۳۱]. هدف در الگوریتم PNN، ساختن یک شبکه چندلایه است که عملکرد آن طبق (۱) می‌باشد و در نتیجه مدلی پیچیده حاصل می‌گردد. به همین دلیل در این روش، مسئله به چندین مسئله کوچک‌تر با تعداد مجهولات کمتر به فرم چندجمله‌ای مرتبه دو برای تمامی متغیرهای ورودی، طبق (۲) تبدیل می‌شود. پارامتر  $\gamma$  را به عنوان خروجی موقت مسئله در نظر گرفته که به آن متغیر میانی گفته می‌شود. در واقع  $\gamma$  فقط تابعی از  $x_i$  و  $x_j$  و همچنین  $m$  بیانگر تعداد ویژگی‌ها است

$$y = C_0 + \sum_{k_1} C_{k_1} x_{k_1} + \sum_{k_1, k_2} C_{k_1, k_2} x_{k_1} x_{k_2} + \sum_{k_1, k_2, k_3} C_{k_1, k_2, k_3} x_{k_1} x_{k_2} x_{k_3} + \dots \quad (1)$$

$$y = C_0 + C_1 x_i + C_2 x_j + C_3 x_i x_j + C_4 x_i^2 + C_5 x_j^2 \quad (2)$$

$i = 1, \dots, m-1, j = i+1, \dots, m$

الگوریتم PNN با استفاده از همه ترکیبات ممکن از هر دو متغیر ورودی، صرف نظر از دیگر متغیرها، جمعیت PD را در هر لایه بر اساس رابطه

$$k = \binom{m}{2}$$

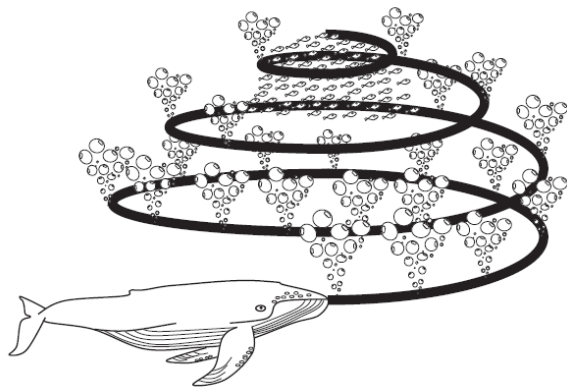
تولید می‌کند.

در شکل ۱ مدل شبکه عصبی چندجمله‌ای پایه برای چهار ورودی نمایش داده شده است.

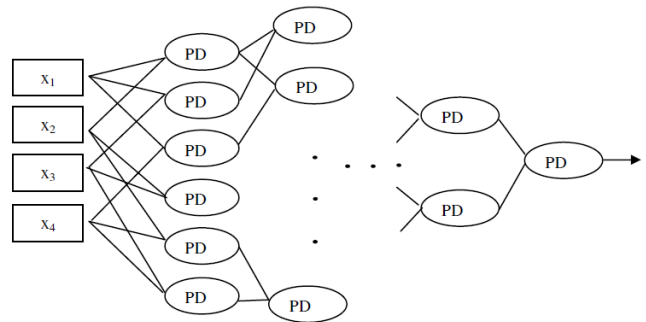
حال اگر بخواهیم (۲) را برای همه ویژگی‌های ورودی به فرم ماتریسی نشان دهیم، خواهیم داشت

ضریب انتشار حرارت نیز به عنوان پارامتر هدف می‌باشند. نتایج به دست آمده نشان داد که این دو روش، نسبتاً خوب هستند اما با وجود این، روش FCM-ANFIS نسبت به مدل GA-PNN موفق‌تر عمل می‌کند. سبحانی و همکاران [۲۰] برای تقریب مقاومت فشار بتن، از توسعه‌دادن شبکه عصبی چندجمله‌ای فازی (FPNN) که در واقع ترکیبی از شبکه عصبی فازی (FNN) و شبکه عصبی چندجمله‌ای است، استفاده کرده‌اند. نتایج نشان می‌دهند که مدل FPNN به منظور پیش‌بینی مقاومت فشار بتن، بسیار مناسب می‌باشد. میسرا و همکاران [۲۱]، یک طبقه‌بندی جدید از شبکه‌های عصبی چندجمله‌ای با استفاده از تکنیک بهینه‌سازی ذرات ارائه داده‌اند. برخلاف PNN، شبکه پیشنهادشده نیاز به تولید توصیف جزئی برای یک لایه دارد. از بهینه‌سازی ازدحام ذرات گسسته (DPSO) برای انتخاب یک مجموعه مناسب PDها و همچنین ویژگی‌هایی با هدف به دست آوردن بهترین دقت استفاده شده است. چن و همکاران [۲۲]، یک مدل بهبود تشخیص الگوی چندکلاسه را با استفاده از ترکیب شبکه عصبی مبتنی بر توابع پایه شعاعی<sup>۱</sup> (RBFNN) و الگوریتم هم‌تکاملی همکار<sup>۲</sup> (Co-CEA) ارائه دادند که با مدل‌های دیگر از جمله شبکه عصبی احتمالاتی<sup>۳</sup> (PNN) مقایسه شده است. میسرا و همکاران [۲۳]، نوع خاصی از شبکه عصبی چندجمله‌ای را که فقط دارای یک لایه است، به نام شبکه عصبی چندجمله‌ای کاهش‌یافته قابل درک پیشنهاد دادند. برای به روز رسانی وزن‌ها در RCPNN، از دو روش کاهش گرادیان و الگوریتم بهینه‌سازی ذرات استفاده شده است. در این تحقیق علاوه بر نتایج مدل‌های RCPNN-GD و RCPNN-PSO، نتیجه مدل شبکه عصبی چندجمله‌ای ارائه شد. در [۲۴]، تیان و همکاران در زمینه یادگیری نیمه‌نظارتی<sup>۴</sup>، دو مدل ماشین بردار پشتیبان لاپلاسیان<sup>۵</sup> (Lap-SVM) و ماشین بردار پشتیبان لاپلاس دوگانه<sup>۶</sup> (Lap-TSVM) را ارائه داده‌اند. در [۲۵]، چو و همکارش با در نظر گرفتن دو هدف که عبارت هستند از به حداقل رساندن پیچیدگی معماری و حداکثر دقت شبکه عصبی چندجمله‌ای، مدل بهینه‌سازی ازدحام ذرات مبتنی بر پارتیولی چندهدفه را پیشنهاد دادند. توسط شی و همکارانش [۲۶]، یک مدل ماشین بردار پشتیبان دوگانه ساختاری<sup>۷</sup> (S-TWSVM) پیشنهاد شد. در این تحقیق روش S-TWSVM با مدل‌هایی از قبیل ماشین بردار پشتیبان ساختاری<sup>۸</sup> (SRSVM)، ماشین بردار پشتیبان دوگانه<sup>۹</sup> (TWSVM) و ماشین ساختاریافته با حاشیه زیاد<sup>۱۰</sup> (SLMM) مقایسه گردیده است. علاوه بر الگوریتم‌های ذکرشده، برخی از الگوریتم‌های جدید نظیر PNPSVM در [۲۷]، NPrSVM در [۲۸]، Twin-NN در [۲۹] و ATP-SVM در [۳۰] نیز توسط محققین ارائه شده که با روش پیشنهادی بر روی مجموعه داده‌های مشترک مقایسه می‌گردند. همچنین در [۱۴]، لین و همکاران نوع خاصی از شبکه عصبی چندجمله‌ای را که فقط دارای دو لایه است، مورد مطالعه قرار دادند. آنها برای بهبود عملکرد طبقه‌بند شبکه PNN از الگوریتم بهینه‌سازی ژنتیک استفاده کردند. مقادیر ژن‌های

1. Radial Basis Function Neural Network
2. Co-Operative Co-Evolutionary Algorithm
3. Probabilistic Neural Network
4. Semi-Supervised Learning
5. Laplacian-Support Vector Machine
6. Laplacian Twin Support Vector Machine
7. Structural Twin Support Vector Machine
8. Structural Support Vector Machine
9. Twin Support Vector Machine
10. Structured Large Margin Machine



شکل ۲: رفتار تغذیه‌ای شبکه حبابی نهنگ‌های کوهان‌دار [۲۸].



شکل ۱: ساختار اصلی شبکه عصبی چندجمله‌ای (PNN) برای چهار ورودی [۲۳].

