

طراحی سیستمهای چندگانه رده بندی با استفاده از الگوریتم ژنتیک

ناصر مزینی دانشکده کامپیوتر دانشگاه علم و صنعت ایران mozayani@iust.ac.ir	بهروز مینایی دانشکده کامپیوتر دانشگاه علم و صنعت ایران b_minaei@iust.ac.ir	اکرم بیگی دانشکده کامپیوتر دانشگاه علم و صنعت ایران beigi@iust.ac.ir	حمید پروین دانشکده کامپیوتر دانشگاه علم و صنعت ایران h_parvin@comp.iust.ac.ir
---	---	---	--

تعمیم یک مجمع⁴ به طور مشخص بهتر از یک رده بند یکه است [2,3]. استفاده از مجمع برای مسائل سخت یک راه حل عملی و موثر است [4]. این روشها تحت عناوین روشهای ترکیبی⁵، ترکیب تصمیم⁶، مجمع رده بندی⁷، آمیزش رده بندها⁸ و مواردی از این قبیل مطرح شده اند.

بدلیل اینکه رده بندها با ویژگیهای مختلف و روشهای گوناگونی کار می کنند و می توانند یکدیگر را کامل کنند؛ بنابراین روشهای ترکیبی معمولاً در نتایج رده بندها بهبود ایجاد می کنند [4-6]. کانچوا [7] در کار خود با استفاده از قضیه هیات کاندورست⁹، نشان داده است که ترکیب رده بندها معمولاً می تواند بهتر از یک رده بند ای که مولفه های آن مستقلند عمل کند. این بدان معنی است که اگر تنوع رده بندهایی که در مجمع استفاده شده اند بیشتر باشد، آنگاه خطای آنها به طور قابل ملاحظه ای کاهش می یابد. در حالت کلی کارهای نظری و عملی نشان داده اند که جمعی خوب عمل می کند که اعضای آن که شامل رده بندها هستند هم تنوع و هم دقت داشته باشند. به عبارت دیگر خطای هر کدام از رده بندهای عضو یک مجمع در بخشی از فضای مساله ایجاد می شود [9,10]. رویکردهای زیادی برای ایجاد اینگونه مجمع ها پیشنهاد شده است. یک گروه از این روشها مانند روشهای تقویتی¹⁰ و کیسه ای¹¹، اعضای مختلف خود را برای اینکه تنوع آنها حفظ شود روی مجموعه های آموزشی مختلف آموزش می دهند [10-13]. روشهای دیگر ساختار متفاوت، وزن دهی اولیه گوناگون، تنظیم پارامترهای متنوع و الگوریتمهای یادگیری گوناگون را به منظور ایجاد رده بندهای متنوع به کار می گیرند.

چکیده: یکی از شیوه های ساخت یک سیستم تشخیص استفاده از یک مجمع رده بند به جای استفاده از رده بندهای مستقل و یکه است. استفاده از مجمع در رده بندی در مسائل مشکل بسیار راه گشا است. در یک مجمع با کنار هم قرار گرفتن رده بندهای متنوع بسیاری از خطاهای رده بندهای عضو مجمع کاهش می یابد. در روش مجمع چند رده بند به طور جداگانه آموزش می بینند و سپس نتایج به یک سیستم تصمیم گیری ارائه می شود.

در این مقاله برای طراحی سیستمهای چندگانه رده بندی از الگوریتم ژنتیک استفاده شده است. بوسیله الگوریتم ژنتیک می توان از گیر افتادن در بهینه های محلی جلوگیری کرده و به یک مجمع تقریباً بهینه نزدیک شد. همچنین برای افزایش سرعت محاسبه تابع برازندگی، بجای استفاده از میزان دقت رده بند از تقریب آن استفاده شده است. این تقریب به کمک یک سری تابع برازش اندازه گرفته می شود. الگوریتم معرفی شده روی داده های واقعی و غیر واقعی آزموده شده است.

واژه های کلیدی: سیستمهای رده بندی - مجمع رده بند - الگوریتم ژنتیک.

۱ - مقدمه

استفاده از سیستمهای تشخیص¹ تقریباً در تمام زمینه ها کاربردهای زیادی پیدا کرده است. هرچند هر نوع از الگوریتمهای رده بندی² برای مسائل خاصی کارایی خوبی داشته اند، اما پایداری آنها برای دیگر مسائل ضعیف شمرده می شود. تحقیقات انجام شده اخیر نشان داده است که ترکیب چندگانه رده بندها³ بسیار قویتر و پایدارتر و با مقاومت و دقت و تعمیم پذیری بالاتری نسبت به استفاده یک الگوریتم رده بند به تنهایی، عمل می کنند. ترکیب چندگانه رده بندها به عنوان یک روش عام برای مسائل تشخیص الگو شناخته شده است.

