سیستم‌های دسته‌بند چندگانه روش‌های طراحی و قواعد ترکیب شورا

[1] محمدعلی باقری، [2] غلامعلی منتظر و [3] احسان‌اله کبیر
[1 و 2] گروه مهندسی فناوری اطلاعات، دانشکدهٔ فنی و مهندسی، دانشگاه تربیت مدرس
[3] آزمایشگاه سیستمهای الکترونیک، دانشگاه تربیت مدرس

چکیده:

یکی از روش‌های مناسب برای بهبود صحت دسته‌بندی، استفاده از چند دسته‌بند مختلف و سپس ترکیب نتایج خروجی آنهاست که اغلب تحت عنوان «سیستم‌های دسته‌بند چندگانه» یا «سیستم‌های شورایی» خوانده می‌شود. سیستم‌های دسته‌بند چندگانه به‌طور کلّی شامل دو بخش اصلی «ایجاد شورای دسته‌بندها» و «قواعد ترکیب» آنهاست. پژوهش‌گران حوزه‌های مختلف از جمله بازشناسایی الگو، یادگیری ماشینی و آمار استفاده از سیستم‌های شورایی را بررسی کرده‌اند. نبوی کریزی و کبیر اولین مقالهٔ مروری فارسی را در این حوزه ارائه و روش‌های ترکیب دسته‌بندها را معرفی کرده‌اند. با این حال، بررسی روند مقالات اخیر نشان می‌دهد پژوهش در حوزهٔ سیستم‌های شورا بر روش‌های ایجاد شورای دسته‌بندها تمرکز کرده‌اند از این‌رو در این مقاله تلاش شده است، ابتدا چارچوبی برای رویکردهای مختلف ایجاد شورای دسته‌بندها ارائه شود. براساس این ساختار، روش‌های مختلف هر رویکرد معرفی شده و درادامه، روش‌های ترکیب دسته‌بندها به‌اجمال بیان شده است. درپایان، با بررسی روند پژوهش‌های موجود، زمینه‌های تحقیقاتی مناسب در حوزهٔ سیستم‌های شورایی ارائه شده است.

واژگان کلیدی: سیستم‌های دسته‌بند چندگانه، ترکیب دسته‌بندها، طراحی شورا، قواعد ترکیب شورا، گوناگونی

۱- مقدمه

سیستم‌های دسته‌بند چندگانه[1]، MCS که در منابع یادگیری ماشینی با نام‌های مختلفی مانند شورای دسته‌بندها[2]، کمیتهٔ یادگیرنده‌ها[3]، اختلاط خبرگان[4] و نظریهٔ اجماع[5] نیز خوانده می‌شود، یکی از چهار مسیر پیشگام در حوزهٔ یادگیری ماشینی را شکل داده است *(Dietterich 1997)*. درحقیقت استفاده از سیستم‌های شورایی در حوزهٔ یادگیری ماشینی، از طبیعت انسان در استفاده از نظرات مختلف برای تصمیم‌گیری‌های مهم الهام گرفته است. ما نظر افراد مختلف را وزن می‌دهیم و آنها را ترکیب می‌کنیم تا به نتیجهٔ نهایی برسیم *(Polikar 2006)*.

بهبود کارایی دسته‌بندی با رویکرد سیستم‌های شورایی موجب رشد روزافزون استفاده از این سیستم‌ها در حوزه‌های کاربردی بسیاری، ازجمله پزشکی *(Das and Sengur, 2010; Wang et al. 2010; Kuncheva and Rodriguez 2010; Bagherian, Beigy, and Irannia Autumn, 2008 (In Persian)*، تحلیل ریسک *(Finlay 2011; Peng et al. 2011)*، دسته‌بندی متن (Shi et al. 2011; Pino-Mejías et al. 2010)، تشخیص چهره *(Ebrahimpour et al. 2008; Ebrahimpour, Kabir, and Yousefi 2011)*، امنیت *(Govindarajan and Chandrasekaran ; Kim and Kang 2010)*، یادگیری الکترونیکی *(Kotsiantis, Patriarcheas, and Xenos 2010)*، بویایی هوشمند (بینی الکترونیکی) *(Bagheri and Montazer 2011)*، تولید *(Rokach 2008)* و ... شده است.

شاید یکی از اولین پژوهشها در حوزهٔ سیستم‌های دسته‌بند چندگانه مقالهٔ *(Dasarathy and Sheela 1979)* در

[1] multiple classifier systems (MCS)
[2] classifier ensembles
[3] committees of learners
[4] mixture of experts
[5] consensus theory

سال ۱۹۷۹میلادی باشد که بر تقسیم فضای ویژگی با استفاده از دو یا چند دسته‌بند بحث می‌کند. بااین حال، پیشرفت اصلی در این حوزه در دهۀ ۱۹۹۰ میلادی رخ داد. در سال ۱۹۹۰ میلادی، «هانسن»[1] و «سالامُن»[2] نشان دادند که می‌توان عملکرد شبکه‌های عصبی را با استفاده از گروهی از شبکه‌های عصبی با پیکربندی مشابه بهبود داد *(Hansen and Salamon 1990)* در همان سال، پژوهش *(Schapire 1990)* ثابت کرد که دسته‌بندی قوی می‌تواند با ترکیب دسته‌بندهای «ضعیف» (دسته‌بندهایی ساده که صحّت دسته‌بندی آنها تنها کمی بهتر از دسته‌بندی تصادفی است) به‌دست آید؛ این موضوع پایۀ ارائۀ الگوریتم موفّق AdaBoost در سال ۱۹۹۷ میلادی شد.

نبوی کریزی و کبیر اولین مقالۀ مروری فارسی در این حوزه را ارائه و روش‌های ترکیب دسته‌بندها را معرّفی کرده‌اند *(نبوی کریزی و کبیر ۱۳۸۴)*. با این حال بررسی روند مقالات اخیر نشان می‌دهد پژوهش در حوزۀ سیستم‌های شورا بر روش‌های ایجاد شورای دسته‌بندها تمرکز کرده‌اند. از این‌رو در این مقاله ابتدا چارچوبی برای رویکردهای مختلف ایجاد شورای دسته‌بندها ارائه شده است. براساس این ساختار، روش‌های مختلف هر رویکرد معرّفی شده و درادامه، روش‌های ترکیب دسته‌بندها به‌اجمال بیان شده است. در پایان، با بررسی روند پژوهش‌های موجود، زمینه‌های پژوهشی مناسب در این حوزۀ فعال از یادگیری ماشین ارائه شده است. شایان ذکر است تمرکز اصلی این مقاله بر دسته‌بندی با رویکرد شورا است؛ با این حال سیستم‌های شورایی می‌توانند برای مسائل خوشه‌بندی *(Hong et al. 2008)* و رگرسیون *(Gey and Poggi 2006; Tutz and Binder 2007)* نیز به‌کارگرفته شوند.

۱-۱- دلایل استفاده از سیستم‌های دسته‌بند چندگانه

الف) دلایل آماری: کارایی خوب شبکه‌های عصبی یا دیگر دسته‌بندهای خودکار[3] برای نمونه‌های آموزش، قابلیت تعمیم آن را برای نمونه‌های آزمون تضمین نمی‌کند و این از مشکلات مهّم این دسته‌بندها است. بدین معنا که مجموعه‌ای از دسته‌بندها با کارایی آموزش مشابه می‌توانند کارایی متفاوتی را نتیجه دهند. در این موارد، ترکیب خروجی چندین دسته‌بند با میانگین‌گیری می‌تواند ریسک انتخاب نامناسبِ یک دسته‌بند ضعیف را کاهش دهد. متوسط‌گیری ممکن است کارایی بهترین دسته‌بند مجموعه را کاهش دهد؛ امّا درعوض، ریسک انتخابی ضعیف را کم می‌کند. این موضوع به‌طور دقیق همان علّتی است که ما با دیگر خبرگان هم‌فکری و مشورت می‌کنیم تا تصمیم بهتری بگیریم *(Polikar 2006; Dietterich 2000; Kuncheva 2004)*.

ب) حجم بسیار زیاد داده: در کاربردهای خاص ممکن است حجم داده بسیار زیاد باشد؛ به‌طوری‌که یک دسته‌بند نتواند به‌خوبی آنها را تحلیل کند. برای نمونه در بازرسی لوله‌های انتقال گاز با استفاده از روش‌های نشت شار مغناطیسی[4]، در هر ۱۰۰ کیلومتر ۱۰ گیگابایت داده تولید می‌شود *(Polikar 2006)*. آموزش یک دسته‌بند با این حجم بسیار زیاد داده به‌طور معمولی عملی نیست. ثابت شده است که تقسیم داده به زیرمجموعه‌های کوچک‌تر، آموزش دسته‌بندهای مختلف با افرازهای مختلف داده و ترکیب خروجی آنها، اغلب رویکردی مؤثرتر است *(Polikar 2006)*.

ج) حجم کم داده: سیستم‌های دسته‌بند چندگانه همچنین می‌توانند در مسائل با تعداد بسیار کم داده نیز به‌کار گرفته شوند. دردسترس‌بودن مجموعه‌های کافی و مناسب از دادۀ آموزش، بیشترین اهمیت را برای الگوریتم‌های دسته‌بندی دارد که رابطۀ پنهان (مکتوم) داده را یاد بگیرند. در غیاب نمونه‌های آموزش کافی، روش‌های «نمونه‌برداری دوباره»[5] می‌توانند برای تولید زیرمجموعه‌هایی تصادفی از نمونه‌های موجود (که هم‌پوشانی نیز دارند) به‌کار گرفته شود؛ هر زیرمجموعه می‌تواند برای آموزش دسته‌بندی متفاوت استفاده شود. ثابت شده که چنین رویکردی بسیار کارا است *(Polikar 2006)*.