### ۱) محاصره کردن طعمه

نهنگ‌های کوهان‌دار قابلیت تشخیص موقعیت طعمه و محاصره کردن آن را دارند. از آنجا که موقعیت بهترین شکار در فضای جستجو مشخص نیست، الگوریتم WOA در ابتدا فرض می‌کند که بهترین نامزد فعلی، از میان دیگر طعمه‌ها، خود هدف و یا آن شکار مطلوب و نزدیک به هدف است. بعد از این که بهترین عامل جستجو مشخص شد، دیگر عامل‌ها موقعیت خود را نسبت به بهترین عامل جستجو تغییر و به روز رسانی می‌کنند. این رفتار نهنگ‌های کوهان‌دار با معادلاتی که در زیر آمده، نشان داده شده است

$$\bar{D} = |\bar{C} \cdot \bar{X}^*(t) - \bar{X}(t)| \quad (6)$$

$$\bar{X}(t+1) = \bar{X}^*(t) - \bar{A} \cdot \bar{D} \quad (7)$$

در (۶) و (۷)،  $t$  به عنوان شمارنده تکرار و این که در چندمین تکرار قرار داریم،  $\bar{A}$  و  $\bar{C}$  بردار ضرایب،  $\bar{X}^*$  بردار موقعیت بهترین جواب به دست آمده تا این لحظه و  $\bar{X}$  بردار موقعیت می‌باشد. نکته قابل توجه این است که  $\bar{X}^*$  باید در هر تکرار، در صورتی که اگر یک جواب بهتر نسبت به بهترین جواب فعلی وجود داشته باشد، به روز شود. نحوه محاسبه بردار  $\bar{A}$  و  $\bar{C}$  در (۸) و (۹) آورده شده است

$$\bar{A} = r_1 \bar{a} \cdot \bar{r} - \bar{a} \quad (8)$$

$$\bar{C} = r_2 \bar{r} \quad (9)$$

در (۸)، مقدار بردار  $\bar{a}$  به صورت خطی از ۲ به سمت صفر در طول دوره کامل تکرارها، در هر دو مرحله اکتشاف و بهره‌برداری کاهش می‌یابد و بردار  $\bar{r}$ ، یک بردار تصادفی در بازه  $[0, 1]$  است.

### ۲) روش شکار شبکه حبابی با به روز رسانی موقعیت در ماریپیج

در این روش، ابتدا فاصله میان موقعیت فعلی عامل  $(X, Y)$  و بهترین موقعیت عامل جستجو  $(X^*, Y^*)$  طبق (۱۰) به دست آورده شده و سپس با توجه به (۱۱)، حرکت نهنگ‌ها در جهت مسیر طعمه به صورت ماریپیج شبیه‌سازی می‌گردد

$$\bar{D}' = |\bar{X}^*(t) - \bar{X}(t)| \quad (10)$$

$$\bar{X}(t+1) = \bar{D}' \cdot e^{bl} \cdot \cos(r_3 \pi l) + \bar{X}^*(t) \quad (11)$$

در (۱۱)،  $l$  یک عدد تصادفی در بازه  $[-1, 1]$ ،  $b$  یک مقدار ثابت برای تعیین شکل ماریپیج لگاریتمی و  $\cdot$  به عنوان ضرب داخلی و ضرب جزء به جزء است.

دو روشی که گفته شد در مرحله بهره‌برداری می‌باشند.

$$Y = \begin{bmatrix} 1 & x_{1,i} & x_{1,j} & x_{1,i}x_{1,j} & x_{1,i}^2 & x_{1,j}^2 \\ 1 & x_{r,i} & x_{r,j} & x_{r,i}x_{r,j} & x_{r,i}^2 & x_{r,j}^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{N,i} & x_{N,j} & x_{N,i}x_{N,j} & x_{N,i}^2 & x_{N,j}^2 \end{bmatrix} \begin{bmatrix} C_{k,1} \\ C_{k,2} \\ C_{k,3} \\ C_{k,4} \\ C_{k,5} \end{bmatrix} \quad (3)$$

$$Y = X \cdot A$$

در (۳)، ماتریس‌های  $Y$ ،  $X$  و  $A$  به ترتیب بیانگر مقادیر به دست آمده برای PDها، رگرورها و ماتریس ضرایب وزنی می‌باشند که ماتریس‌های  $Y$  و  $A$  مجهول هستند. لذا فقط با ضرب  $X^T$  در طرفین معادله بالا، می‌توان ماتریس ضرایب وزنی را طبق (۴) محاسبه نمود که  $Y$  دلالت بر مقدار خروجی اصلی دارد. به متغیر  $(X^T X)^{-1} X^T Y$  شبه‌معکوس سمت چپ گفته می‌شود، چون از سمت چپ در  $Y$  ضرب شده است

$$A = (X^T X)^{-1} X^T Y \quad (4)$$

سپس با به دست آوردن مقادیر ضرایب طبق (۴)، هدف زیر را برای داده‌های آموزشی به ازای  $1 \leq i \leq n$  و  $1 \leq j \leq k$ ،  $k = m(m-1)/2$  در نظر می‌گیریم

$$\hat{y}_i = C_{j_1} + C_{j_2} x_{ip} + C_{j_3} x_{iq} + C_{j_4} x_{ip} x_{iq} + C_{j_5} x_{ip}^2 + C_{j_6} x_{iq}^2 \quad (5)$$

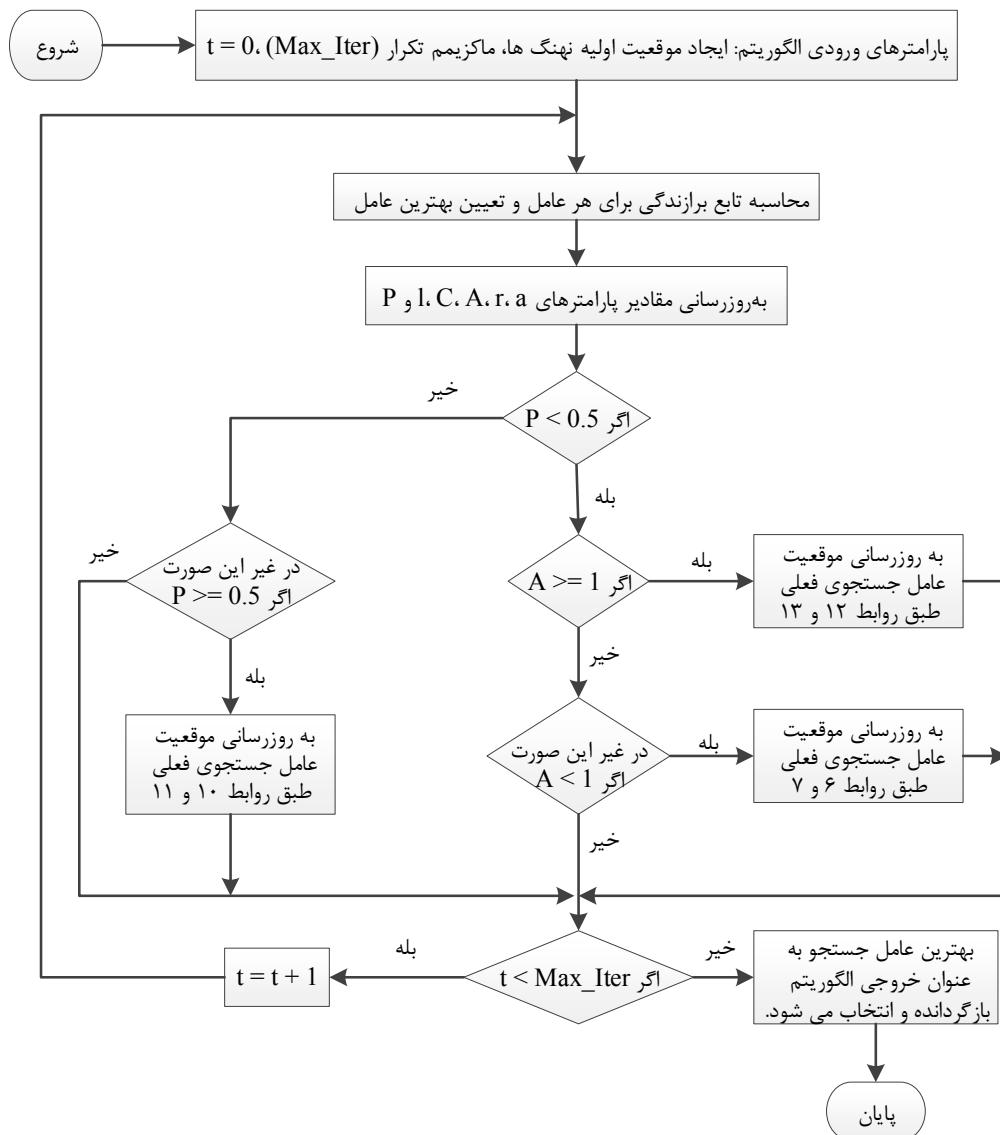
حال اگر میزان خطای به دست آمده، مقدار مورد انتظار نباشد، باید از PDها برای ایجاد لایه بعدی الگوریتم PNN بر اساس خروجی‌های به دست آمده از لایه قبل استفاده کرد. این روند تا زمانی که مقدار خطا به کمترین مقدار خود ( $\epsilon \geq 0$ ) برسد، تکرار می‌شود.