ورودیهای الگوریتم CMC نتایج جداگانه رده بندها و خروجی آن تصمیم ترکیبی این الگوریتم است [1,2]. این روشها چند رده بند ساده را آموزش می دهد و سپس پیش بینی آنها را ترکیب می کند. توانایی

⁴ Ensemble

⁵ Hybrid Methods

⁶ Decision Combination

⁷ Classifier Ensembles

⁸ Classifier Fusion

⁹ Condorcet Jury Theorem

¹⁰ Boosting

¹¹ Bagging

¹ Recognition Systems

² Classification Algorithms

³ Combination of Multiple Classifiers (CMC)

۲- کارهای مرتبط

فرض موجود در انتخاب رده بندها این است که یک اوراکل^{۱۲} تعیین کننده بهترین رده بند برای هر الگو وجود دارد. تصمیم آن رده بند به عنوان تصمیم کل مجمع در نظر گرفته می شود. ایده انتخاب رده بندها اولین بار در [14] ارایه شده است. در [15] نیز یک روش کاملا مدون برای استفاده از انتخاب رده بندها ارایه شده است. فرض کنیم که می توانیم در یک مجمع از رده بند ها، اعضا را بر اساس دقت از پیش تعیین شده آنها، مرتب کنیم. حال برای یک الگوی ورودی، فرض کنیم میزان اعتماد در نظر گرفته شده برای هر رده بند، حداکثر احتمالی باشد که آن رده بند به یکی از رده ها می دهد. سپس از اولین رده بند (دقیقترین رده بند) شروع کرده و روال زیر را دنبال می کنیم. اگر میزان اعتماد آن رده بند برای الگوی ورودی بالا باشد، تصمیم آن رده بند را به عنوان تصمیم کل مجمع در نظر می گیریم. در غیر این صورت، سراغ رده بند بعدی رفته و میزان اعتماد آن را بررسی می کنیم. این کار را تا جایی ادامه می دهیم که یک رده بند قابل اعتماد (با میزان اعتماد بالا) یافت نشود. اگر همه رده بند ها برای الگوی ورودی دارای میزان اعتماد کمی باشند، آنگاه یا مجمع هیچ تصمیمی در رابطه با آن الگو نمی گیرد و یا بالاجبار یک رده که مثلا دارای بیشترین اعتبار است را انتخاب می کند [14,16,17]. روندهای دیگری نیز از انتخاب رده بندها در [18] آمده است. دو روند کلی برای انتخاب رده بندها وجود دارد که در ادامه بیان می شود:

۲-۱ تخمین بهترین رده بند محلی به صورت پویا

یعنی در حین ساخت مجمع، بهترین رده بند برای هر ناحیه انتخاب می شود. در [19]، انتخاب رده بندها بر اساس موقعیت الگوی ورودی در فضای ویژگیها است. در این روش برای تخمین k -تا همسایه آن الگو را به عنوان ورودی به همه رده بندها می دهیم و میزان اعتماد یک الگوی ورودی دقت هر رده بند برای ناحیه آن الگو بدست می آوریم. این دقت را به عنوان تخمینی از میزان اعتماد آن رده بندها برای این الگو در نظر می گیریم [18,20]. در [19] این کار به صورت وزن دار انجام شده است. در ادامه کارهای قبل، در [18] برای رده بندی یک الگوی ورودی به صورت زیر عمل می شود. ابتدا فرض کنید که رده بند n ، الگوی ورودی را به رده m انتساب دهد. مجموعه S ، را به عنوان مجموعه ای از نمونه های موجود در داده های آموزشی که توسط رده بند n به رده m منتسب شده اند، در نظر بگیرید. در اینجا، دقت آن رده بند در این مجموعه S میزان اعتماد آن رده بند برای الگوی ورودی در نظر گرفته می شود. همچنین نسخه وزن دار آن در [18] پیشنهاد شده است.

۲-۲ تخمین نواحی لیاقت

در روش قبل محاسبه k نزدیک ترین همسایه کمی زما نبر است. به منظور کاهش پیچیدگی زمانی می توان فضا را به نواحی لیاقت^{۱۳} تقسیم بندی کرد. وقتی الگویی برای رده بندی به یک مجمع می رود، ابتدا ناحیه ای که الگوی ورودی در آن قرار دارد تعیین می شود. سپس با رده بند (یا رده بندهای) مخصوص آن ناحیه کار رده بندی را انجام می دهیم. یعنی در حقیقت ما کار رده بندی را به زیرسیستم هایی تقسیم کرده ایم. با این کار ممکن است بعضی از نواحی شامل تعداد اندکی از داده ها باشد. برای جلوگیری از این واقعه و متعادل کردن نواحی، ما می توانیم داده ها را خوشه بندی کنیم و سپس برای هر خوشه یک ناحیه در نظر بگیریم. الگوریتم این فرایند اینگونه است:

۱. رده بندهای D_1, D_2, \dots, D_L را به صورت مجزا بوجود می آوریم.
۲. تعداد k را برای ناحیه ها تعیین می کنیم. داده ها را به k خوشه تقسیم می کنیم و مرکز خوشه ها را به نام C_1, C_2, \dots, C_K ذخیره می کنیم.
۳. برای C_i یک ناحیه در نظر می گیریم و یک یا چند رده بند از مجموعه رده بند ها (D_i ها) را برای تصمیم گیری در مورد این ناحیه در نظر می گیریم.