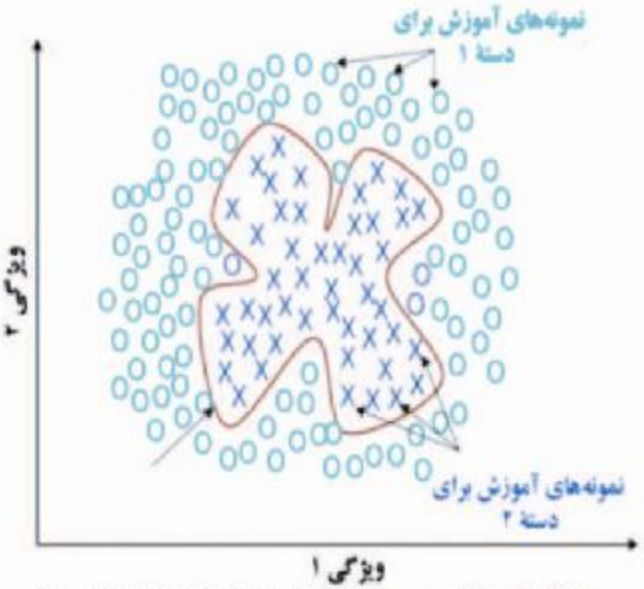
د) ایجاد تقسیم و تسخیر[6]: بدون درنظر گرفتن حجم دادۀ موجود، حلّ برخی مسائل خاص برای یک دسته‌بند بسیار مشکل است. به‌ویژه، مرز تصمیم[7] که نمونه‌های دسته‌های مختلف را از هم جدا می‌کند، ممکن است

[1] Hansen
[2] Salamon
[3] automated classifiers

[4] magnetic flux leakage techniques
[5] resampling
[6] divide and conquer
[7] decision boundary

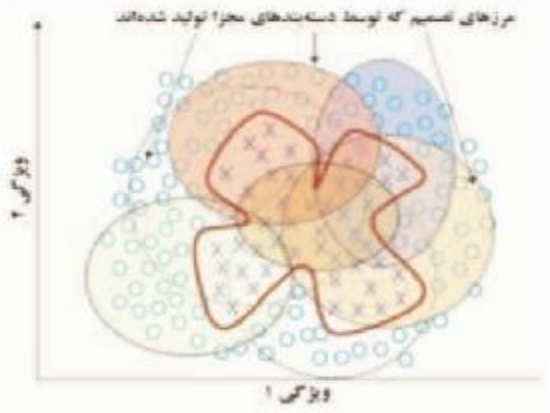
بسیار پیچیده یا بیرون از فضای توابعی قرار گیرد که می‌تواند با مدل دسته‌بند موردنظر ایجاد شود. به‌عنوان مثال، مسألهٔ (شکل ۱) را با دادهٔ دوبعدی و دو دسته در نظر بگیرید یک الگوریتم دسته‌بندی که تنها می‌تواند مرزهای خطّی را یاد بگیرد، نمی‌تواند این مرز غیرخطّی را ایجاد کند. با این حال، ترکیبی مناسب از مجموعه‌ای از این دسته‌بندهای ساده می‌تواند این مرز (یا مشابه چنین مرزهای غیرخطی) را یاد گیرد *(Polikar 2006)*.

فرض کنید دسته‌بندی داریم که می‌تواند مرزهای بیضوی/ دایره‌ای شکل را ایجاد کند. چنین دسته‌بندی نمی‌تواند مرز (شکل ۱) را یاد گیرد. حال مجموعه‌ای از مرزهای تصمیم دایره‌ای را درنظر بگیرید که توسط شورایی از این دسته‌بندها ایجاد شده است (شکل ۲). هر دسته‌بند نمونه‌ها را با توجه به این‌که درون یا بیرون مرز دسته‌بند قرار می‌گیرند، نمونه‌های دستهٔ اول (O) و نمونه‌های دستهٔ دوم (X) برچسب می‌زند. تصمیمی براساس رأی‌گیری اکثریت[1] از تعدادی کافی از این دسته‌بندها به سادگی می‌تواند این مرز غیرخطی پیچیده را یاد گیرد. از سوی دیگر، این سیستم دسته‌بندی با تقسیم فضای داده به افرازهای کوچک‌تر که یادگیری آنها ساده‌تر است، رویکرد ایجاد تقسیم و تسخیر را دنبال می‌کند که هر دسته‌بند تنها یک افراز ساده‌تر را یاد می‌گیرد. بدین ترتیب ترکیبی مناسب از دسته‌بندهای مختلف می‌تواند مرز تصمیم پیچیده را تقریب بزند.



(شکل ۱): مرز تصمیم پیچیده که نمی‌تواند با دسته‌بندهای خطی یا دایره‌ای یادگرفته شود. برگرفته از *(Polikar 2006)*

[1] majority voting



(شکل ۲): دسته‌بندهای چندگانه فضای تصمیم را پوشش می‌دهند. برگرفته از *(Polikar 2006)*

ه) همجوشی داده[2]: اگر مجموعه داده‌های موجود از منابع مختلفی به‌دست آمده باشند که طبیعت ویژگی‌هایشان متفاوت است (ویژگی‌های ناهمگن[3])، یک دسته‌بند واحد معمولاً نمی‌تواند برای یادگیری اطلاعات موجود در همهٔ نمونه‌ها استفاده شود. برای نمونه برای تشخیص یک بیماری عصبی، پزشک به‌طور معمول از اطّلاعات مختلف مانند تصاویر MRI، اطلاعات ثبت‌نگارهٔ مغز (الکتروآنسفالوگرامی)[4] و آزمایش‌های خون استفاده می‌کند. هر آزمایش، نمونه‌هایی با تعداد و نوع ویژگی‌های متفاوت تولید می‌کند؛ این اطلاعات نمی‌تواند با هم برای آموزش یک دسته‌بند استفاده شود. دراین‌گونه موارد، نمونه‌های از یک نوع می‌تواند برای آموزش دسته‌بندهای متفاوت استفاده شود و خروجی آنها درنهایت ادغام می‌شود. کاربردهایی را که در آن نمونه‌هایی از منابع مختلف با هم ترکیب می‌شوند تا تصمیمی آگاهانه‌تر گرفته شود کاربردهای «هم‌جوشی داده» گویند و استفاده از رویکردهای شورا برای این کاربردها نتایج خوبی را نشان داده است *(Bloch 1996; Polikar 2006)*.

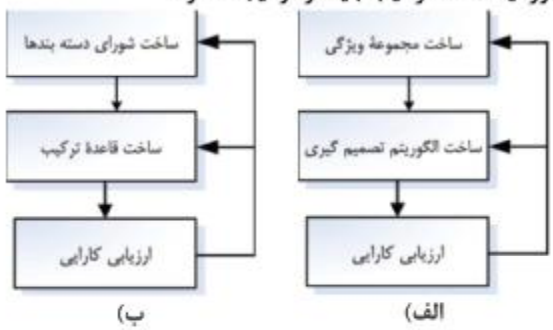
۱-۲- طراحی سیستم‌های دسته‌بند چندگانه

از نگاه کلّی، طراحی سیستم‌های دسته‌بند چندگانه بر ادغام چند دسته‌بند با یک قاعدهٔ ترکیب استوار است. فرآیند طراحی شامل سه مرحله است *(Yang, Qin, and Huang 2004)*: الف) ساخت شورای دسته‌بندها، ب) طراحی قاعدهٔ ترکیب[5]، ج) ارزیابی کارایی.

این فرایند یادآور طراحی سیستم دسته‌بند واحد[6]، SCS، است که شامل سه مرحله است: انتخاب ویژگی‌ها،

[2] data fusion
[3] heterogeneous features
[4] EEG (electroencephalogram)
[5] combination rule
[6] single classifier system

ساخت الگوریتم تصمیم‌گیری و ارزیابی کارایی نهایی. فرایند طراحی این دو سیستم در (شکل ۳) نشان داده شده است (Giacinto and Roli, 2001). در این دو سیستم حلقهٔ بازخوردی از آخرین مرحله به مراحل قبلی وجود دارد. مفهوم این حلقه در سیستم SCS روشن است، امّا در MCS به این معنی است که اگر کارایی خروجی مناسب نیست، شورا یا قاعدهٔ ترکیب باید از نو ایجاد شود.



(شکل ۳): فرآیند طراحی الف) سیستم دسته‌بند واحد و ب) سیستم دسته‌بند چندگانه

در SCS، مجموعهٔ ایده‌آلی از ویژگی می‌تواند تا حدّ زیادی ایجاد الگوریتم تصمیم را ساده کند و الگوریتم قوی نیز می‌تواند حتّی با یک مجموعهٔ ویژگی ضعیف، نتایج خوبی را بدهد. همین موضوع در MCS نیز صادق است: مجموعهٔ ایده‌آلی از دسته‌بندها که شامل دسته‌بندهایی گوناگون[1] و دقیق[2] هستند می‌تواند تا حدّ زیادی ایجاد قاعدهٔ ترکیب را ساده کند و قاعدهٔ ترکیبی قوی می‌تواند حتّی با شورای دسته‌بندهای ضعیف، نتایج خوبی را بدهد. این دو جنبهٔ مختلف، دو خطّ سیر موازی پژوهش در حوزهٔ ترکیب دسته‌بندها هستند که در منابع یادگیری ماشینی به‌ترتیب تحت عنوان‌های «طراحی شورای دسته‌بندها» و «طراحی قاعدهٔ ترکیب» (یا بهینه‌سازی تصمیم[3]) آمده است. این دو موضوع اساسی در سیستم‌های شورایی در بخش‌های دوم و سوم معرفی شده است.