## ۳- روش پیشنهادی

### ۳-۱ الگوریتم بهینه‌سازی نهنگ

الگوریتم بهینه‌سازی نهنگ، یک الگوریتم فراابتکاری جدید مبتنی بر جمعیت می‌باشد که توسط میرجلیلی و همکاران ارائه گردیده است [۳۲]. این الگوریتم از روش خاص شکار نهنگ‌های کوهان‌دار به نام روش تغذیه شبکه حبابی برگرفته شده است [۳۳]. نهنگ‌های کوهان‌دار ترجیح می‌دهند که شکار خود را بر روی سطح دریا انجام دهند و به همین دلیل حباب‌هایی خاص به شکل ماریپیج یا به صورت عدد ۹ انگلیسی ایجاد می‌کنند. در شکل ۲، رفتار تغذیه‌ای نهنگ‌های کوهان‌دار آمده است.

روند جستجوی الگوریتم بهینه‌سازی نهنگ شامل دو مرحله اکتشاف و بهره‌برداری می‌شود که با توجه به پارامتر احتمال و بردار  $A$ ، از یکی از سه روش کاهش فضا و محاصره کردن طعمه، شکار و به روز رسانی بهترین موقعیت نهنگ در حرکات ماریپیج و جستجوی شکار با انتخاب بهترین موقعیت نهنگ به صورت تصادفی استفاده می‌شود.



شکل ۳: روندنمای الگوریتم بهینه‌سازی نهنگ (WOA).

### ۳-۲ بهبود شبکه عصبی چندجمله‌ای با استفاده از الگوریتم بهینه‌سازی نهنگ (روش پیشنهادی)

مدل پیشنهادشده در این تحقیق یک سیستم بهبود عملکرد طبقه‌بند شبکه عصبی چندجمله‌ای با استفاده از الگوریتم بهینه‌سازی نهنگ در حالی که PNN فقط دارای دو لایه می‌باشد، است. همان طور که گفته شد، یکی از قوانین مهمی که در این نوع شبکه عصبی چندجمله‌ای مورد بررسی قرار می‌گیرد، این است که شبکه PNN فقط دو لایه دارد و از گسترش بیش از دو لایه به دلیل پیچیدگی بیشتر شبکه، جلوگیری می‌شود. در ادامه، روند کار این نوع شبکه عصبی چندجمله‌ای به طور کامل شرح داده شده است.

در بخش دوم، چگونگی به دست آمدن ضرایب PDها به همراه روابط ریاضی به طور کامل عنوان شد. حال در لایه دوم، PDهای لایه اول به همراه داده‌های ورودی و بایاس باید در ماتریس ضرایب نظیر به نظیر ضرب شوند و در نهایت مقادیر به دست آمده از تابع فعال‌سازی سیگما شبکه PNN عبور کنند. کاربرد الگوریتم بهینه‌سازی نهنگ در شبکه PNN این است که ماتریس ضرایب به گونه‌ای انتخاب شود که شبکه عصبی نام‌برده کمترین خطا را در طبقه‌بندی داده‌های آموزش داده نشده

### ۳ روش جستجوی شکار با انتخاب بهترین عامل جستجو به صورت تصادفی

در مرحله اکتشاف، روش جستجوی شکار بیان می‌شود که نهنگ‌های کوهان‌دار به صورت تصادفی به دنبال طعمه برای خود می‌گردند. حقیقتاً در مرحله اکتشاف، بهترین عامل جستجو به صورت تصادفی انتخاب می‌شود، در حالی که طی مرحله بهره‌برداری به دنبال یافتن بهترین عامل جستجو هستیم. مبنای روش جستجوی شکار، تغییرات بردار  $\vec{A}$  است و بنابراین اگر مقدار تصادفی بردار  $\vec{A}$  از عدد ۱ بزرگ‌تر و یا کمتر از مقدار ۱- باشد، عامل جستجو مجبور به حرکت و تغییر موقعیت خود نسبت به موقعیت بهترین عامل جستجو است. روابط ریاضی (۱۲) و (۱۳) مربوط به این مدل در ادامه آورده شده‌اند

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_{rand} - \vec{X}(t) \right| \quad (12)$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A} \cdot \vec{D} \quad (13)$$

که  $\vec{X}_{rand}$  بردار موقعیت یک نهنگ می‌باشد که به طور تصادفی از میان جمعیت فعلی انتخاب شده است. شکل ۳ روندنمای الگوریتم بهینه‌سازی نهنگ را نشان می‌دهد.

پس از تغییر و به روز رسانی موقعیت عامل‌های جستجو با استفاده از الگوریتم بهینه‌سازی نهنگ، در صورت بیشتر بودن مقدار تابع برازندگی نهنگ فعلی در مقایسه با بهترین عامل جستجو، آن نهنگ به عنوان بهترین عامل جستجو انتخاب گردیده و موقعیت آن، جایگزین موقعیت بهترین عامل جستجو می‌شود. این فرایند تا رسیدن به ماکسیمم تکرار الگوریتم ادامه می‌یابد. ساختار مدل پیشنهادی برای یک مسئله دو کلاسه با چهار ورودی در شکل ۴ نشان داده شده است.

### ۳-۳ استفاده از آرشیو در روش پیشنهادی

در روش پیشنهادی، با تغییر کوچکی در الگوریتم بهینه‌سازی نهنگ، به جای ذخیره کردن تنها بهترین عامل جستجو (یک عامل)، بهترین عوامل جستجو (به تعداد ثابت و مشخص شده از قبل) در یک آرشیو نگهداری می‌شوند. استفاده از این آرشیو به طور کلی به دو دلیل است:

- وجود چندین بهترین عامل جستجو: در برخی موارد و برای برخی از مجموعه‌های داده، مقدار برازندگی عامل‌های جستجو با یکدیگر برابر می‌گردد. برای مثال در برخی از مجموعه داده‌های مورد استفاده در این تحقیق، مقدار برازندگی چندین عامل جستجو در فرایند آموزش برابر با ۱۰۰٪ شد؛ در حالی که دقت آنها در زمان تست، بسیار متفاوت بود. هرچند استفاده از راهکارهای متداول تقسیم مجموعه داده به مجموعه داده آموزشی و تست می‌تواند از این مشکل بکاهد، اما احتمال رخ دادن این اتفاق بالاست. این مورد نشان می‌دهد که در برخی مجموعه‌های داده، ممکن است چندین بهترین عامل وجود داشته باشد و بنابراین نیاز است که تمامی آنها در نظر گرفته شده تا در زمان تست، عامل با بیشترین دقت انتخاب گردد.

- وجود مجموعه داده‌هایی با تعداد نمونه کم از هر کلاس: در برخی مجموعه‌های داده، ممکن است نمونه‌های تعدادی از کلاس‌ها کم بوده یا اصطلاحاً کلاس‌های با نمونه‌های نادر<sup>۱</sup> وجود داشته باشند. در این شرایط که برای برخی از مجموعه داده‌های این تحقیق نیز وجود دارد، ممکن است عامل‌های با دقت پایین‌تر از بهترین عامل در زمان آموزش، هنگام تست، دقت بالاتری را از بهترین عامل نشان دهند. دلیل این امر، به تقسیم تصادفی نمونه‌ها بین مجموعه داده‌های آموزش و تست برمی‌گردد. در این موارد نیاز می‌باشد که عوامل جستجو بر حسب برازندگی و به صورت نزولی مرتب شده و علاوه بر بهترین عامل جستجو، چند عامل بعدی نیز انتخاب گردند. در این تحقیق، مقادیر مختلف برای سایز آرشیو امتحان گردید و به شکل تجربی، سایز آرشیو برابر ۱۰۰ قرار داده شد. هر زمان که تعداد عامل‌های درون آرشیو از سایز آن بیشتر شد، عامل‌های با کمترین برازندگی از آن حذف گردیده تا سایز آرشیو همیشه ثابت (در اینجا ۱۰۰) بماند.