در [21,22] بعد از ایجاد خوشه ها، روی هر خوشه یک رده بند آموزش داده شده است. در فرآیند آموزش، داده ها را به دو مجموعه Z^- و Z^+ تقسیم می کنیم. به طوریکه شامل الگوهایی است که توسط رده بند D_i درست رده بندی شده اند و Z^- شامل بقیه الگوها است. فرض کنید که C تا خوشه در Z^+ داریم که هر خوشه معادل یک رده باشد و k_i تا خوشه نیز از Z^- بدست آورده ایم؛ به طوریکه k_i را در حین فرآیند آموزش به دست آورده ایم. به علاوه میزان اعتماد هر رده بند D_i در هر کدام هر کدام از خوشه های موجود، با استفاده از یک روش خاص محاسبه می شود. برای رده بندی یک الگوی ورودی، در اولین گام، خوشه متناظر با الگوی ورودی را بر اساس فاصله ماهالانوبیس^{۱۴} مشخص می کنیم. سپس میزان کفایت هر رده بند برای آن خوشه را به آن الگو منتسب می کنیم. در نهایت آن رده بندی که بیشترین میزان کفایت را دارد، برای اتخاذ تصمیم نهایی کل مجمع انتخاب می شود [23]. تنها مشکل روش قبل تعیین مناسب میزان اعتماد هر رده بند در خوشه ها است که این مشکل بسیار بزرگی است.

در [24-27] چند معیار^{۱۵} برای پیچیدگیهای^{۱۶} هر مجموعه داده ارائه شده است. فرض کنید که یک مجموعه داده از مجموعه داده های گوناگون به نام S داریم. به طوریکه هر عنصر از مجموعه داده S خود

¹³ Competence Regions

¹⁴ Mahalanobis

¹⁵ Measurement

¹⁶ Complexity

¹² Oracle

یک مجموعه داده است. با توجه به معیارهای پیچیدگی موجود، می توان هر مجموعه داده را به یک بردار از معیارهای پیچیدگی نگاشت کرد. در نتیجه مجموعه داده S به یک مجموعه داده S^* نگاشته می شود؛ به گونه ای که هر نمونه از مجموعه داده S^* یک بردار خواهد بود. آنگاه یک رده بند بخصوص C (مثلا شبکه عصبی) را بر روی هر مجموعه داده آزمایش می کنیم و خطای آن رده بند را به عنوان برچسب برای مجموعه داده S^* در نظر می گیریم. حال حال یک مجموعه داده بانظر داریم و با یک مسئله رده بندی مواجه هستیم. در این صورت می توانیم یک رده بند برای تشخیص میزان مناسب بودن رده بند C برای هر کدام از مجموعه داده ها، به دست آوریم. آنگاه رده بندهای دیگری را به شکل مشابه برای میزان مناسب بودن بقیه انواع رده بندهای پایه بدست آوریم.

از طرفی در [27-29] دیده شده است که هر کدام از ویژگی های پیوسته^{۱۷} را می توان به یک ویژگی گسسته تبدیل کرد. همچنین در آنجا یک نوع رده بندی ارائه شده است که از یک سری قانون برای رده بندی استفاده می کند. هر قانون شامل یک بخش شرط^{۱۸} (A) و یک بخش نتیجه (B) به شکل $A \rightarrow B$ است. برای هر قانون بخش شرط (که اغلب این را بررسی می کند که آیا در یک زیرمجموعه از فضای ویژگیها هستیم یا نه) بررسی می شود. اگر آن شرط درست باشد بخش نتیجه را (که می توان برای رده بندی از آن استفاده شود) اجرا می کند.

۳- شرح مساله

برای یک مجموعه داده بزرگ چگونه میتوان یک چارچوب به منظور ردهبندی تقریبا بهینه ارائه نمود. از آنجاییکه نشان داده شده است که ردهبندهای پایه ذاتا به تنهایی نمیتوانند رده بند بهینه باشند، استفاده از آمیزش رده بندها ضروری به نظر می رسد و از آنجا که بهینگی در رده بندی مورد نظر است، انتخاب رده بندها مورد نیاز است. همچنین می خواهیم سربار زمانی این سیستم را تا حد ممکن به زمان آموزش منتقل کنیم و زمان آزمایش آن بسیار سریع باشد.

۴- چالشهای مسئله

- ما در انتخاب رده بندها از این ویژگی رده بندها استفاده می کنیم که هر رده بند توزیع دقت نامتوازن در فضای ویژگی ها دارد. یعنی سعی می کنیم آن رده بندهایی که در ناحیه مطلوب (همسایگی الگوی ورودی) بهترین کارایی را دارند، بیابیم.
- در آمیزش رده بندها سعی می کنیم تا این ایراد (که هر رده بند توزیع دقت نامتوازن در فضای ویژگی ها دارد) را با زیاد کردن آرا بپوشانیم. این می تواند باعث مستحکم^{۱۹} شدن نتایج شود.

البته خود این روش هم با مشکل بزرگی به نام نوع سلیقه در رای دهنده ها مواجه است.