۱-۳- گوناگونی[4]: اساس سیستم‌های دسته‌بند چندگانه

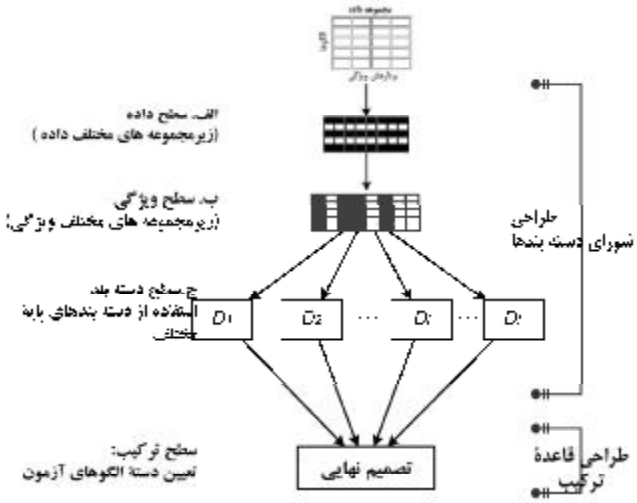
نتایج نظری (Hansen and Salamon Oct. 1990; Krogh and Vedelsby 1995) و تجربی (Hashem 1997; Kuncheva, Skurichina, and Duin 2000) پژوهش‌ها نشان می‌دهد زمانی یادگیری چندگانه از یادگیری بهترین دسته‌بند پایه بهتر است که دسته‌بندهای پایه دارای کارایی قابل قبول بوده و گوناگون در خطا باشند. دو دسته‌بند، زمانی گوناگونی در خطا دارند که الگوهایی که به‌صورت نادرست دسته‌بندی می‌کنند، متفاوت باشند. تفاوت در موارد خطای دسته‌بندهای پایه، باعث می‌شود دسته‌بندها خطاهای یکدیگر را بپوشانند و به همین علّت گوناگونی در خطا، از نکات اساسی در موفقیت سیستم دسته‌بندی چندگانه است. به‌طور کلّی سه روش برای ایجاد شورایی از دسته‌بندهای گوناگون وجود دارد:

الف) *رویکرد زیر نمونه*[5]: ساخت دسته‌بندهای مختلف با استفاده از زیرمجموعه‌های مختلف آموزش که با نمونه‌گیری تصادفی از نمونه‌های آموزش اوّلیه به‌دست آمده‌اند. روش‌های نمونه‌برداری bagging (Breiman 1996) و boosting (Freund and Schapire 1996) مهم‌ترین روشها با این رویکرد هستند.

ب) *رویکرد زیر فضا*[6]: ساخت دسته‌بندهای مختلف با زیرمجموعه‌های ویژگی مختلف.

ج) استفاده از رویکردهای یادگیری مختلف مانند شبکهٔ عصبی، دسته‌بندهای بیز، درخت‌های تصمیم.

(شکل ۴) چارچوب سیستم دسته‌بند چندگانه را نشان می‌دهد. از نگاهی دیگر، سه روش ایجاد گوناگونی، روش‌های طراحی شورای دسته‌بندها در سه سطح مختلف هستند. ساختار نگارش این مقاله بر این اساس تنظیم شده است. دو بخش اصلی سیستم‌های دسته‌بند چندگانه (شامل طراحی شورای دسته‌بندها و ایجاد قواعد ترکیب) در بخش‌های دوم و سوم معرفی شده است. همچنین سه رویکرد ایجاد تنوع دسته‌بندها، به‌تفکیک در زیربخش‌های بخش دوم آورده شده است. در بخش چهارم، روش‌های اندازه‌گیری گوناگونی در سیستم‌های شورایی آورده شده و در پایان، زمینه‌های پژوهشی پیشگام در این حوزه بیان شده است.



(شکل ۴): چارچوب سیستم دسته‌بند چندگانه

[1] diverse
[2] accurate
[3] decision optimization
[4] diversity
[5] subsample
[6] subspace

۲- طراحی شورای دسته‌بندها

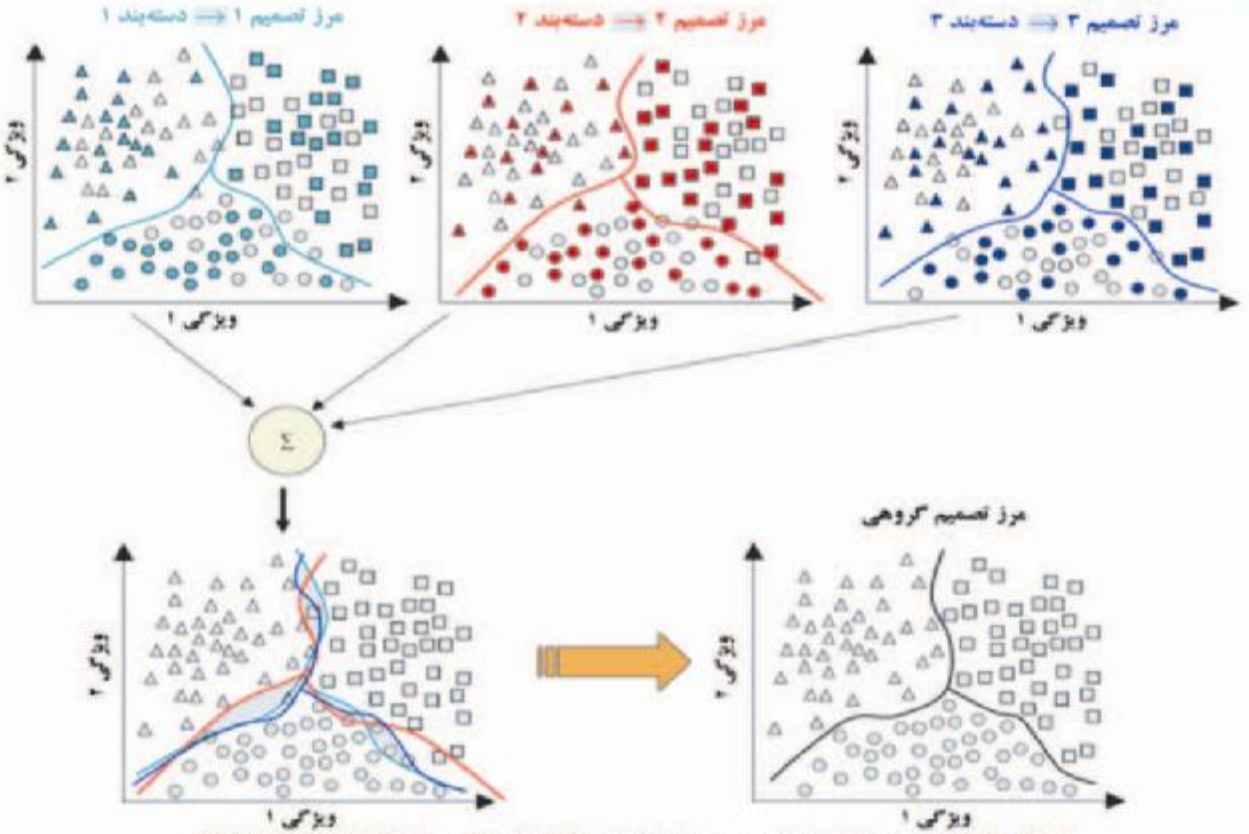
طراحی شورای دسته‌بندها، که در برخی منابع «بهینه‌سازی پوشش»[1] نیز نامیده می‌شود، ساخت مجموعه‌ای دقیق و گوناگون از دسته‌بندهاست. بهینه‌سازی پوشش تلاش می‌کند مجموعه‌ای از دسته‌بندهای پایه را بسازد که متقابلاً مکمل همدیگر بوده و بتوانند با ترکیب همدیگر، کارایی بهتری داشته باشند. برای طراحی شورای دسته‌بندها، باید به دو سؤال وابسته به هم پاسخ داده شود: ۱) چگونه دسته‌بندهای مجزا (دسته‌بندهای پایه[2]) ایجاد شوند؟ و۲) چگونه این دسته‌بندها با یکدیگر متفاوت شوند؟ پاسخ‌های این دو پرسش، درنهایت گوناگونی دسته‌بندها را مشخص می‌کند و در نتیجه بر کارایی کلّ سیستم اثر می‌گذارد *(Yang, Qin, and Huang, 2004)*.

همچنان که اشاره شد، ایجاد شورای دسته‌بندها باید به‌دنبال بهبود گوناگونی باشد. با این حال الگوریتم‌های طراحی شورا، به‌طور عمومی برای بیشینه‌کردن معیار گوناگونی خاص طراحی نشده‌اند؛ بلکه گوناگونی بیشتر از طریق روش‌های نمونه‌برداری مختلف (تا حدّی ابتکاری) و یا انتخاب پارامترهای مختلف آموزش به‌دست می‌آید *(Polikar, 2006)*.

معیارهای مختلف گوناگونی (بخش چهارم) می‌توانند برای مقایسهٔ شوراهایی که با الگوریتم‌های متفاوت ساخته شده‌اند، به‌کارگرفته شوند.

۲-۱- رویکرد زیرنمونه: سطح داده

محبوب‌ترین روش ایجاد گوناگونی، استفاده از مجموعه‌های دادهٔ مختلف برای آموزش دسته‌بندهای پایه است. این مجموعه‌ها، اغلب توسط روش‌های نمونه‌برداری دوباره حاصل می‌شود؛ به‌طوری‌که مجموعه‌های آموزش با انتخاب تصادفی، به‌طور معمول با جایگذاری، از میان تمام الگوهای آموزش تولید می‌شوند. این موضوع در (شکل ۵) نشان داده شده است. زیرمجموعه‌های دادهٔ تصادفی و هم‌پوشان[3] برای آموزش سه دسته‌بند انتخاب شده است که سه مرز تصمیم متفاوت را شکل می‌دهند. آنگاه این مرزها با یکدیگر ترکیب شده‌اند تا بتوان به دسته‌بندی دقیق‌تری دست یافت.