### ۴- نتایج و بحث

مدل پیشنهادی در محیط برنامه‌نویسی متلب نسخه ۲۰۱۷ پیاده‌سازی شد و بر روی سیستم عامل ویندوز ۱۰ (۶۴بیتی) با مشخصات پردازنده اینتل ۷ هسته‌ای با فرکانس ۴ گیگاهرتز و دارای حافظه موقتی ۱۶ گیگابایت اجرا گردید. مجموعه داده‌های استفاده شده از پایگاه داده UCI جمع‌آوری و اسامی آنها در جدول ۱ آورده شده است. لازم به ذکر است که مقایسه روش PNN-WOA با سایر مدل‌های نام‌برده، بر اساس مقدار

داشته باشد. به طور کلی ساختار هر موقعیت نهنگ شامل مقادیر PDها، ویژگی‌ها و بایاس است. حال اگر  $k$  تعداد کل PDها،  $m$  تعداد کل ویژگی‌های ورودی شبکه،  $p$  تعداد کلاس‌ها و مقدار یک به عنوان بایاس در نظر گرفته شود، طول ماتریس ضرایب از (۱۴) حاصل می‌گردد

$$Dim = p \cdot (k + m + 1) \quad (14)$$

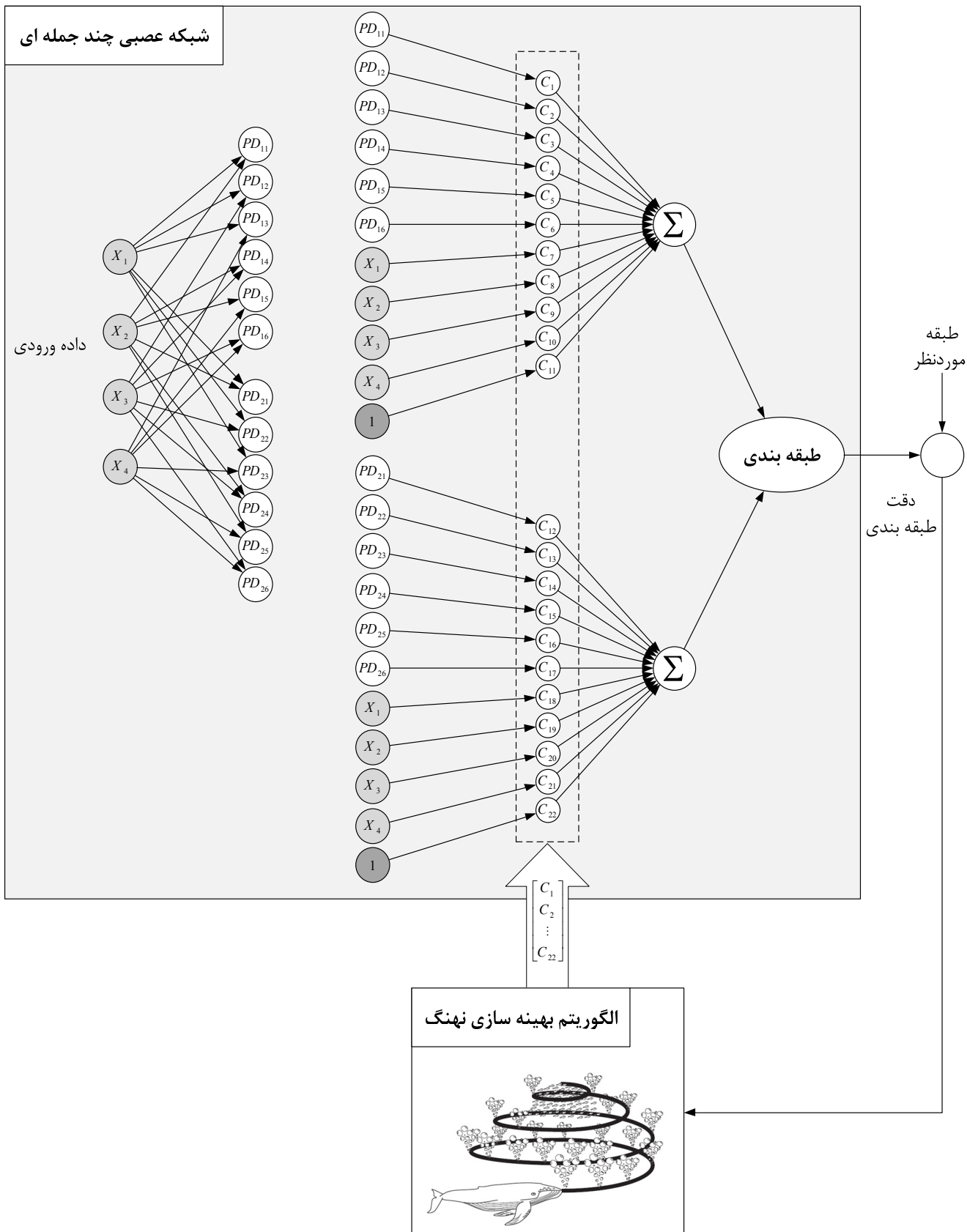
پس اگر طول بردار ضرایب را با  $Dim$  و تعداد عوامل جستجو را که همان تعداد نهنگ‌ها در WOA هستند، با  $SearchAgent\_Number$  بیان کنیم، ابعاد ماتریس جمعیت در الگوریتم WOA برابر است با  $SearchAgent\_Number \times Dim$ . نحوه چیدمان عصب‌ها در لایه دوم شبکه PNN با توجه به تعداد کلاس‌های مجموعه داده با فرض داشتن  $n$  کلاس، بدین گونه است که ابتدا مقادیر PDهای به دست آمده از لایه پیشین مربوط به کلاس اول، متغیرهای ورودی  $X$  و بایاس، مقادیر PDهای کلاس دوم، مجموعه مقادیر  $X$  و بایاس همگی در ماتریس تهی قرار می‌گیرند. این روند جایگذاری کمیت‌ها در ماتریس تا زمان رسیدن به مقادیر PDهای کلاس  $n$ ام و داده‌های ورودی  $X$  و بایاس، ادامه می‌یابد. در گام بعد باید مقادیر ماتریس را نظیر به نظیر در ماتریس ضرایب وزنی به تفکیک کلاس‌ها ضرب کرده و سپس از تابع فعال‌سازی سیگمای شبکه PNN عبور داده که در اینجا مقادیر نیز با هم جمع می‌شوند. حاصل به دست آمده به عنوان مقدار پیش‌بینی شده برای هر کدام از کلاس‌های نمونه اول است و همین مراحل برای مابقی نمونه‌های دیتاست استفاده شده، اعمال می‌گردد. باید به این نکته توجه داشت که طبقه‌بندی در شبکه عصبی اغلب به صورت صفر و یک است و بنابراین خروجی اصلی و پیش‌بینی شده شبکه باید به صورت صفر و یک درآورده شود. به همین دلیل در ابتدای کار، خروجی اصلی نمونه‌ها که همان شماره کلاس‌ها می‌باشند، به صفر و یک تبدیل شدند.

پس از حاصل شدن PDها از ترکیب ویژگی‌ها و ایجاد ساختار عصب‌ها در لایه دوم شبکه PNN، پارامترهای تعداد نهنگ‌ها و ماکسیمم تکرار مقادیر و موقعیت‌ها مشخص می‌شوند. با توجه به این که هدف در سیستم پیشنهادی، پیدا کردن بهترین ماتریس ضرایب با بیشترین مقدار تابع برازندگی است، بنابراین در ابتدای الگوریتم مقدار بهترین عامل جستجو برابر با منفی بی‌نهایت  $(-\infty)$  و برای موقعیت آن، ماتریسی با درایه‌های صفر به طول  $Dim$  در نظر گرفته می‌شود. سپس تابع برازندگی که همان معیار دقت بوده و در بخش ۴ تعریف شده است، برای هر یک از نهنگ‌ها با توجه به موقعیتشان محاسبه و بهترین عامل جستجو انتخاب می‌گردد. در این مرحله از داده‌های اعتبارسنجی برای محاسبه و یافتن بهترین ماتریس ضرایب در تابع برازندگی WOA استفاده شده است. در فاز اصلی الگوریتم WOA، در مرحله به روز رسانی موقعیت نهنگ‌ها، برای تغییر موقعیت هر کدام از وال‌ها با توجه به مقادیر  $P$  و  $A$  مربوط به همان نهنگ، یکی از ۳ حالت زیر انتخاب می‌شود:

(۱) اگر  $P < 0.5$  و  $|A| < 1$  باشد: برای به روز رسانی موقعیت عامل جستجوی فعلی از حالت محاصره کردن طعمه با توجه به (۶) و (۷) استفاده می‌شود.