- بعضی الگوریتم ها مثل الگوریتم ژنتیک، قادر هستند تا مسائل سخت را با دقت قابل قبولی حل کنند. بخصوص نشان داده شده است که برای حل مسئله یافتن مجمع رده بندهای متنوع، این سری الگوریتم ها قابل استفاده است [30,31].
- اگر ما بتوانیم به گونه ای از این سه با هم استفاده کنیم، نتایج حاصله، هم با استحکام است (ویژگی آمیزش رده بندها) و هم بنا به موقعیت تصمیم می گیرد و در نتیجه نتایج دقیق تر هستند (ویژگی انتخاب رده بندها). همچنین، موضوع تنوع می تواند با الگوریتم ژنتیک حل شود.
- چگونه میتوان با ویژگیهای پیوسته برخورد کرد؛ چراکه لازم است فضای داده خود را به زیرسیستمهایی تقسیم کنیم که [27-29] به این مسئله پرداخته است.
- چنانچه پیشتر گفته شد چالش دیگری به روشهای موجود با روشهای گفته شده در بند الف مرتبه زمانی بالا برای بدست آوردن k نزدیکترین همسایه است که این باعث طولانی شدن زمان آزمایش می شود. در حالی که ما می خواهیم سربار زمانی را تا حد ممکن به زمان آموزش منتقل کنیم و زمان آزمایش بسیار سریع باشد.
- در مورد روشهای گفته شده در بند ب نیز چنانچه گفته شد، به سختی میتوان میزان اعتماد مناسبی برای نواحی تعریف کرد.

۵- روش پیشنهادی

در اینجا ما برای یک مجموعه داده واحد، مجموعه داده را به زیربخش هایی تقسیم کرده ایم و هر کدام از این زیربخشها را یک مجموعه داده فرض کرده ایم. البته شاید زیرمجموعه ها همپوشانی هم داشته باشند. برای هر کدام از این زیر بخش ها بهترین رده بند را ایجاد می کنیم. احتمالا به موازات میتوان قواعدی را نیز بیرون بکشیم. آنگونه که در چه محدوده هایی از فضای ویژگی (یعنی در چه زیربخشهایی)، کدام رده بندها بهترین اند؛ یا حتی مجمع کدام رده بندها بهترین است که این خود یعنی کدام رده بندها بیشترین تنوع را دارند. این کار به خاطر پیچیدگی بالا باید به صورت تقریبی حل شود نه مطلق که این عمل را الگوریتم ژنتیک با تقریب احتمالا خوبی می تواند حل کند. هدف اصلی از این کار این است که دقت عمل رده بندی را بالاتر ببریم.

در حقیقت ما از الگوریتم ژنتیک استفاده کرده ایم تا از بهینه های محلی فرار کرده و به یک مجمع تقریبا بهینه نزدیک شویم. هر کروموزوم در حقیقت یک سری زیرفضا بر روی مجموعه داده آموزشی تعریف می کند که داده های آموزشی موجود در این زیرفضاها به کمک یک رده بند تقریب دقت، با تقریبا بهترین رده بند ممکن آموزش داده میشود. طول کروموزوم به فضای ویژگی مساله و همچنین تعداد حداکثر

¹⁷ Continuous Features

¹⁸ Condition

¹⁹ Robust

```

test accuracy = evaluate Ensemble(GA_Ensemble,
Ensemble, test data);
return GA_Ensemble, Ensemble, validation accuracy, test
accuracy;
function evaluate Ensemble(GA_Ensemble, Ensemble,
data)
Result = Test Ensembl(GA_Ensemble, data);
for c= 1 to number of data
if(isEmpty(Result(c)))
Result(c) = Test Ensembl(GA_Ensemble, data(c));
end if
end for
return compute accuracy(Result);

```

شکل ۱- الگوریتم مجمع رده بندی مبتنی بر ژنتیک

چنانچه از شکل فوق مشاهده می شود، ابتدا مجموعه داده را به سه بخش آموزشی، آزمایشی و اعتبارسنجی افزایش می‌کنیم. داده های اعتبارسنجی ۲۰٪ و داده های آموزشی ۶۵٪ و داده های آزمایشی ۱۵٪ از کل داده ها را در بر می گیرد. سپس ما رده بندهای SVM، MLP، KNN، MKNN1، MKNN2 و PNN را بر روی مجموعه داده آموزشی آموزش می دهیم و در مجمع قرار می دهیم. سپس با به کارگیری تکنیک LDA و همان رده بندهای مرحله قبل شش رده بند دیگر به مجموعه رده بندهای موجود در مجمع می افزاییم. بعد از آن ما هزار مجموعه داده غیرواقعی تولید کرده و از هر کدام از این مجموعه داده ها ۹ ویژگی استخراج می کنیم و این مجموعه داده بدون برچسب را مجموعه داده مجموعه داده ها می نامیم.