(شکل ۵): ترکیب دسته‌بندها با زیرمجموعه‌های دادهٔ مختلف آموزش . برگرفته از (Polikar 2006)

[1] coverage optimization
[2] base classifiers
[3] overlapping

ازآنجا که برای آموزش دسته‌بندها از الگوهایی به‌طور اساسی مشابه استفاده شده است، برای اطمینان از اینکه مرزهای مجزا به حدّ کافی متفاوت هستند، از *دسته‌بندهای ناپایدار*[1] به‌عنوان دسته‌بندهای پایه استفاده می‌شود. زیرا این دسته‌بندهای ناپایدار می‌توانند مرزهای تصمیمی تولید کنند که حتّی برای آشفتگی‌های[2] کم در پارامترهای آموزش خود، تا حدّکافی متفاوت باشند. اگر زیرمجموعهٔ دادهٔ آموزش بدون جایگذاری ایجاد شده باشد، این روش «تقسیم *k*- بخشی داده»[3] نیز نامیده می‌شود. در این روش، تمام مجموعه داده به *k* بلوک تقسیم می‌شود و هر دسته‌بند، یک بلوک را برای آزمون و *k-1* بلوک دیگر را برای آموزش استفاده می‌کند.

مهم‌ترین روش‌های رویکرد زیرنمونه در ادامه معرّفی شده است:

□ **Bagging**: این روش توسط «بریمَن»[4] در سال ۱۹۹۶ میلادی ابداع شده *(Breiman 1996)* و یکی از اوّلین الگوریتم‌های ایجاد شوراست. همچنین این روش یکی از شهودی‌ترین و ساده‌ترین روش‌های ایجاد سیستم دسته‌بند چندگانه است که در بسیاری از مسائل از کارایی بسیار خوبی برخوردار است. روش Bagging با نمونه‌برداری *باجایگذاری*[5] از مجموعهٔ آموزش اولیه، زیرمجموعه‌های آموزش مختلفی می‌سازد که هر نمونه شانس یکسان برای انتخاب‌شدن دارد. اندازهٔ هر زیرمجموعهٔ آموزش برابر با مجموعهٔ آموزش اوّلیه است. بنابراین بعضی نمونه‌ها در مجموعه‌های آموزش مختلف نیستند؛ درحالی‌که برخی نمونه‌ها ممکن است بیش از یک‌بار ظاهر شوند. این فرایند، تجمیع خودراه‌انداز (*bootstrap aggregating*) و به‌اختصار *bagging* خوانده شده است *(Breiman 1996)*.

روش bagging برای دسته‌بندهای ناپایدار[6] نتایج بهتری می‌دهد؛ دسته‌بندهای ناپایدار (مانند شبکهٔ عصبی یا درخت تصمیم)، دسته‌بندهایی هستند که درصورت تغییر کوچکی در مجموعهٔ آموزش یا پارامترهای الگوریتم، خروجی آنها تغییر زیادی می‌کند. این روش به‌ویژه برای زمانی که الگوهای موجود کم هستند، بسیار مطلوب است. تحلیل نظری و تجربی روش bagging در پژوهش *(Fumera, Roli, and Serrau 2008)* به‌تفصیل آورده شده است.

□ **جنگل تصادفی**: روش *جنگل تصادفی*[7]، یکی از انواع الگوریتم bagging است که چون از روش درخت تصمیم به‌عنوان دسته‌بندهای پایه استفاده کرده، بدین نام خوانده می‌شود *(Breiman 2001)*. جنگل تصادفی از درخت‌های تصمیم مجزا ساخته می‌شود که نمونه‌های آموزش آن به‌طور تصادفی تغییر می‌کنند. این تغییر می‌تواند همانند روش bagging با نمونه‌برداری از الگوهای آموزش باشد؛ همچنین می‌توانند همانند رویکرد زیرفضای تصادفی براساس زیرمجموعه‌های ویژگی متفاوت باشد.

□ **Boosting**: این روش یکی از مهم‌ترین پیشرفت‌های سال‌های اخیر در حوزهٔ سیستم‌های شورایی است *(Freund and Schapire 1996)*. مشابه روش bagging، این روش مجموعه‌ای از دسته‌بندها را با نمونه‌برداری دوباره از داده ایجاد و درنهایت با «رأی اکثریت» ترکیب می‌کند. امّا در روش تقویتی برای دسته بندهای متوالی، نمونه‌برداری دوباره با راهبرد انتخاب نمونه‌هایی است که حاوی بیشترین اطلاعات باشند. به‌بیان دیگر، درحالی‌که در روش bagging نمونه‌ها با احتمال مساوی انتخاب می‌شوند و دسته‌بندهای تولید شده مستقل هستند، روش boosting به نمونه‌هایی که اشتباه دسته‌بندی شده‌اند اهمیت بیشتری می‌دهد و دسته‌بندها را به‌طور متوالی ایجاد می‌کند. الگوریتم boosting مجموعه‌ای از وزن‌ها را برای نمونه‌های مجموعهٔ آموزش ذخیره می‌کند. در گام اوّل، تمام وزن‌ها مساویند و دسته‌بند اوّل با این مجموعهٔ آموزش ایجاد می‌شود. سپس وزن‌ها براساس کارایی این دسته‌بند تغییر می‌کند؛ وزن نمونه‌هایی که اشتباه دسته‌بندی شده‌اند افزایش می‌یابد و بالعکس. لذا الگوریتم boosting، الگوریتمی تکراری است که در هر چرخه از تکرار، وزن‌ها دوباره تنظیم می‌شوند. این روند می‌تواند در مسائل یادگیری، که دارای نمونه‌هایی با پیچیدگی متفاوت هستند، کارا باشد. همچنین در روش bagging دسته‌بندها نباید پایدار باشند؛ اما در روش boosting می‌توان از دسته‌بندهای پایدار به شرط داشتن خطای دسته‌بندی کمتر از ۰/۵ استفاده کرد.

از مهم‌ترین رویکردهای boosting، می‌توان به رویکردهای فیلترینگ، نمونه‌برداری و وزن‌دهی مجدد[8] اشاره کرد *(Haykin and Network 2004)*. رویکرد اوّل، شامل روش‌های فیلترکردن نمونه‌های آموزش با نسخه‌های مختلف الگوریتم‌های یادگیری هستند. در رویکرد نمونه‌برداری، که روش مطرح AdaBoost نیز براساس این رویکرد ارائه شده، دسته‌بندهای مختلف با استفاده از مجموعه‌های آموزش با اندازهٔ ثابت ایجاد می‌شوند. این نمونه‌ها در فرآیند آموزش، با

[1] unstable classifiers
[2] perturbations
[3] *k*-fold data split
[4] Breiman
[5] with replacement
[6] unstable
[7] Random Forests
[8] reweighting

نمونه‌برداری براساس توزیع احتمال آنها انتخاب می‌شوند. در رویکرد وزن‌دهی مجدّد نیز اندازهٔ مجموعهٔ آموزش برای دسته‌بندها ثابت است؛ اما این رویکرد، فرض می‌کند که الگوریتم یادگیری ضعیف شورا می‌تواند نمونه‌ها را با وزن‌های مختلف بپذیرد.

□ **AdaBoost:** در سال ۱۹۹۷ میلادی، «فروند»[1] و «شاپیر»[2] الگوریتم AdaBoost را معرّفی کردند (*Freund and Schapire, 1997*) که از آن زمان توجّه زیادی را در حوزهٔ هوش مصنوعی و یادگیری ماشینی به خود جلب کرده است. در میان نسخه‌های مختلف این الگوریتم، AdaBoost.M1 و AdaBoost.R بیشترین استفاده را دارند که به‌ترتیب برای مسائل دسته‌بندی چنددسته‌ای[3] و رگرسیون استفاده می‌شوند. شبه‌کد AdaBoost.M1 در ادامه آورده شده است.

□

ورودی:

■ نمونه‌های آموزش $(S = [(x_i, y_i)], i = 1, \dots, N)$ با برچسب‌های صحیح $y_i \in \Omega = \{\omega_1, \omega_1, .., \omega_c\}$ ،

■ الگوریتم یادگیری ضعیف (WeakLearn)

■ عدد صحیح L برابر با تعداد دسته‌بندها (مساوی با تعداد تکرارها)

۱. مقدار دهی اولیه

■ تنظیم وزن نمونه‌ها $\mathbf{w}^1 = [w_1, \dots, w_N]$ ؛ $w_j^1 \in [0,1]$ ؛ $\sum_{j=1}^{N} w_j^1 = 1$

(معمولاً $w_j^1 = \frac{1}{N}$)

۲. انجام بده: (به‌ازای $k = 1, \dots, L$)

۱. زیرمجموعه‌ای از نمونه‌های آموزش (S_k) براساس وزنهای $\mathbf{w}^k$ انتخاب کن.

۲. دسته‌بند D_k را با استفاده از زیرمجموعهٔ S_k ایجاد کن.

۳. خطای وزن‌دار شورای دسته‌بند در گام k را محاسبه کن:

$$\varepsilon_k = \sum_{j=1}^{N} w_j^k l_k^j \qquad (۱)$$

($l_k^j = 1$ اگر دسته‌بند D_k نمونهٔ z_j را اشتباه دسته‌بندی کند و درغیراین‌صورت $l_k^j = 0$)

□ اگر $\varepsilon_k = 0$ یا $\varepsilon_k \geq 0.5$ از دسته‌بند D_k صرف‌نظر کن؛ و وزنهای w_j^k را برابر $\frac{1}{N}$ قرار بده.