(۲) اگر  $P < 0.5$  و  $|A| \geq 1$  باشد: بر اساس (۱۲) و (۱۳)، موقعیت نهنگ فعلی با استفاده از روش جستجوی شکار در حالتی که بهترین عامل جستجو به طور تصادفی انتخاب می‌گردد، به روز می‌شود.

(۳) اگر  $P \geq 0.5$  باشد: با استفاده از (۱۰) و (۱۱)، نهنگ فعلی موقعیت خود را با استفاده از حرکات مارپیچ به دور طعمه به روز می‌کند.



شکل ۴: ساختار مدل پیشنهادی (PNN-WOA).

روش PNN-WOA در بیشتر مجموعه داده‌ها بهترین یا دومین بهترین روش بوده و در سایر موارد نیز نتیجه مناسبی داشته است. همگی دیتاست‌ها با شرایط یکسان در سیستم مورد بررسی قرار گرفته که عبارت هستند از: در سیستم پیشنهادی، فرایند نرمال‌سازی مقادیر ویژگی مجموعه داده‌ها بین صفر و یک، به عنوان اولین گام صورت

میانگین نتایج به دست آمده از ۱۰ بار اجرای مدل پیشنهادی بر روی داده‌های مرحله آزمایش می‌باشد. نتایج در جداول ۲ و ۳ و همچنین شکل‌های ۵ و ۶ مورد مقایسه قرار گرفته‌اند. در هر کدام از جداول ۲ و ۳، بهترین روش روی هر مجموعه داده به شکل Bold و دومین بهترین نتیجه به شکل Italic نمایش داده شده است. نتایج نشان می‌دهند که

جدول ۱: مجموعه داده‌های استفاده‌شده در مدل پیشنهادی.

ردیف	مجموعه داده	تعداد نمونه‌ها	تعداد ویژگی‌ها	تعداد کلاس‌ها	تعداد PDها	ابعاد ماتریس ضرایب وزنی
۱	Wine	۱۷۸	۱۳	۳	۷۸	۲۷۶×۱
۲	Balance	۶۲۵	۴	۳	۶	۳۳×۱
۳	Heart_Statlog	۲۶۷	۱۳	۲	۷۸	۱۸۴×۱
۴	Australian	۶۹۰	۱۴	۲	۹۱	۲۱۲×۱
۵	Iris	۱۵۰	۴	۳	۶	۳۳×۱
۶	Bupa Liver	۳۴۵	۶	۲	۱۵	۴۴×۱
۷	Hepatitis	۱۵۵	۱۹	۲	۱۷۱	۳۸۲×۱
۸	Soybean	۴۷	۳۵	۴	۵۹۵	۲۵۲۴×۱
۹	German	۱۰۰۰	۲۰	۲	۱۹۰	۴۲۲×۱
۱۰	WBC <sup>۱</sup>	۶۹۹	۹	۲	۳۶	۹۲×۱
۱۱	Ionosphere	۳۵۱	۳۴	۲	۵۶۱	۱۱۹۲×۱

## I. Wisconsin Breast Cancer

جدول ۲: مقایسه دقت طبقه‌بندی برای روش‌های RCPNN-GD و GMDH-PNN, PNN, CO-RBFNN, PNN-MOPPSO, RCPNN-PSO, PNN-RCGA, PNN-WOA.

دیتاست	روش	PNN-WOA		PNN	PNN-RCGA	GMDH-PNN	RCPNN-GD	RCPNN-PSO	CO-RBFNN	PNN-MOPPSO	ATP-SVM
		حداکثر	میانگین	[۲۲]	[۱۴]	[۲۳]	[۲۳]	[۲۳]	[۲۲]	[۲۵]	[۳۰]
Bupa Liver		۷۳٫۷۹	۷۰٫۱۹	-	۷۴٫۵۴	۶۵٫۲۹	۶۹٫۵۷	۷۰٫۸۷	-	۷۰٫۳۰	۷۵٫۴۸
Heart_Statlog		۸۶٫۲۵	۸۳٫۳۷	۷۳٫۵۸	۷۸٫۱۵	-	-	-	۸۲٫۹۹	-	۸۶٫۳۰
Australian		۸۵٫۵۱	۸۰٫۲۹	-	۸۴٫۷۹	-	-	-	-	-	-
Hepatitis		۹۱٫۴۹	۸۴٫۲۶	-	۸۰٫۷۹	-	-	-	-	-	-
Ionosphere		۹۸٫۰۹	۹۴٫۲۹	۹۴٫۱۰	-	-	-	-	۹۳٫۷۷	-	۸۷٫۴۵
Iris		۱۰۰	۱۰۰	-	۹۹٫۳۳	۸۶٫۲۲	۹۵٫۵۶	۹۸٫۶۷	-	۹۷٫۸۰	۹۷٫۶۷
Soybean		۱۰۰	۱۰۰	۹۰٫۳۷	۹۸٫۰۰	-	-	-	۹۲٫۶۵	-	-
Wbc		۹۷٫۱۳	۹۶٫۹۴	-	۹۷٫۶۶	۹۵٫۹۰	۹۷٫۱۴	۹۷٫۶۴	-	-	-
Wine		۱۰۰	۹۸٫۴۹	۹۵٫۶۸	۹۷٫۳۰	۸۴٫۸۳	۹۵٫۱۳	۹۰٫۹۵	۹۶٫۸۹	۹۸٫۳۱	۹۸٫۳۰
Balance		۹۹٫۴۶	۹۵٫۸۸	-	۸۹٫۲۷	۷۱٫۰۴	۷۷٫۰۸	۸۰٫۰۸	-	-	-

جدول ۳: مقایسه دقت طبقه‌بندی برای روش‌های TWSVM و SLMM, LAP-SVM, LAP-TSVM, SRSVM, S-TWSVM, PNN-WOA.

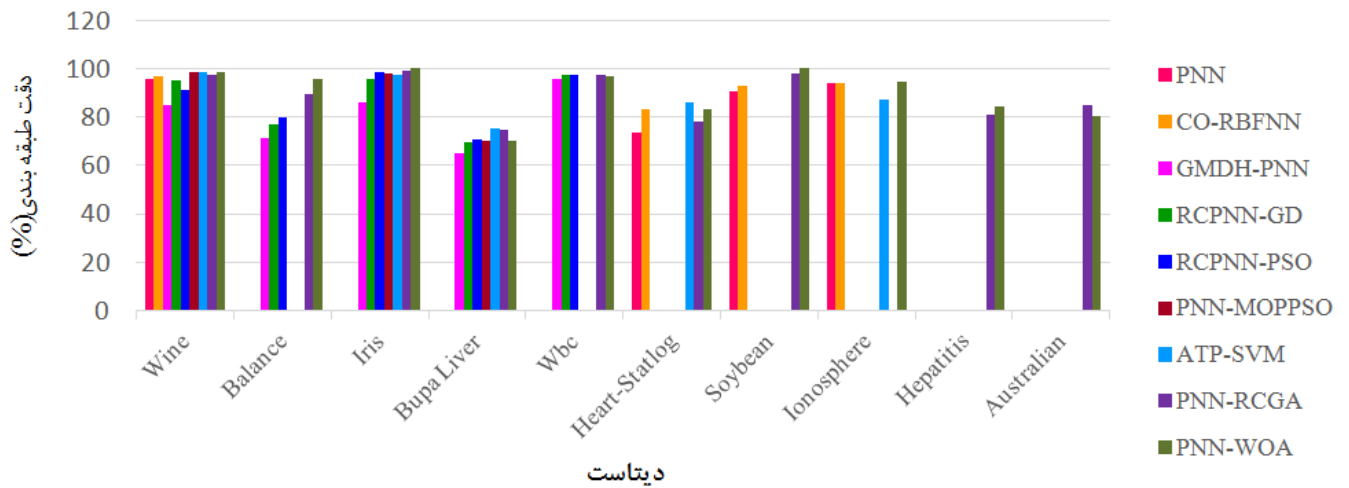
دیتاست	روش	S-TWSVM	SRSVM	SLMM	TWSVM	Lap-TSVM	PNPSVM	NPrSVM	Twin-NN	Lap-SVM	PNN-WOA	
		[۲۶]	[۲۶]	[۲۶]	[۲۶]	[۲۴]	[۲۷]	[۲۸]	[۲۹]	[۲۴]	حداکثر	میانگین
Australian		۷۰٫۵۴±۰٫۸۹	۶۹٫۳۲±۲٫۳۱	۶۸٫۹۲±۳٫۵۴	۶۸٫۴۳±۱٫۱۹	۶۹٫۸۴±۴٫۳۷	۸۷٫۱۰±۲٫۰۶	۸۰٫۵۲±۲٫۹۵	۸۷٫۹۷±۳٫۸۹	۶۶٫۸۹±۴٫۸۷	۸۵٫۵۱	۸۰٫۲۹
Bupa Liver		۶۹٫۷۵±۱٫۸۱	۶۸٫۹۶±۱٫۰۹	۶۹٫۴۲±۱٫۲۸	۶۸٫۴۴±۱٫۱۷	۶۸٫۵۵±۵٫۲۶	۶۹٫۸۶±۶٫۵۱	-	-	۶۷٫۶۱±۵٫۶۳	۷۳٫۷۹	۷۰٫۱۹
Heart_Statlog		۷۶٫۲۲±۱٫۹۲	۷۶٫۰۴±۲٫۴۷	۷۵٫۷۹±۳٫۱۸	۷۵٫۸۶±۱٫۳۸	۷۵٫۷۵±۵٫۷۷	۸۵٫۵۶±۶٫۵۶	-	۸۴٫۸۱±۲٫۷۴	۷۴٫۳۱±۶٫۱۹	۸۶٫۲۵	۸۳٫۳۷
Hepatitis		۸۰٫۶۳±۱٫۳۲	۷۹٫۸۳±۱٫۲۷	۸۰٫۱۲±۲٫۲۵	۷۸٫۴۷±۳٫۱۲	۷۹٫۲۳±۵٫۲۲	۸۷٫۱۵±۶٫۲۵	۸۳٫۹۹±۵٫۱۶	۸۶٫۵۰±۵٫۹۸	۷۷٫۲۶±۶٫۲۴	۹۱٫۴۹	۸۴٫۲۶
German		۶۴٫۱۵±۱٫۷۶	۶۳٫۸۴±۱٫۸۸	۶۳٫۴۴±۲٫۱۲	۶۳٫۹۲±۳٫۴۷	۶۳٫۴۸±۳٫۵۵	۷۴٫۴۰±۰٫۲۹	۷۳٫۲۵±۴٫۶۵	۷۶٫۳۰±۱٫۳۵	۶۴٫۵۷±۳٫۶۸	۷۷٫۳۳	۷۴٫۹۰
Ionosphere		۷۶٫۸۲±۱٫۱۱	۷۶٫۵۵±۲٫۶۳	۷۵٫۴۴±۱٫۸۶	۷۶٫۲۱±۲٫۳۳	۷۵٫۵۹±۳٫۶۱	۹۶٫۳۰±۱٫۷۰	-	۹۴٫۰۱±۱٫۸۷	۷۳٫۸۸±۲٫۲۹	۹۸٫۰۹	۹۴٫۲۹