بعد از این کار بر روی هر مجموعه داده هر کدام از ۱۲ رده بند موجود در مجمع را آموزش داده و دقت آن رده بندها را بدست می آوریم. آنگاه دقت هر رده بند بر روی همه مجموعه داده های غیر واقعی را به عنوان برچسب بر روی مجموعه داده مجموعه داده ها قرار می دهیم. سپس با دوازده SVM (یا MLP) این مجموعه مجموعه داده های برچسب زده شده را رگرسیون می کنیم. (چون ما ۱۲ برچسب داریم). پس در حقیقت ما یک SVM داریم که دقت رده بند SVM را بر روی مجموعه داده تقریب می زند. همچنین یک SVM دیگر داریم که دقت رده بند MLP را بر روی مجموعه داده تقریب می زند؛ ... و سرانجام یک SVM داریم که دقت دوازدهمین رده بند ما (LDA+PNN) را تقریب می زند. حال تا وقتی که دقت بر روی مجموعه داده اعتبارسنجی افزایش پیدا می کند ما هر بار یک واحد به تعداد نواحی موجود در الگوریتم ژنتیک می افزاییم. اما الگوریتم ژنتیک در حقیقت به دنبال زیرفضاهایی است که اولاً بیشترین داده های مجموعه داده آموزشی را در بر بگیرد و ثانیاً دقیقترین رده بندها را بر روی آن نواحی داشته باشیم. یک کروموزوم از الگوریتم ژنتیک را میتوان در شکل زیر مشاهده کرد.

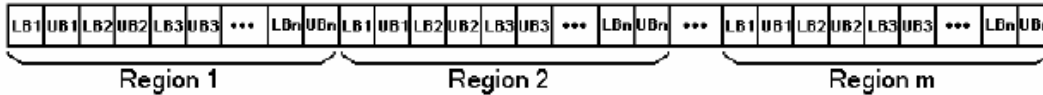
زیرفضاهایی که می خواهیم الگوریتم ژنتیک در آن به دنبال جواب بگردد، بستگی دارد.

تابع برازندگی الگوریتم ژنتیک به گونه ای تعریف شده است که مجمعی را انتخاب کند که هر کدام از رده بندها در زیرفضای خود بیشترین دقت را داشته باشند و همچنین کل مجمع داده های بیشتری را در بر بگیرد. در نتیجه در نهایت خروجی الگوریتم ژنتیک یک سری زیرفضاهایی است که بر روی آنها رده بندهایی وجود دارد که بیشترین دقت را دارند و این مجمع تا حد ممکن همه جای مجموعه داده ما را در بر می گیرد. شبه کد این الگوریتم را در شکل زیر مشاهده می کنید:

```

AlgorithmGa(original data set, min_number of regions,
max_number of regions);
training data, validation data, test data = extract
(original data set);
Ensemble = Create_ensemble_of_our_12_classifier
(training data, validation data);
max_number of classes = number of classes;
max_number of features = number of features;
Datasets = Generate_dataset(1000, max_number of class,
max_number of features);
Features = Extract_features(Datasets);
Classifiers = train_classifier1(Datasets);
Accuracies = Evaluate(classifiers, Datasets);
Dataset_of_datasets(1) = Combine(Features,
Accuracies);
Classifiers = train_classifier2(Datasets);
Accuracies = Evaluate(classifiers, Datasets);
Dataset_of_datasets(2) = Combine(Features,
Accuracies);
.
Classifiers = train_classifier12(Datasets);
Accuracies = Evaluate(classifiers, Datasets);
Dataset_of_datasets(12) = Combine(Features,
Accuracies);
for c= 1 to 12
Classifier(c) = train_classifier1(Dataset_of_datasets(c));
\\classifier1 is SVM
end for
previous_acc = 0;
for c= min_number of classes to max_number of regions
subspaces = Run_Ga(c);
GA_Ensemble = Analysis (subspaces);
current_acc = evaluate Ensemble(GA_Ensemble,
Ensemble, validation data);
If(current_acc<previous_acc)
Break;
end if
previous_acc = current_acc;
end for
GA_Ensemble = Analysis (subspaces);
validation accuracy = evaluate Ensemble(GA_Ensemble,
Ensemble, validation data);

```

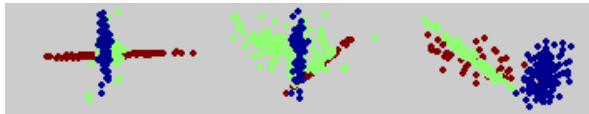


شکل ۲- یک کروموزوم نمونه شامل m ناحیه و n ویژگی

- میانگین تعداد همسایه های همرده بر روی مجموعه داده های آموزشی

۶- نتایج

این الگوریتم را بر روی مجموعه داده های واقعی Iris و Wine آزمایش کرده ایم و نتایج آن را در جداول آمده قابل مشاهده است: شکل زیر مجموعه داده های دست ساز 1، 2 و 3 را به ترتیب از چپ به راست نشان می دهد.



شکل ۳- مجموعه داده های دست ساز شماره 1، 2 و 3

نتایج این الگوریتم بر روی سایر مجموعه داده ها نیز در جدول ۳ آورده شده است.