در غیراین‌صورت خطای هنجارشدهٔ شورای دسته‌بند در گام k (β_k) را محاسبه کن:

$$\beta_k = \frac{\varepsilon_k}{1 - \varepsilon_k} \quad ، \quad \varepsilon_k \in (0, 0.5) \text{ که} \qquad (۲)$$

۴. وزن نمونه‌ها (برای گام بعد) را بروزرسانی کن:

$$w_j^{k+1} = \frac{w_j^k \beta_k^{(1-l_k^j)}}{\sum_{i=1}^{N} w_i^k \beta_k^{(1-l_k^j)}} \qquad (۳)$$

□

[1] Freund
[2] Schapire
[3] multiclass

۳. خروجی مرحلهٔ آموزش: شورای دسته‌بند و وزنهای هر دسته‌بند $\beta_1, \dots, \beta_L$

آزمون: رأی‌گیری حداکثری وزن‌دار (Weighted Majority Voting)

برای نمونهٔ برچسب نخورده $\mathbf{x}$:

۴. رأی (امتیاز) هر دسته را برای این نمونه محاسبه کن:

$$\mu_t^{(x)} = \sum_{D_k(x)=w_t} ln\left(\frac{1}{\beta_k}\right) \qquad (۴)$$

۵. دسته‌ای که بیشترین رأی (امتیاز) را دارد، به‌عنوان دستهٔ نمونهٔ $\mathbf{x}$ انتخاب می‌شود.

این الگوریتم توزیع وزنها ، w_i^k، برای نمونه‌های آموزش را $x_i, i = 1, \dots, N$ حفظ می‌کند. براساس این توزیع، زیرمجموعه‌های آموزش در هر گام، S_k ، برای آموزش دسته‌بند بعدی انتخاب می‌شوند. توزیع وزن نمونه‌ها در ابتدا یکنواخت است؛ لذا هر نمونه شانس انتخاب یکسانی در زیرمجموعهٔ اول آموزش دارد. خطای آموزشِ دسته‌بندِ D_k ، ε_k، نیز بر اساس توزیع وزن نمونه‌ها محاسبه می‌شود؛ به‌طوری که ε_k مجموع وزن نمونه‌هایی است که توسط دسته‌بندِ D_k اشتباه دسته‌بندی شده‌اند (رابطهٔ ۱). سپس خطای نرمال‌شدهٔ دسته‌بند ، β_k ، محاسبه می‌شود؛ به‌طوری که برای هر $0 < \varepsilon_k < 1/2$ داریم: $0 < \beta_k < 1$.

رابطهٔ ۳ قاعدهٔ روزآمدکردن وزن‌ها را بیان می‌کند: وزن نمونه‌هایی که به‌کمک دسته‌بند جاری صحیح دسته‌بندی شده‌اند با ضریب β_k کاهش می‌یابد؛ درحالی‌که وزن نمونه‌هایی که اشتباه دسته‌بندی شده‌اند تغییر نمی‌کند. زمانی که وزن‌های جدید، هنجار شوند (درنتیجه $\mathbf{w}^{k+1}$ توزیع صحیحی خواهد بود)، وزن نمونه‌های اشتباه دسته‌بندی‌شده افزایش می‌یابد. ازاین‌رو در هر تکرار، الگوریتم برروی نمونه‌هایی سخت‌تر تمرکز می‌کند. از آنجا که لازم است خطای الگوریتم یادگیری (دسته‌بند) کمتر از ½ باشد، این الگوریتم تضمین می‌کند که حداقل یک نمونه که در قبل اشتباه دسته‌بندی‌شده بود، درست دسته‌بندی شود. زمانی که تعداد L دسته‌بند ایجاد شود، الگوریتم وارد مرحلهٔ آزمون می‌شود. برخلاف bagging و boosting، الگوریتم AdaBoost از الگوی رأی‌گیری اکثریت وزن‌دار استفاده می‌کند. ایدهٔ شهودی این موضوع آن است که دسته‌بندهایی که در مرحلهٔ آموزش کارایی بهتری داشته‌اند و خطای دسته‌بندی β_k کمتری داشته‌اند، وزن بیشتری می‌گیرند. ازاین‌رو، معکوس مقدار β_k معیاری ازکارایی است. بااین حال چون β_k خطای آموزش است، اغلب نزدیک به صفر است و معکوس آن عددی بسیار بزرگ خواهد بود. برای جلوگیری از بروز مشکل ناپایداری، به‌طور معمول لگاریتم $1/\beta_k$ به‌عنوان وزن دسته‌بند استفاده می‌شود. درنهایت،

دسته‌ای که بیشترین رأی (امتیاز) را از دسته‌بندها بگیرد، تصمیم شورای دسته‌بندها برای نمونهٔ آزمون خواهد بود.

□ **Wagging**: این روش نسخه‌ای از الگوریتم bagging است که برای آموزش هر دسته‌بند، از همهٔ نمونه‌ها استفاده می‌کند؛ امّا وزن هر نمونه به‌طور تصادفی تعیین می‌شود *(Bauer and Kohavi, 1999)*. درحقیقت bagging را می‌توان نوعی از الگوریتم wagging با توزیع وزنی پواسون دانست؛ چرا که هر نمونه چند بار (با مقادیر گسسته) در الگوریتم استفاده می‌شود.

□ **Rotation Forest**: این روش از دیگر روش‌های موفق ایجاد شورای دسته‌بندهاست که در سال ۲۰۰۶ میلادی مطرح شده است *(Rodrguez, Kuncheva, and Alonso, 2006)*. در این روش با اعمال تحلیل مؤلفه‌های اصلی[1] بر روی زیرمجموعهٔ آموزش هر دسته‌بند پایه، محورهای اولیهٔ ویژگی می‌چرخد. ابتدا زیرمجموعهٔ ویژگی F، به‌طور تصادفی به K زیرمجموعه تقسیم و سپس PCA بر روی هر زیرمجموعه اعمال می‌شود. از این زیرمجموعه‌ها برای آموزش دسته‌بندهای پایه استفاده می‌شود. در این روش، تمام مؤلفه‌های اصلی حفظ می‌گردد تا پراکندگی اطلاعات داده حفظ شود. لذا K محور چرخش‌یافته به‌دست می‌آید تا ویژگی‌های جدیدی را برای دسته‌بندهای پایه بسازند. ایدهٔ اصلی این روش بهبود همزمان گوناگونی و صحّت هر دسته‌بند در شوراست: گوناگونی از طریق استخراج ویژگی برای هر دسته‌بند پایه با استفاده از PCA و صحّت با نگهداری تمام مؤلّفه‌های اصلی و نیز استفاده از تمام مجموعهٔ داده برای هر دسته‌بند به‌دست می‌آید. در پژوهش *(Nanni and Lumini, 2009)* توانایی این روش برای تشخیص دانش‌آموزان دارای مشکل ناتوانی یادگیری نشان داده شده؛ با این تفاوت که از «تحلیل مؤلفه‌های مستقل»[2] به‌جای PCA استفاده شده است.

□ **RotBoost**: این روش براساس ترکیب روش‌های Rotation Forest و AdaBoost ارائه شده است *(Zhang and Zhang, 2008)*. آزمون این روش بر روی ۳۶ مجموعه دادهٔ UCI نشان داده که کارایی آن بهتر از روش‌های Rotation Forest، AdaBoost، MultiBoost و Bagging است.

□ **اختلاط خبرگان**[3]: در این روش که در سال ۱۹۹۱ میلادی مطرح شده است *(Jacobs et al., 1991)*، ابتدا دسته‌بندهای $D_1, D_2, \ldots, D_L$ شورای دسته‌بندها را تشکیل می‌دهند؛ در مرحلهٔ بعد، دسته‌بند D_{L+1}، برای تخصیص وزن برای ترکیب دسته‌بندها استفاده می‌شود. دسته‌بند مرحلهٔ دوم که به‌طور معمول بر اساس شبکهٔ عصبی است، صرفاً برای توزیع وزن به دسته‌بندهای مرحلهٔ اوّل است و شبکهٔ میانجی[4] نامیده می‌شود. نکتهٔ قابل توجّه در این روش این است که ورودی این شبکه، داده‌های آموزش است و لذا وزن‌های قاعدهٔ ترکیب براساس نمونه‌های مسأله به‌دست می‌آیند و قاعدهٔ ترکیب پویا خواهد بود. آموزش شبکهٔ میانجی به‌طور معمول با استفاده از روش بیشترین شیب[5] یا الگوریتم بیشینه‌سازی مورد انتظار[6] است.

روش اختلاط خبرگان می‌تواند به‌عنوان الگوریتم انتخاب دسته‌بند نیز دیده شود. دسته‌بندهای مجزا، خبرگانی در بخشی از فضای ویژگی هستند و قاعدهٔ ترکیب، مناسب‌ترین دسته‌بند را انتخاب می‌کند یا وزدسته‌بندها بر اساس خبرگی‌شان برای دسته‌بندی نمونهٔ آزمون تعیین می‌شود.

در پژوهش *(Jordan and Jacobs, 1994)*، روش پایهٔ اختلاط خبرگان توسعه داده شده و «اختلاط خبرگان سلسله‌مراتبی»[7] نامیده شده است. این روش، فضای مسأله را به‌زیرفضاهایی تجزیه می‌کند و به‌طور سلسله‌مراتبی هر زیرفضا را به زیرفضاهایی کوچک‌تر تقسیم می‌کند.