تقسیم‌بندی شده‌اند، به جز دیتاست Soybean که به دلیل ماهیتش با fold-10 غیر قابل بخش‌بندی بوده و به همین دلیل برای این مجموعه داده مقدار  $k$  برابر با ۳ در نظر گرفته شده است. از دقت جهت ارزیابی کارایی سیستم استفاده می‌شود که (۱۵)، این رابطه را نشان می‌دهد. این معیار علاوه بر ارزیابی کارایی، به عنوان تابع برازندگی (هدف) در الگوریتم بهینه‌سازی نهنگ نیز مورد استفاده است

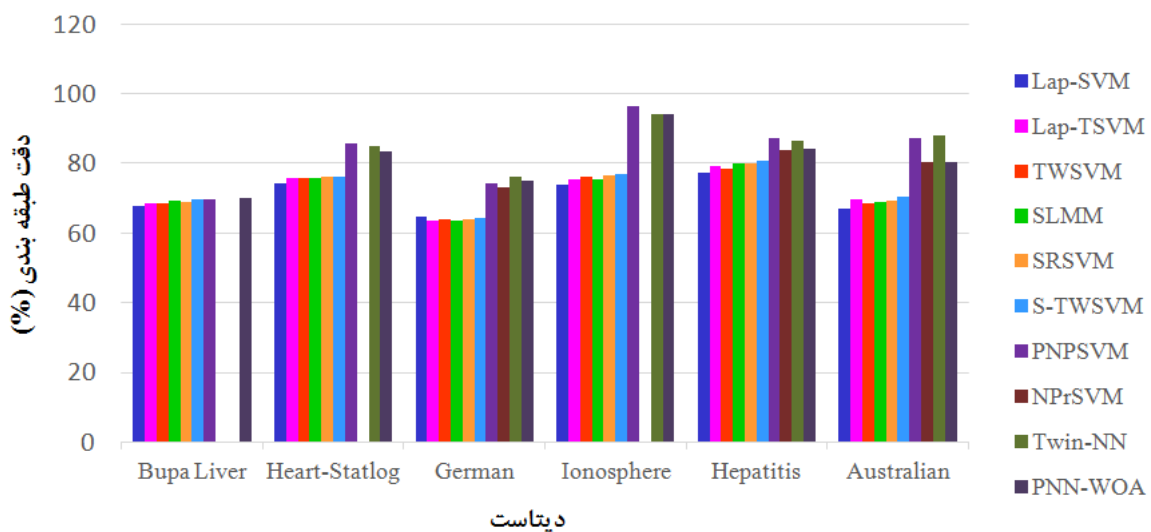
$$Accuracy = \frac{TP + TN}{N} \times 100 \quad (15)$$

در اینجا  $TP + TN$  تعداد نمونه‌هایی می‌باشد که سیستم پیشنهادی به درستی، آنها را در روند آزمایش طبقه‌بندی کرده و  $N$  تعداد کل مجموعه

گرفته است. ۷۰ درصد از داده‌ها برای داده‌های آموزش و ۳۰ درصد برای داده‌های مرحله آزمایش در نظر گرفته شده‌اند. ماکسیمم تعداد تکرار الگوریتم بهینه‌سازی نهنگ برابر با ۱۰۰ است و تعداد ۱۰۰ نهنگ که در واقع همان عامل‌های جستجو هستند، در این الگوریتم مقداردهی شده‌اند. مقادیر کران پایین و کران بالا در الگوریتم WOA به ترتیب برابر با صفر و یک می‌باشند و اندازه پارامتر Archive در الگوریتم WOA برابر با مقدار ثابت ۱۰۰ فرض شده است. به طور کلی، برنامه اصلی ۱۰ بار اجرا گردیده و در هر مرتبه اجرا از  $k$ -fold با فرض مقدار  $k$  برابر با ۱۰ استفاده شده است. سپس برای به دست آوردن مقدار دقت نهایی، میانگین مقادیر دقت به دست آمده از ۱۰ بار اجرای برنامه محاسبه می‌شود. لازم به ذکر است که اکثریت مجموعه داده‌ها در زمان اجرا بر اساس  $k$ -fold-10



شکل ۵: مقایسه دقت طبقه‌بندی برای روش‌های PNN, CO-RBFNN, GMDH-PNN, RCPNN-GD و RCPNN-PSO, PNN-MOPPSO, ATP-SVM, PNN-RCGA, PNN-WOA



شکل ۶: مقایسه دقت طبقه‌بندی برای روش‌های PNN-WOA, S-TWSVM, SRSVM, Lap-TSVM, Lap-SVM, SLMM و TWSVM