در تمامی نتایج بدست آمده در جداول فوق، حداکثر و حداقل تعداد نواحی را ۱۰ در نظر گرفته ایم که از اینجا می توان برداشت کرد که از مجموعه داده اعتبارسنجی هیچ گونه استفاده ای در حین اجرای الگوریتم ژنتیک نشده است و برآورد دقت بر روی این مجموعه داده معتبر است. در این نتایج ما از عملگر جهش گوسین استفاده کرده ایم و همچنین عملگر تقطیع، تقطیع یکنواخت در نظر گرفته شده است. با مشاهده این جداول، افزایش قابل ملاحظه دقت چه بر روی مجموعه داده آزمایشی و چه بر روی مجموعه داده اعتبارسنجی واضح است. دقت داشته باشید که این افزایش در مقایسه با مجمع رده بندها است که خود از دقت خوب و ثابتی برخوردار است. این خود نشان دهنده این است که این روش ساخت مجمع به کمک الگوریتم ژنتیک بسیار مناسب است. البته زمان آموزش این روش در مقایسه با روش مجمع ساده بسیار فرآیند کندتری است. لیکن این روش در مقایسه با روش مجمع ساده دقت بالاتری دارد که این باز نشان میدهد اگر ما دقت بالاتری بخواهیم باید مرتبه زمانی بیشتری صرف کنیم. دقت داشته باشید که این روش در حین آزمایش نیز نسبت به روش مجمع ساده از نظر زمانی کندتر است ولی به طور کلی اگر دقت در فرآیندی برای ما از اهمیت بسیار زیادی برخوردار باشد این روش می تواند گزینه خوبی برای بدست آوردن نتایجی تقریباً بهینه باشد. چرا که این روش از الگوریتم ژنتیک استفاده می کند و در نتیجه می تواند به نتایج بسیار نزدیک به بهینه دست یابد.

چنانچه در شکل مشاهده می شود هر کروموزوم در حقیقت تمام نواحی را تعیین کرده و داده هایی را که در هر ناحیه قرار میگیرند به راحتی میتوان تعیین کرد. سپس برای هر ناحیه داده هایی که از مجموعه داده آموزشی انتخاب می شوند را به کمک SVM های رگرسیون کننده تقریب دقت زد.

پس در حقیقت ما در محاسبه میزان برازش یک کروموزوم نواحی را انتخاب میکنیم که دقت در آن نواحی زیاد باشد و همچنین تمام نواحی داده های زیادی را شامل شوند. به طور خاص فرض کنید که $Acc(1..m)$ حداکثر دقت تقریب زده شده توسط رگرسیون کننده های یک تا دوازده برای نواحی یک تا m باشد و همچنین NUM تعداد داده هایی از مجموعه آموزشی که لااقل در نواحی یک تا m قرار می گیرند. اگر که تعداد داده هایی از مجموعه آموزشی که در یک ناحیه قرار می گیرند کمتر از یک آستانه (در اینجا ۳۰) باشد آن ناحیه را در نظر نمی گیریم. در مورد نواحی دیگر مجموع دقتها را بر مجموع خطاها تقسیم می کنیم و در تعداد کلیه داده هایی که توسط این نواحی انتخاب شده اند ضرب می کنیم و به عنوان برازش این کروموزوم در نظر میگیریم:

$$Fitness = \frac{\sum_{i=1}^m (Acc_i)}{\sum_{i=1}^m (1 - Acc_i) + 1} * Num \quad (1)$$

اما در اینجا در مورد ویژگیهای ۹ گانه مرور کوتاهی خواهیم کرد.

- نرخ تغییر رده در درخت پوشای مینیمم بر روی مجموعه داده
- میزان خطی بودن مجموعه داده با انتخاب تصادفی دو نقطه در مجموعه داده و آزمایش داده های موجود بر یک خط واصل بین آنها با یک رده بند خطی بسیار ساده
- تعداد رده های موجود در مجموعه داده های آموزشی
- نرخ حداکثر و حداقل داده های یک رده به کل داده ها بر روی مجموعه داده های آموزشی
- بیشترین معیار فیشر بر روی ویژگیها بر روی مجموعه داده های آموزشی
- نسبت تعداد داده ها به تعداد ویژگیها بر روی مجموعه داده های آموزشی
- میانگین فواصل اولین همسایه همرده تا اولین همسایه غیر همرده در مجموعه داده آموزشی

جدول ۱- نتایج مجمع رده بندها به کمک الگوریتم ژنتیک در مقابل مجمع ساده روی مجموعه داده واقعی Iris (مقادیر موجود در پرانتز بهترین نتایج موجود در کروموزم نسل آخر الگوریتم ژنتیک است).