۲-۲- رویکرد زیرفضا: سطح ویژگی

یکی از رویکردهای مؤثر برای ایجاد شورایی از دسته‌بندهای گوناگون، استفاده از زیرمجموعه‌های ویژگی مختلف و آموزش دسته‌بندهای پایه با این زیرمجموعه‌هاست که «شورای انتخاب ویژگی»[8] گفته می‌شود. هدف شورای انتخاب ویژگی، یافتن زیرمجموعه‌های ویژگی است که صحّت دسته‌بندی خوبی داشته و تا حدّ ممکن گوناگون باشند.

زمانی‌که تعداد نمونه‌های مجموعهٔ آموزش نسبت به ابعاد داده کم است، این رویکرد به‌طور معمول گزینهٔ بهتری در مقایسه با روش‌های رویکرد زیرنمونه است *(Skurichina and Duin, 2002)*. در این رویکرد، ابعاد زیرفضا کمتر از ابعاد فضای ویژگی اولیه خواهد بود؛ درحالی‌که تعداد نمونه‌های

[1] principal components analysis (PCA)
[2] independent component analysis (ICA)
[3] mixture of experts
[4] gating network
[5] gradient decent
[6] Expectation Maximization (EM)
[7] Hierarchical Mixtures of Experts (HME)
[8] ensemble feature selection

آموزش ثابت می‌ماند. بنابراین اندازهٔ نسبی مجموعهٔ آموزش افزایش می‌یابد. همچنین زمانی که نمونه‌های مسأله دارای ویژگی‌های زائد زیادی است، استفاده از روش زیرفضای تصادفی نسبت به فضای ویژگی اوّلیه نتایج بهتری به‌دست آید. تصمیم ادغامی این دسته‌بندها به‌طور معمول برتر از تصمیم دسته‌بند واحدی است که با مجموعهٔ آموزش اوّلیه از تمام فضای ویژگی ساخته می‌شود. «هُو»[1] نشان داد درحالی‌که اغلب روش‌های دسته‌بندی با مشکل ابعاد زیاد داده مواجه‌اند، روش زیرفضای تصادفی می‌تواند از این موضوع بهره‌مند شود *(Ho, 1998)*.

نتایج پژوهش‌های اخیر نشان می‌دهد الگوریتم مناسب انتخاب زیرمجموعه‌های ویژگی، کارایی و صحت روش‌های زیرفضا را به‌طور قابل ملاحظه‌ای بهبود می‌دهد *(Chen, Lin, and Chou, 2011)*؛ چرا که کاهش فضای ویژگی «مشکل نمونه‌های با ابعاد زیاد»[2] را کاهش داده و فرایند آموزش و آزمون سریع‌تر انجام خواهد شد *(Bryll, Gutierrez-Osuna, and Quek, 2003; Tumer and Oza, 2003)*. در الگوریتم پیشنهادی «هُو»، زیرمجموعه‌های ویژگی به‌طور تصادفی ایجاد می‌شوند. با این حال انتخاب زیرمجموعه‌ها می‌تواند تصادفی نباشد. سه راهکار اصلی برای ایجاد شورای دسته‌بندها با زیرمجموعه‌های ویژگی مختلف می‌توان معرّفی کرد *(Rokach, 2008)*: الف) انتخاب تصادفی[3] زیرمجموعه‌های ویژگی؛ ب) انتخاب فروکاست‌های[4] ویژگی؛ و ج) انتخاب برمبنای کارایی زیرمجموعه‌ها[5].

الف) راه‌کار انتخاب تصادفی: ساده‌ترین روش ایجاد زیرمجموعه‌های ویژگی مختلف براساس انتخاب تصادفی ویژگی‌ها است. «هو» *(Ho, 1998)* جنگلی از درخت‌های تصمیم با انتخاب شبه‌تصادفی ویژگی‌ها ایجاد کرد و نشان داد انتخاب تصادفی زیرمجموعه‌های ویژگی می‌تواند روش مؤثّری باشد؛ چرا که گوناگونی دسته‌بندها، کم‌بودن میزان صحت دسته‌بندی آنها را جبران می‌کند. این الگوریتم، «روش زیرفضای تصادفی»[6] نامیده می‌شود. در این روش، تعداد معیّنی زیرمجموعهٔ ویژگی m تایی به‌طور تصادفی از بردار ویژگی n بُعدی مسأله انتخاب می‌شود $(m < n)$.

سپس آموزش دسته‌بندها با این مجموعه‌ها انجام می‌شود و سرانجام در مرحله آزمون، تصمیم اعضای شورا با قاعده ترکیب مناسبی ادغام می شوند.

در روش پیشنهادیِ «هو»، تعداد ویژگی‌های هر زیرمجموعه ، m، مقدار ثابت و به‌طور تقریبی برابر با نصف ویژگی‌های موجود ، n، است. به‌جای انتخاب تعدادی ثابت از ویژگی‌ها در هر زیرمجموعه، زیرمجموعه‌های ویژگی می‌تواند تعداد ویژگی‌های متفاوتی داشته باشد. با این تغییر، هر ویژگی بخت یکسانی برای انتخاب شدن در زیرمجموعهٔ ویژگی دارد. همان‌گونه که پژوهش *(Tsymbal and Puuronen, 2002)* نشان می‌دهد این روش، گوناگونی شورای دسته‌بندها را افزایش می‌دهد و درنتیجه به کارایی بهتری در دسته‌بندی منجر می‌شود.

«بریل»[7] و همکارانش روش *Attribute Bagging (AB)* را پیشنهاد کردند که مشابه رویکرد «هو» زیرمجموعه‌های تصادفی را با هم ترکیب می‌کرد؛ با این تفاوت که در این روش زیرمجموعه‌های ویژگی بهتر با رویکرد پوششی[8] انتخاب می‌شوند *(Bryll, Gutierrez-Osuna, and Quek, 2003)*. در این روش، با جستجویِ تصادفی ابعاد فضای ویژگی، اندازهٔ مناسب زیرمجموعه‌ها تعیین می‌شوند. سپس زیرمجموعه‌های ویژگی مختلف به‌صورت تصادفی ایجاد شده و زیرمجموعه‌هایی را که نتایج بهتری بر روی نمونه‌های آموزشی داشتند برای دسته‌بندی نمونه‌های آزمون انتخاب می‌شوند. در پژوهشی دیگر *(Tsymbal and Puuronen, 2002)*، روشی برای ایجاد شورایی از دسته‌بندهای بیز با زیرمجموعه‌های تصادفی از ویژگی پیشنهاد شده است .

ب) انتخاب فروکاست‌های ویژگی: فروکاست کوچک‌ترین زیرمجموعهٔ ویژگی است که توانایی پیش‌بینی یکسانی با مجموعهٔ تمام ویژگی‌ها دارد. در این روش، اندازهٔ شورای دسته‌بندها که با فروکاست‌های مختلف ویژگی ایجاد می‌شوند محدود به تعداد ویژگی‌هاست. پژوهش‌های مختلفی بر اساس این رویکرد انجام شده است. ایدهٔ ترکیب چندین دسته‌بند KNN برای دسته‌بندی متن با استفاده از فروکاست‌ها در *(Bao and Ishii, 2002)* ارائه شده است. روش «حذف بدترین ویژگی» برای یافتن مجموعه‌ای از فروکاست‌ها و ترکیب آنها با روش بیز ساده در *(Wu and Bell, 2003)* ارائه

---

[1] Ho
[2] curse of dimensiality
[3] random-based
[4] reduct
[5] performance-based strategy
[6] Random Subspace Method (RSM)
[7] Bryll
[8] wrapper

شده است. «هو» و همکارانش روشهای مختلفی را برای ساخت جنگل‌های تصمیم با فروکاست‌های متفاوت پیشنهاد کردند (Hu, Yu, and Wang 2005). همچنین ایجاد شورای دسته‌بندها با فروکاست‌های مختلف براساس نظریهٔ مجموعه‌های شولا[1] نیز نتایج بسیار خوبی را نشان می‌دهد (Hu et al. 2007; Wanga et al. 2010).

ج) انتخاب برمبنای کارایی زیرمجموعه‌ها: در این رویکرد، زیرمجموعه‌های ویژگی به‌گونه‌ای انتخاب می‌شوند که کارایی نهایی سیستم شورای دسته‌بندها افزایش یابد. در (Cunningham and Carney 2000) ابتدا دسته‌بندهای پایه با زیرمجموعه‌های تصادفی ویژگی ایجاد می‌شوند؛ سپس فرایند تکراریِ بهبود زیرمجموعه‌ها براساس روش جستجوی تپه‌نوردی[2] انجام می‌شود تا صحت و گوناگونی دسته‌بندهای پایه افزایش یابد؛ به‌نحوی که برای تمام زیرمجموعه‌های ویژگی به‌دنبال کاستن یا افزودن یک ویژگی هستیم. اگر زیرمجموعهٔ ویژگی جدید، کارایی بهتری برروی نمونه‌های ارزیابی داشت، این تغییر انجام می‌شود. درغیراین صورت این فرآیند تا زمانی تکرار می‌شود که بهبود بیشتری به‌دست نیاید. در پژوهشی دیگر، روشی با نام «حذف یک‌در ده ورودی»[3] (ID) ارائه شده که ویژگی‌ها براساس همبستگی آنها با برچسب دسته‌ها انتخاب می‌شوند (Oza and Tumer 2001) و نشان داده شده که این روش کارایی بهتری نسبت به انتخاب تصادفی زیرمجموعه‌های ویژگی دارد. استفاده از روش‌های مختلف انتخاب ویژگی همراه با معیار گوناگونی در تابع برازش برای انتخاب بهترین زیرمجموعه‌های ویژگی در (Tsymbal, Pechenizkiy, and Cunningham 2005) بررسی شده است. این مقاله نشان می‌دهد برخی معیارهای گوناگونی به‌طور کلی نتایج بهتری می‌دهند.