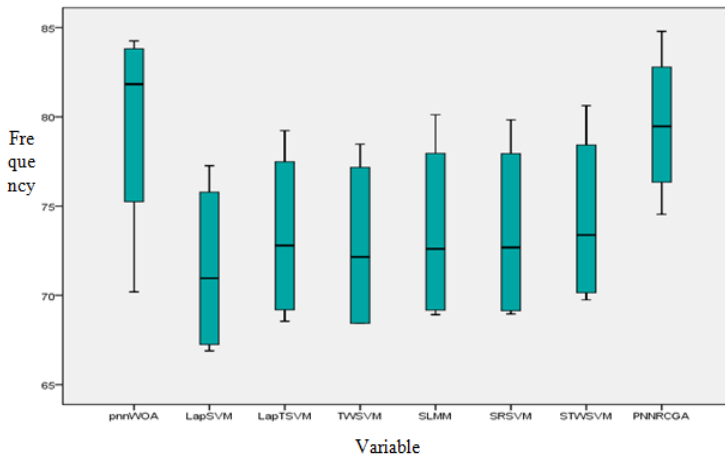
### ۵- نتیجه‌گیری

نتایج گزارش‌شده، نشان‌دهنده کارایی قابل قبول سیستم شبکه عصبی چندجمله‌ای پیشنهادی با استفاده از الگوریتم بهینه‌سازی نهنگ در طبقه‌بندی نمونه‌های مجموعه داده‌ها می‌باشد. گمان می‌رود که روش PNN-WOA بر روی مجموعه داده‌هایی با تعداد ویژگی زیاد و همچنین تعداد نمونه‌های نسبتاً زیاد عملکرد خوبی را از خود نشان دهد. همان‌گونه که می‌دانید، مدل پیشنهادی بر روی یازده مجموعه داده استخراج‌شده اعمال و نتایج در بخش قبل مورد بررسی قرار گرفت. از میان دیتاست‌ها، نتایج به دست آمده در روش PNN-WOA بر روی مجموعه داده‌های Wine, Soybean, Ionosphere, Iris, Hepatitis, Heart-Statlog, German و Balance پژوهش‌های پیشین را مغلوب کرده است. باید به این نکته توجه داشت که مدل پیشنهادی نیز مانند سایر روش‌های موجود ممکن است بنا به دلایلی از جمله ماهیت مجموعه داده، عملکرد نه چندان مطلوبی از خود نشان دهد. در مورد دیتاست‌های Wbc, Bupa Liver و Australian، مدل PNN-WOA دقت طبقه‌بندی کمتری نسبت به روش PNN-RCGA دارد. در همین راستا مطابق با دقت به دست آمده از مدل پیشنهادی و روش‌های پیشین بر روی مجموعه داده‌های عنوان‌شده، تحلیل آماری بر پایه آزمون فریدمن صورت گرفته است. نتایج آزمون فریدمن این موضوع را به اثبات رسانده که سیستم PNN-WOA برترین

داده است. پارامتر  $TP$  بیانگر تعداد داده‌هایی می‌باشد که در اصل مثبت هستند و سیستم آنها را مثبت تشخیص داده است. همچنین  $TN$  دلالت بر تعداد داده‌هایی دارد که واقعاً منفی بوده و سیستم پیشنهادی به درستی آنها را در دسته منفی، طبقه‌بندی کرده است.

به دلیل این که همه مدل‌های ذکرشده بر روی دیتاست‌های مشترک و یکسان پیاده‌سازی و بررسی نشده‌اند، برای مقایسه سیستم پیشنهادی با سایر روش‌های پیشین، اعمال آزمون فریدمن و تحلیل نتایج آن در ۳ نمودار (شکل‌های ۷ تا ۹) و جداول جداگانه صورت گرفته است. آزمون فریدمن یکی از آزمون‌های پارامتری است که فرض واریانس برابر و توزیع نرمال در آن اهمیتی ندارد و تعداد متغیرهای بررسی‌شده باید برابر با ۳ یا بیشتر از آن باشد. این روش برای مقایسه میانگین رتبه‌ها به کار می‌رود. در آزمون فریدمن دو فرض صفر و یک وجود دارد که به ترتیب دلالت بر یکسان بودن اولویت‌ها و یکسان نبودن اولویت‌ها است. بنابراین اگر بعد از آزمون فریدمن مقدار پارامتر Asymptotic Significance کمتر از ۰/۰۵ باشد، فرض صفر رد شده و بر اساس پارامتر میانگین رتبه‌ای، اولویت متغیرها و برتری یا عدم برتری هر کدام از آنها مشخص می‌شود. در صورتی که مقدار پارامتر Asymptotic Significance بیشتر از ۰/۰۵ باشد، بدین معنا است که همگی متغیرها اولویت یکسان و برابری دارند.

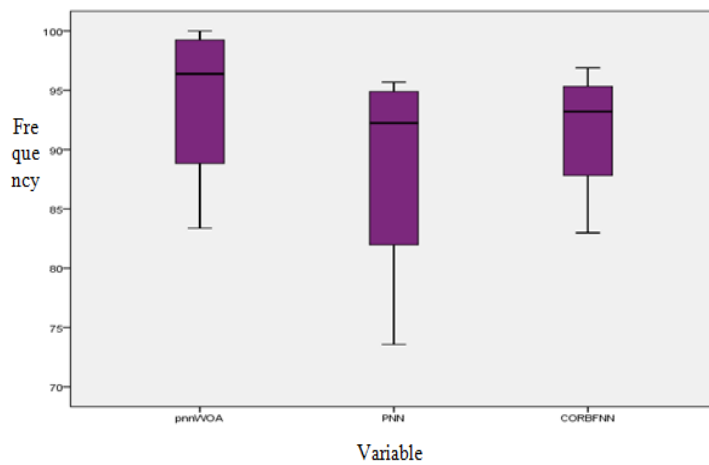




میانگین رتبه ای	
۷/۵۰	PNN-WOA
۱/۰۰	Lap-SVM
۳/۲۵	Lap-TSVM
۲/۵۰	TWSVM
۴/۰۰	SLMM
۴/۲۵	SRSVM
۶/۰۰	S-TWSVM
۷/۵۰	RCGA-PNN

۴	N
۲۵/۵۸۳	Chi-Square
۷	df
۰/۰۰۱	Asymp. Sig.

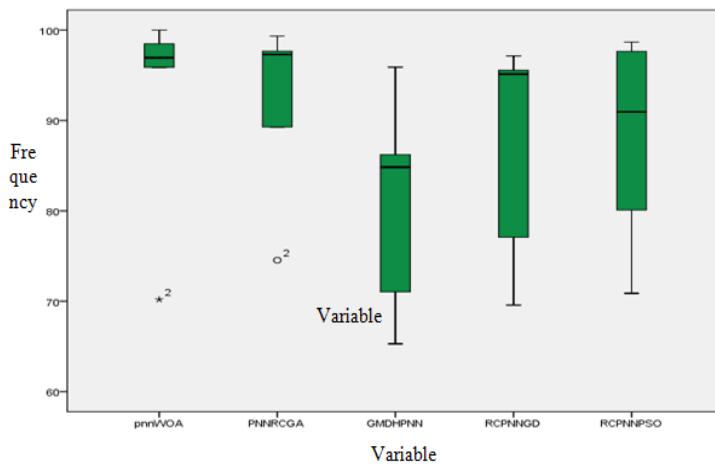
شکل ۷: مقایسه رتبه متغیرهای PNN-WOA, PNN-RCGA, Lap-SVM, Lap-TSVM, TWSVM, SLMM, SRSVM و S-TWSVM بر روی نمونه‌های آزمون فریدمن .Bupa Liver و Australian, Hepatitis, Heart-Statlog.



میانگین رتبه ای	
۳/۰۰	PNN-WOA
۱/۲۵	PNN
۱/۷۵	CO-RBFNN

۴	N
۶/۵۰۰	Chi-Square
۲	df
۰/۰۳۹	Asymp. Sig.

شکل ۸: مقایسه رتبه متغیرهای PNN-WOA, PNN, CO-RBFNN بر روی نمونه‌های آزمون فریدمن .Wine و Soybean, Ionosphere, Heart-Statlog.



میانگین رتبه ای	
۴/۰۰	PNN-WOA
۴/۴۰	PNN-RCGA
۱/۰۰	GMDH-PNN
۳/۲۰	RCPNN-PSO
۲/۴۰	RCPNN-GD

۵	N
۱۴/۷۲۰	Chi-Square
۴	df
۰/۰۰۵	Asymp. Sig.

شکل ۹: مقایسه رتبه متغیرهای PNN-WOA, PNN-RCGA, RCPNN-PSO, GMDH-PNN و RCPNN-GD بر روی نمونه‌های آزمون فریدمن .Wbc, Bupa Liver, Iris, Balance و Wine.

روش در مقایسه با سایر مدل‌های پیشین است.

## مراجع

weights," *Expert Syst. Appl.*, vol. 168, no. 7, Article ID: 114312, Apr. 2021.

[3] J. Zhou, H. G. Amir, F. Chen, and A. Holzinger, "Evaluating the quality of machine learning explanations: a survey on methods and metrics," *Electronics*, vol. 10, no. 5, Article ID: 593, 19 pp., Mar. 2021.

[4] M. Kantardzic, *Data Mining: Concepts, Models, Methods, and Algorithms*, John Wiley, 2020.

[5] J. Beyerer, M. Richter, and M. Nagel, *Pattern Recognition: Introduction, Features, Classifiers and Principles*. Walter de Gruyter GmbH & Co KG, 2017.

[1] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, "State-of-the-art in artificial neural network applications: a survey," *Heliyon*, vol. 4, no. 11, Article ID: e00938, Nov. 2018.