Iris	Accuracy of Ga based Ensemble on Validation	Accuracy of Ga based Ensemble on Test	Accuracy of Ga based Ensemble	Accuracy of Ensemble on Validation	Accuracy of Ensemble on Test	Accuracy of Ensemble
Run 1	0.9091	0.9565	0.9328	0.8696	0.9091	0.8894
Run 2	0.9565	1	0.9783	0.9091	1	0.9546
Run 3	0.9565(1)	0.9565	0.9565	0.9091	0.913	0.9111

جدول ۲- نتایج مجمع رده بندها به کمک الگوریتم ژنتیک در مقابل مجمع ساده روی مجموعه داده واقعی Wine

Wine	Accuracy of Ga based Ensemble on Validation	Accuracy of Ga based Ensemble on Test	Accuracy of Ga based Ensemble	Accuracy of Ensemble on Validation	Accuracy of Ensemble on Test	Accuracy of Ensemble
Run 1	1	1	1	0.963	0.9615	0.9623
Run 2	1	0.9615(1)	0.9808	0.963	0.9615	0.9623
Run 3	1	0.9565(1)	0.9783	1	0.9231	0.9616

جدول ۳- نتایج مجمع رده بندها به کمک الگوریتم ژنتیک در مقابل مجمع ساده روی مجموعه داده‌های واقعی Scale.Balance و غیرواقعی شماره ۱ تا ۳ در یک بار اجرا

	Accuracy of Ga based Ensemble on Validation	Accuracy of Ga based Ensemble on Test	Accuracy of Ga based Ensemble	Accuracy of Ensemble on Validation	Accuracy of Ensemble on Test	Accuracy of Ensemble
Balance Scale	0.9362(ND)	0.8817(ND)	0.9090	0.8936	0.8387	0.8661
Dataset 1	0.7778(0.8000)	0.8000(0.8387)	0.7889	0.7556	0.7556	0.7556
Dataset 2	0.8444(ND)	0.7778(ND)	0.8111	0.7778	0.7111	0.7445
Dataset 3	0.8889(ND)	0.8222(ND)	0.8556	0.8444	0.8000	0.8222

به عنوان ادامه این کار می توان از معیارهای بیشتری برای نگاشت مجموعه داده ها به فضای ویژگیها استفاده کرد و همچنین می توان تعداد مجموعه داده های بیشتری را به عنوان مجموعه داده های آموزشی رگرسیون کننده ها در نظر گرفت. همچنین این موضوع که تعداد نواحی را برای هر مجموعه داده چه عددی در نظر بگیریم می تواند مورد بررسی قرار گیرد. همچنین تابع برازندگی در این روش بسیار نقش کلیدی ایفا می کند و اینکه تابع برازندگی دیگری را برای این الگوریتم ژنتیک در نظر بگیریم می تواند موضوع بسیار حیاتی ای باشد. چرا که در بررسی هایی که انجام شد همیشه برازنده ترین (Fittest) کروموزوم بالاترین دقت را بر روی مجموعه داده مورد آزمایش ارائه نمی داد که این نشان می دهد امکان اینکه این تابع برازش تابع بسیار خوبی نباشد وجود دارد.

۷- جمع بندی

در این مقاله برای طراحی سیستمهای چندگانه رده بندی از الگوریتم ژنتیک استفاده کرده ایم تا از بهینه های محلی فرار کرده و به یک مجمع تقریباً بهینه نزدیک شویم. در اینجا ما برای یک مجموعه داده واحد، مجموعه داده را به زیربخش هایی تقسیم کرده ایم و هر کدام از این زیربخشها را یک مجموعه داده فرض کرده ایم. این اولین جایی است که ما یک مجمع بهینه را از اساس به کمک الگوریتم ژنتیک نمایش دادیم. همچنین زمان زیاد محاسبه تابع برازندگی در صورتی که تابع برازندگی میزان محاسبه دقت باشد باعث شد تا از تخمین دقت استفاده نماییم. نتایج بدست آمده بهبود قابل توجهی را روی داده های واقعی و غیر واقعی نشان می دهد.