همچنین روش‌های مختلفی برمبنای الگوریتم ژنتیک برای یافتن زیرمجموعه‌های ویژگی ارائه شده است (Kuncheva and Jain 2000; Yu and Cho 2006; Opitz 1999). «کونچوا» دو رویکرد کلّی برای ایجاد شورای انتخاب ویژگی با الگوریتم ژنتیک بیان می‌کند (Kuncheva 2004) که در ادامه آورده شده است. شایان ذکر است، هدف GA انتخاب بهترین زیرمجموعهٔ ویژگی نیست؛ بلکه GA به‌دنبال یافتن دسته‌بندهایی است که درمجموع از بهترین دسته‌بند پایه بهتر باشند. ازاین‌رو، دسته‌بندهای مجموعه، باید صحّت دسته‌بندی قابل قبول و درعین‌حال گوناگون باشند.

رویکرد اوّل (Opitz 1999): (شکل ۶)، مدل‌سازی رویکرد اوّل را برای مسأله‌ای با تعداد ده ویژگی نشان می‌دهد که جمعیت اوّلیه شامل چهار کروموزوم است. در این رویکرد، زیرمجموعه‌های ویژگی به طور مجزا انتخاب می‌شوند. هر کروموزوم نمایندهٔ یک زیرمجموعهٔ ویژگی (یک دسته‌بند پایه) و مجموعهٔ تمام کروموزوم‌های تولید شده، شورای دسته‌بندها را نشان می‌دهد. هر ژن می‌تواند مقادیر ۰ یا ۱ داشته باشد. مقدار ۱ برای ژن iام ($1 \le i \le$ تعداد ویژگی) نشان دهندهٔ انتخاب ویژگی iام در آن زیرمجموعه است.

کروموزوم ۱	۱	۱	۰	۱	۰	۰	۱	۰	۱	۰
کروموزوم ۲	۰	۰	۰	۰	۱	۰	۰	۰	۱	۰
کروموزوم ۳	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱
کروموزوم ۴	۱	۰	۰	۱	۰	۰	۰	۱	۱	۰

(شکل ۶): مدل‌سازی شورای انتخاب ویژگی با الگوریتم ژنتیک

در این روش، گوناگونی دسته‌بندها به‌طور معمول به‌صورت فاصلهٔ زیرمجموعه‌های ویژگی محاسبه می‌شود. برای نمونه، فاصلهٔ کروموزوم اوّل از دیگر کروموزوم‌ها به‌ترتیب برابر ۴، ۷، و ۴ است. درنتیجه، میانگین گوناگونی دسته‌بند اوّل، برابر ۵ است. بااین‌حال، محاسبهٔ گوناگونی دسته‌بندها به این روش، گوناگونی خروجی دسته‌بندها را تضمین نمی‌کند. این معیار گوناگونی همراه با صحّت گوناگونی هر دسته‌بند در تابع برازش شورا استفاده می‌شود.

رویکرد دوّم (Kuncheva and Jain 2000): در این رویکرد، هر کروموزوم در جمعیت، نمایان‌گر کلّ شوراست. همچنین، زیرمجموعه‌های ایجاد شده می‌توانند مجزا یا هم‌پوشان باشند. در شیوهٔ مجزا، طول هرکروموزوم برابر با تعداد ویژگی‌های مسأله است. هر ژن می‌تواند مقادیر ۰ تا L (تعداد دسته‌بندهای شورا) را داشته باشد و نشان‌دهندهٔ دسته‌بندی است که آن ویژگی را دارد. برای مثال، مسأله‌ای با تعداد ۱۰ ویژگی را در نظر بگیرید که شورا شامل ۴ دسته‌بند $\{D_1, D_2, D_3, D_4\}$ است. در این مسأله، کروموزوم به شکل [۳، ۳، ۲، ۱، ۳، ۰، ۳، ۱، ۴، ۱] نشان می‌دهد که D_1 شامل ویژگی‌های ۱، ۳ و ۷؛ D_2 شامل ویژگی ۸، D_3 شامل ویژگی‌های ۴، ۶، ۹ و ۱۰؛ و D_4 شامل ویژگی ۲ است. ویژگی ۵ در هیچ زیرمجموعه‌ای استفاده نشده است. در

[1] rough sets
[2] hill-climbing search
[3] input decimation (ID)

نشده است. در شیوهٔ هم‌پوشان، هر شورای دسته‌بندی، با یک کروموزم دودویی با طول $n \times L$ نمایش داده می‌شود. n بیت اوّل نمایان‌گر زیرمجموعهٔ ویژگی اوّل برای D_1 است. به همین ترتیب n بیت‌های بعد زیرمجموعه‌های ویژگی دیگر را نمایش می‌دهند. در این روش، هر دسته‌بند می‌تواند ویژگی‌های مشترک با سایر دسته‌بندها را داشته باشد.

مشکل رویکرد اوّل این است که صحّت دسته‌بندی شورا در فرآیند ارزیابی روش تأثیری ندارد. لذا، تضمینی وجود ندارد که شورا صحّت دسته‌بندی بیشتری از بهترین دسته‌بند پایه (کروموزوم) داشته باشد. درمقابل، در رویکرد دوّم، صحّت شورای دسته‌بندی به‌طور دقیق محاسبه می‌شود. مزیت رویکرد اوّل، هزینهٔ محاسباتی کمتر است.

۲-۳- سطح دسته‌بند

روش اصلی در ایجاد گوناگونی در سطح دسته‌بند، استفاده از دسته‌بندهای مختلف مانند شبکه‌های عصبی، نزدیک‌ترین k همسایه، درخت‌های تصمیم و غیره است. رویکرد دیگر برای دستیابی به گوناگونی در این سطح، استفاده از پارامترهای آموزش متفاوت برای دسته‌بندهای مختلف است. برای نمونه، دسته‌ای از شبکه‌های عصبی پرسپترون چندلایه[1] می‌تواند با وزن‌های اولیهٔ متفاوت، تعداد لایه‌ها و نرون‌های متفاوت، خطای هدف متفاوت و ... آموزش ببیند. تنظیم این پارامترها کنترل ناپایداری[2] دسته‌بندهای مجزا را ممکن می‌سازد و لذا به گوناگونی آنها منجر می‌شود. توانایی کنترل ناپایداری شبکه‌های عصبی و دسته‌بندهایی از نوع درخت تصمیم، آنها را نامزدهایی مناسب برای استفاده در سیستم دسته‌بند شورایی می‌کند. از سوی دیگر، دسته‌بندهای به‌طور کامل متفاوت، مانند شبکه‌های عصبی MLP، درخت‌های تصمیم[3]، دسته‌بندهای نزدیک‌ترین همسایه، و ماشین‌های بردار پشتیبان[4] می‌توانند برای افزودن گوناگونی ترکیب شوند *(Kuncheva, 2004)*.

۲-۴- کدهای خروجی تصحیح‌کنندهٔ خطا[5]

از نگاه سیستم‌های شورایی، روش ECOC را می‌توان از جمله رویکردهای شورایی مناسب برای حل مسائل چنددسته‌ای دانست. پایهٔ روش ECOC طرّاحی یک رشته‌کد[6] برای هر کدام از دسته‌های مسأله است. این روش که در ابتدا توسط «دیتریچ»[7] و «بکیری»[8] ارائه شده است *(Dietterich and Bakiri, 1995)*، ماتریس M با ابعاد $N_c \times L$ ایجاد می‌کند، که L طول هر رشته‌کد است و هر عنصر این ماتریس مقادیر $\{-1, 1\}$ را دارد. هر ستون این ماتریس نمایان‌گر یک دسته‌بند دودویی است که دودوساز[9] نیز خوانده می‌شود. هر دودوساز، دسته‌های مسأله را به دو فرادسته[10] تقسیم می‌کند. اگر $M_{ij} = 1$ باشد، نمونهٔ x که به دستهٔ iام تعلق دارد، نمونهٔ مثبتی برای دسته‌بند jام محسوب می‌شود و در غیراین صورت نمونهٔ منفی خواهد بود. (شکل ۷)، یک نمونه ماتریس ECOC برای مسأله‌ای ۴ دسته‌ای را نشان می‌دهد که شامل شش دسته‌بند است. در این ماتریس، هر ستون متناظر با یک دسته‌بند و هر سطر متناظر رشته‌کدی منحصربه‌فرد برای هر دسته است. برای نمونه، دسته‌بند h_3، دو فرادسته را ایجاد می‌کند: دسته‌های ۱ و ۴ فرادستهٔ اوّل و دسته‌های ۲ و ۳ فرادستهٔ دوّم را شکل می‌دهند. لذا دسته‌بند h_3، با نمونه‌های دستهٔ ۱و ۴ به‌عنوان نمونه‌های مثبت و نمونه‌های دستهٔ ۲ و ۳ به‌عنوان نمونه‌های منفی آموزش می‌بیند.