[2] N. L. da Costa, M. D. de Lima, and R. Barbosa, "Evaluation of feature selection methods based on artificial neural network

- Pattern Recognition Letters*, vol. 29, no. 12, pp. 1705-1712, Sept. 2008.
- [24] Z. Qi, Y. Tian, and Y. Shi, "Laplacian twin support vector machine for semi-supervised classification," *Neural Networks*, vol. 35, no. 4, pp. 46-53, Nov. 2012.
- [25] S. Dehuri and S. B. Cho, "Multi-criterion Pareto based particle swarm optimized polynomial neural network for classification: a review and state-of-the-art," *Computer Science Review*, vol. 3, no. 1, pp. 19-40, Feb. 2009.
- [26] Z. Qi, Y. Tian, and Y. Shi, "Structural twin support vector machine for classification," *Knowledge-Based Systems*, vol. 43, no. 7, pp. 74-81, May 2013.
- [27] Q. Hou, L. Liu, L. Zhen, and L. Jing, "A novel projection nonparallel support vector machine for pattern classification," *Eng. Appl. Artif. Intell.*, vol. 75, no. 7, pp. 64-75, Oct. 2018.
- [28] W. J. Chen, Y. H. Shao, C. N. Li, Y. Q. Wang, M. Z. Liu, and Z. Wang, "NPrSVM: nonparallel sparse projection support vector machine with efficient algorithm," *Appl. Soft Comput.*, vol. 90, no. 3, Article ID: 106142, May 2020.
- [29] H. Pant, M. Sharma, and S. Soman, "Twin neural networks for the classification of large unbalanced datasets," *Neurocomputing*, vol. 343, no. 4, pp. 34-49, May 2019.
- [30] R. Rastogi, P. Saigal, and S. Chandra, "Angle-based twin parametric-margin support vector machine for pattern classification," *Knowl. Based. Syst.*, vol. 139, no. 6, pp. 64-77, Jan. 2018.
- [31] S. K. Oh, W. Pedrycz, and B. J. Park, "Polynomial neural networks architecture: analysis and design," *Computers & Electrical Engineering*, vol. 29, no. 6, pp. 703-725, May 2003.
- [32] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in Engineering Software*, vol. 95, no. 6 pp. 51-67, May 2016.
- [33] W. A. Watkins and W. E. Schevill, "Aerial observation of feeding behavior in four baleen whales: *Eubalaena glacialis*, *Balaenoptera borealis*, *Megaptera novaeangliae*, and *Balaenoptera physalus*," *J. of Mammalogy*, vol. 60, no. 1, pp. 155-163, Feb. 1979.
- [6] T. A. Al-Asadi, O. J. Ahmed, R. Hidayat, and A. A. Ramli, "A survey on web mining techniques and applications," *Int. J. on Adv. Sci. Eng. and Inf. Tech.*, vol. 7, no. 4, pp. 1178-1184, 2017.
- [7] Y. Cao, T. A. Geddes, J. Y. H. Yang, and P. Yang, "Ensemble deep learning in bioinformatics," *Nat. Mach. Intell.*, vol. 2, no. 9, pp. 500-508, Aug. 2020.
- [8] I. Sadgali, N. Sael, and F. Benabbou, "Performance of machine learning techniques in the detection of financial frauds," *Procedia Comput Sci*, vol. 148, no. 6, pp. 45-54, 2019.
- [9] J. Henriquez and W. Kristjanpoller, "A combined independent component analysis-neural network model for forecasting exchange rate variation," *Appl Soft Comput*, vol. 83, no. C, Article ID: 105654, Oct. 2019.
- [10] I. N. Da Silva, D. H. Spatti, R. A. Flauzino, L. H. B. Liboni, and S. F. dos Reis Alves, "Artificial neural network architectures and training processes," *Artif. Neural Netw., Springer, Cham*, pp. 21-28, 2017.
- [11] C. Singh, W. J. Murdoch, and B. Yu, *Hierarchical Interpretations for Neural Network Predictions*, arXiv preprint arXiv: 1806.05337, 2018.
- [12] F. Ros, M. Pintore, and J. R. Chrétien, "Automatic design of growing radial basis function neural networks based on neighborhood concepts," *Chemometrics and Intelligent Laboratory Systems*, vol. 87, no. 2, pp. 231-240, Jun. 2007.
- [13] A. G. Ivakhnenko, "Polynomial theory of complex systems," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 1, no. 4, pp. 364-378, Oct. 1971.
- [14] C. T. Lin, M. Prasad, and A. Saxena, "An improved polynomial neural network classifier using real-coded genetic algorithm," *IEEE Trans. on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 11, pp. 1389-1401, Nov. 2015.
- [15] L. L. Huang, A. Shimizu, Y. Hagihara, and H. Kobatake, "Face detection from cluttered images using a polynomial neural network," *Neurocomputing*, vol. 51, no. 12, pp. 197-211, Apr. 2003.
- [16] S. B. Roh, T. C. Ahn, and W. Pedrycz, "Fuzzy linear regression based on Polynomial Neural Networks," *Expert Systems with Applications*, vol. 39, no. 10, pp. 8909-8928, Aug. 2012.
- [17] C. C. Huang, W. C. Chen, C. Y. Shen, Y. J. Chen, C. Y. Chang, and R. C. Hwang, "Signal processing by polynomial NN and equivalent polynomial function," in *Proc. 1st Int. Conf. on Pervasive Computing Signal Processing and Applications, PCSPA'10*, pp. 460-463, Harbin, China, 17-19 Sept. 2010.
- [18] L. Zjavka and W. Pedrycz, "Constructing general partial differential equations using polynomial and neural networks," *Neural Networks*, vol. 73, no. 1, pp. 58-69, Jan. 2016.
- [19] M. Mehrabi, M. Sharifpur, and J. P. Meyer, "Application of the FCM-based neuro-fuzzy inference system and genetic algorithm-polynomial neural network approaches to modelling the thermal conductivity of alumina water nanofluids," *International Communications in Heat and Mass Transfer*, vol. 39, no. 7, pp. 971-977, Aug. 2012.
- [20] M. F. Zarandi, I. B. Türksen, J. Sobhani, and A. A. Ramezani-pour, "Fuzzy polynomial neural networks for approximation of the compressive strength of concrete," *Applied Soft Computing*, vol. 8, no. 1, pp. 488-498, Jan. 2008.
- [21] S. Dehuri, B. B. Misra, A. Ghosh, and S. B. Cho, "A condensed polynomial neural network for classification using swarm intelligence," *Applied Soft Computing*, vol. 11, no. 3, pp. 3106-3113, Apr. 2011.
- [22] M. Li, J. Tian, and F. Chen, "Improving multiclass pattern recognition with a co-evolutionary RBFNN," *Pattern Recognition Letters*, vol. 29, no. 4, pp. 392-406, Mar. 2008.
- [23] B. B. Misra, S. Dehuri, P. K. Dash, and G. Panda, "A reduced and comprehensible polynomial neural network for classification,"

**مهسا معماری** در سال ۱۳۹۳ مدرک کارشناسی مهندسی کامپیوتر- نرم افزار خود را از دانشگاه پیام نور کازرون و در سال ۱۳۹۷ مدرک کارشناسی ارشد مهندسی کامپیوتر- هوش مصنوعی و رباتیک خود را از دانشگاه هرمزگان دریافت نمود. زمینه های تحقیقاتی مورد علاقه نامبرده شامل موضوعاتی مانند ایده های نو در شبکه های عصبی، الگوریتم های تکاملی و پردازش تصویر می باشد.

**عباس حریفی** در سال ۱۳۸۱ مدرک کارشناسی مهندسی برق- کنترل خود را از دانشگاه شیراز و در سال های ۱۳۸۴ و ۱۳۸۹ به ترتیب مدرک کارشناسی ارشد و دکتری مهندسی برق- کنترل خود را از دانشگاه تبریز دریافت نمود. دکتر عباس حریفی از سال ۱۳۸۹ استادیار گروه مهندسی برق و کامپیوتر دانشگاه هرمزگان می باشند. زمینه های تحقیقاتی مورد علاقه ایشان عبارتند از: رباتیک، یادگیری عمیق، یادگیری تقویتی، سیستم های کنترل هوشمند و سیستم های کنترل بهینه.

**عبدالله خلیلی** مدارک کارشناسی و کارشناسی ارشد خود را در رشته مهندسی کامپیوتر- نرم افزار در سال ۱۳۸۸ و ۱۳۹۰ به ترتیب از دانشگاه اصفهان و دانشگاه علم و صنعت ایران دریافت نمود. همچنین نامبرده موفق به اخذ مدرک در مقطع دکترا در سال ۱۳۹۵ رشته مهندسی کامپیوتر- سیستم های نرم افزاری از دانشگاه شیراز شد. ایشان از سال ۱۳۹۵ استادیار گروه مهندسی برق و کامپیوتر دانشگاه هرمزگان می باشند و زمینه های علمی مورد علاقه شان شامل این موارد است: یادگیری ماشین، امنیت سایبری، امنیت شبکه های برق و امنیت سیستم های کنترل صنعتی.