- [17] L.I. Kuncheva. Change-glasses approach in pattern recognition. *Pattern Recognition Letters*, 14:619–623, 1993.
- [18] K. Woods, W. P. Kegelmeyer, and K. Bowyer. Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:405–410, 1997.
- [19] G. Giacinto and F. Roli. Design of effective neural network ensembles for image classification processes. *Image Vision and Computing Journal*, 19(9–10):699–707, 2001.
- [20] L. A. Rastrigin and R. H. Erenstein. Method of Collective Recognition. *Energoizdat*, Moscow, 1981.
- [21] L. I. Kuncheva. Clustering-and-selection model for classifier combination. In *Proc. Knowledge-Based Intelligent Engineering Systems and Allied Technologies*, Brighton, UK, 2000, pp. 185–188.
- [22] L. I. Kuncheva. Switching between selection and fusion in combining classifiers: An experiment. *IEEE Transactions on Systems, Man, and Cybernetics*, 32(2):146–156, 2002.
- [23] R. Liu and B. Yuan. Multiple classifier combination by clustering and selection. *Information Fusion*, 2:163–168, 2001.
- [24] T.K. Ho, M. Basu, Complexity measures of supervised classification problems, *IEEE Trans. On PAMI*, 24, 3, March 2002, 289-300.
- [25] Tin Kam Ho: Complexity of Classification Problems and Comparative Advantages of Combined Classifiers. *Multiple Classifier Systems 2000*: 97-106.
- [26] Domain of Competence of XCS Classifier System in Complexity Measurement Space, Ester Bernado Mansilla, Tin Kam Ho, *IEEE Transaction on Evolutionary Computation*. 9, 1, February 2005, 82-104.
- [27] S. W. Wilson, .Mining Oblique Data with XCS. in *Advances in Learning Classifier Systems: Proceedings of the Third International Workshop*, ser. *Lecture Notes in Artificial Intelligence*, P. Lanzi, W. Stolzmann, and S. Wilson, Eds., vol. 1996. Springer-Verlag Berlin Heidelberg, 2001, pp. 158.176.
- [28] E. Bernad’o Mansilla, X. Llor a F abrega, and J. M. Garrell Guiu, .XCS and GALE: a Comparative Study of Two Learning Classifier Systems on Data Mining. in *Advances in Learning classifier Systems*, 4th International Workshop, ser. *Lecture Notes in Computer Science*, P. Lanzi, W. Stolzmann, and S. Wilson, Eds., vol. 2321. Springer, 2002, pp. 115.132.
- [29] Kotagiri Ramamohanarao, Hongjian Fan: Patterns Based Classifiers. *World Wide Web* 2007:71-83.
- [30] On Classifier Domains of Competence, Ester Bernado Mansilla, Tin Kam Ho, *Proceedings of the 17th International Conference on Pattern Recognition*, Cambridge, U.K., August 22-26,2004, Volume 1, 136-139.
- [31] H. Parvin, H. Alizadeh and B. Minaei-Bidgoli, “A New Approach to Improve the Vote-Based Classifier Selection”, *International Conference on Networked Computing and advanced Information Management (NCM 2008)*, Korea, Sep. 2008, Published by IEEE CS & Indexed by ISI.
- [1] B. Minaei-Bidgoli, G. Kortemeyer and W.F. Punch, Optimizing Classification Ensembles via a Genetic Algorithm for a Web-based Educational System, (SSPR /SPR 2004), *Lecture Notes in Computer Science (LNCS)*, Volume 3138, Springer-Verlag, ISBN: 3-540-22570-6, pp. 397-406, 2004.
- [2] A. Saberi., M. Vahidi , B. Minaei-Bidgoli, Learn to Detect Phishing Scams Using Learning and Ensemble Methods, *IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Workshops (IAT 07)*, pp. 311-314, Silicon Valley, USA, November 2-5, 2007.
- [3] T.G. Dietterich, Ensemble learning, in *The Handbook of Brain Theory and Neural Networks*, 2nd edition, M.A. Arbib, Ed. Cambridge, MA: MIT Press, 2002.
- [4] H. Parvin, H. Alizadeh, M. Fathi and B. Minaei-Bidgoli, Improved Face Detection Using Spatial Histogram Features, *The 2008 International Conference on Image Processing, Computer Vision, and Pattern Recognition (ICP’08)*, Las Vegas, Nevada, USA, July 14-17, 2008 (in press).
- [5] S. Gunter and H. Bunke, Creation of classifier ensembles for handwritten word recognition using feature selection algorithms, *IWFHR 2002* on January 15, 2002.
- [6] B. Minaei-Bidgoli, G. Kortemeyer, W. F. Punch, Mining Feature Importance: Applying Evolutionary Algorithms within a Web-Based Educational System, *Proc. of the Int. Conf. on Cybernetics and Information Technologies, Systems and Applications, CITSA 2004*.
- [7] L. I. Kuncheva, *Combining Pattern Classifiers, Methods and Algorithms*, New York: Wiley, 2005.
- [8] L. Shapley and B. Grofman, Optimizing group judgmental accuracy in the presence of interdependencies, *Public Choice*, 43:329-343, 1984.
- [9] L. K. Hansen, P. Salamon, Neural network ensembles. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 12(10):993-1001, 1990.
- [10] A. Krogh, J. Vedelsdy, Neural Network Ensembles Cross Validation, and Active Learning, In: G. Tesauro, D. Touretzky, T. Leen (Eds.), *Advances in Neural Information Processing Systems*, Volume 7. MIT Press, Cambridge, MA, p.231-238, 1995.
- [11] L. Breiman, Bagging predictors. *Machine Learning*, 24(2):123-140, 1996.
- [12] R.E. Schapire, The strength of weak learn ability, *Machine Learning*, 5(2):1971-227, 1990.
- [13] P. Melville, R. Mooney, Constructing Diverse Classifier Ensembles Using Artificial Training Examples, *Proc. of the IJCAI-2003, Acapulco, Mexico*, p.505-510, 2003.
- [14] L.Lam. Classifier combinations:implementations and theoretical issues. In J. Kittler and F. Roli, editors, *Multiple Classifier Systems*, Vol. 1857 of *Lecture Notes in Computer Science*, Cagliari, Italy, 2000, Springer, pp. 78–86.
- [15] F. Roli and J. Kittler, editors. *Proc. 2nd International Workshop on Multiple Classifier Systems (MCS 2001)*, Vol. 2096 of *Lecture Notes in Computer Science LNCS* Springer-Verlag, Cambridge, UK, 2001.
- [16] E. Alpaydin and C.Kaynak. Cascading classifiers. *KYBERNETIKA*, 34(4):369–374, 1998.