dichotomizer

دسته						
۱	1	0	1	0	0	1
۲	1	1	0	1	1	0
۳	0	1	0	1	0	1
۴	1	0	1	1	1	1

(شکل ۷): مثالی از ماتریس ECOC برای مسأله‌ای با ۴ دسته و ۶ دسته‌بند

برای دسته‌بندی نمونهٔ آزمون x، خروجی هر دسته‌بند مقدار صفر یا یک خواهد بود؛ که درنتیجه بردار خروجی با طول L ایجاد خواهد شد. این بردار خروجی با رشته‌کدهای هر دسته مقایسه می‌شود و دسته‌ای که کمترین فاصله را با بردار خروجی دارد، به‌عنوان دستهٔ نمونهٔ آزمون پیش‌بینی می‌شود. فرآیند ادغام خروجی دسته‌بندهای منفرد رمزگشایی[11] خوانده می‌شود. مرسوم‌ترین روش‌های رمزگشایی، تابع فاصلهٔ همینگ و نرم L^2 هستند که به‌دنبال کمترین فاصلهٔ بُردار خروجی با هر رشته‌کد هستند:

$$y_H = arg \min_{r\in\{1,\dots,N_c\}} \sum_{i=1}^{L} \left(\frac{1 - sign(M(r,i), f_i(x))}{2} \right),$$

[1] multilayer perceptron (MLP)
[2] instability
[3] decision trees
[4] support vector machines
[5] Error Correcting Output Codes (ECOC)
[6] codeword

[7] Dietterich
[8] Bakiri
[9] dichotomizer
[10] metaclass
[11] decoding

$$y_{L^1} = arg \min_{r\in\{1,\dots,N_c\}} \sum_{i=1}^{L} |f_i(x) - M(r,i)|$$

که اگر $z > 0$، $sign(z)$ برابر با ۱+؛ اگر $z < 0$، $sign(z)$ برابر با ۱-؛ و درغیراین‌صورت صفر خواهد بود. $M(r,.)$ نشان‌دهندهٔ رشته‌کد r در ماتریس است و $y \in \{1,\dots,N_c\}$ دستهٔ پیش‌بینی نمونهٔ آزمون را نشان می‌دهد. روش‌های رمزگشایی مختلفی علاوه‌بر روش‌های فاصله ارائه شده است که برای مطالعهٔ بیشتر مرجع *(Windeatt and Ghaderi, 2003)* پیشنهاد می‌شود.

روش ECOC توسط «آلوین»[1] و همکارانش توسعه داده شد *(Allwein, Schapire, and Singer, 2001)*. در این شیوه، عناصر ماتریس ECOC می‌توانند شامل مقادیر $\{1, 0, -1\}$ باشند؛ که مقدار صفر نشان‌دهندهٔ عدم استفاده از یک دسته در فرآیند آموزش دسته‌بند است. در *(Allwein, Schapire, and Singer, 2001)*، هر عنصر ماتریس مقدار صفر با احتمال ۰/۵ و مقدار ۱+ یا ۱- با احتمال ۰/۲۵ دارد. از این‌رو، نمونه‌های یک دسته می‌توانند در فرآیند آموزش یک دسته‌بند نادیده گرفته شوند. این روش ایجاد ماتریس ECOC، کدهای تصادفی تنک[2] خوانده شده؛ درحالی‌که روش اولیهٔ ایجاد ماتریس ECOC کدهای تصادفی متراکم[3] گفته می‌شود. با استفاده از این رویکرد یکپارچه، روش‌های قدیمی مسائل چنددسته‌ای (یک- درمقابل- یک و یک- درمقابل-همه) را می‌توان به‌آسانی در قالب ماتریس ECOC بازنمایی کرد. در روش یک- درمقابل- یک، ماتریس ECOC شامل $\binom{N_c}{2}$ ستون است؛ به‌طوری‌که هر ستون متناظر با یک زوج دسته ω_i, ω_j است. در این ستون، مقدار سطر iام برابر ۱+، مقدار سطر jام برابر ۱- و مقدار بقیهٔ عناصر صفر است. در رویکرد یک- درمقابل-همه، M ماتریسی با ابعاد $N_c \times N_c$ است که مقادیر عناصر قطری آن برابر ۱+ و مقادیر سایر عناصر ۱- است.

بحث و نتیجه‌گیری: هرچند روش‌های معرفی‌شده براساس رویکرد زیرنمونه، کارایی خود را برای مجموعه‌داده‌های مختلف نشان داده‌اند، امّا محبوبیت روش‌های Bagging و AdaBoost در حوزه‌های مختلف کاربردی به‌مراتب بیشتر است. همچنین نتایج برخی پژوهش‌های اخیر نشان می‌دهد که آموزش دسته‌بندها با زیرمجموعه‌های ویژگی مختلف یکی از مؤثرترین روش‌های ایجاد گوناگونی در شورای دسته‌بندهاست. به‌بیان دیگر، روش‌های افراز فضای ویژگی، کارایی بهتری نسبت به روش‌های زیرنمونه (مانند bagging و boosting) نشان می‌دهند *(Banfield et al., 2007; Tumer and Ghosh, August 1996)*.

برخی پژوهش‌ها روش‌های مهم شورای دسته‌بندها را مقایسه کرده‌اند. مقایسهٔ روش‌های ایجاد شورا با دسته‌بند درخت تصمیم در *(Banfield et al., 2007)*، از جامع‌ترین پژوهش‌های اخیر در سیستم‌های شورایی است. در این مقاله، روش bagging با روش‌های RSM، دو نسخهٔ از روش AdaBoost (شامل ۵۰ و ۱۰۰۰ درخت)، و سه نسخهٔ از روش جنگل تصادفی (RF) با تعداد یک، دو و لگاریتم نصف به‌علاوهٔ یک ویژگی (RF.lg) مقایسه شده است. نتایج تجربی این پژوهش با ۵۷ مجموعه داده نشان می‌دهد روش AdaBoost.1000 و RF.lg بهترین صحت دسته‌بندی را نتیجه می‌دهند. با این حال، برای ۳۷ مجموعهٔ داده آزمایش، هیچ کدام از این روش‌های موجود تفاوت معناداری را در مقایسه با bagging نشان نمی‌دهند. مقایسهٔ روش‌های طراحی شورا شامل bagging، AdaBoost، Arc-X4 با دسته‌بندهای SVM نتایج متفاوتی را نشان می‌دهد *(Wang et al., 2009)*. بر اساس نتایج تجربی این پژوهش با استفاده از بیست مجموعه داده، bagging-SVM به‌ویژه با تابع هستهٔ چندجمله‌ای[4] صحت دسته‌بندی بهتری را ارائه می‌کند. یکی از نتایج بسیار مهم این پژوهش این است که در بسیاری از مجموعه‌داده‌های آزمایش، روش‌های شورایی با استفاده از SVM، صحت دسته‌بندی بدتری را در مقایسه با دسته‌بند منفرد SVM نتیجه می‌دهند. برای نمونه، میانگین صحت SVM با هستهٔ تابع پایهٔ شعاعی، از تمام روش‌های شورایی بهتر است. این نتایج تا حدّ زیادی با نتایج پژوهش *(Meyer, Leisch, and Hornik, 2003)* مشابه است.

۲-۵- راهکار آزمون-انتخاب[5]

هرچند روش‌های مختلفی برای تشکیل شورای دسته‌بندها وجود دارد، تعیین بهترین روش برای مسألهٔ خاص موضوعی فراخور پژوهش است. برای این موضوع، رویکرد «آزمون و انتخاب» *(Sharkey et al., 2000)* به‌عنوان گزینه‌ای مناسب ارائه شده است؛ هرچند که از «جواب اصلی»[6] دور باشد. مرحلهٔ اوّل این روش، فراتولید[7] ترکیب‌های ممکن

[1] Allwein
[2] sparse random codes
[3] dense random codes
[4] polynominal kernel
[5] test-select strategy
[6] complete solution
[7] over-producing

شوراهاست. سپس این شوراها با مجموعهٔ اعتباریابی[1] محک زده می‌شوند و بهترین شورا انتخاب می‌شود. درنهایت، این شورا با مجموعهٔ آزمون (که در فرایند انتخاب وارد نشده بود)، آزمایش می‌شود. ازاین‌رو رویکرد «آزمون و انتخاب» درعمل به دو مجموعهٔ آزمون نیاز دارد: مجموعهٔ اعتباریابی و مجموعهٔ آزمون؛ که به‌ترتیب برای انتخاب شورا و آزمون کارایی آن استفاده می‌شوند. مجموعهٔ اعتباریابی، می‌تواند مانع از فراتخمین[2] کارایی شورا شود که به‌طور معمول زمانی‌که انتخاب و ارزیابی براساس یک مجموعهٔ آزمون است، رخ می‌دهد.

۳- طراحی قواعد ترکیب

موضوع مهم دیگر در ساخت شورای دسته‌بندها، انتخاب قاعدهٔ ترکیب دسته‌بندهای پایه است. انتخاب قاعدهٔ ترکیب،که تحت عنوان بهینه‌سازی تصمیم نیز نامیده می‌شود، ساخت تابع ترکیب قدرتمندی برای ادغام تصمیم دسته‌بندهای مجزاست. به‌بیان دیگر بهینه‌سازی تصمیم سعی در یافتن قاعدهٔ ترکیب بهینه‌ای دارد که تصمیم مجموعهٔ دسته‌بندها را ادغام کند. نشان داده شده که درصورت عدم استفاده از روش ترکیب مناسب، مزیت گوناگونی دسته‌بندها موجب بهبود کارایی شورا نخواهد شد *(Brodley and Lane 1996)*. در ادامه پس از معرّفی دو مفهوم مهم، مهم‌ترین قواعد ادغام دسته‌بندها آورده شده است. در این مقاله، روش‌هایی که در مقالهٔ *(نیوی کریزی و کبیر ۱۳۸۴)* معرّفی شده است، به‌اجمال بیان شده است.

۳-۱- روش‌های ترکیب آموزش‌پذیر و غیرآموزش‌پذیر

ترکیب غیرآموزش‌پذیر[3] به این معناست که روش ترکیب‌کننده، پارامتر اضافه‌ای برای آموزش ندارد؛ به‌بیان دیگر، پس از اینکه آموزش دسته‌بندهای پایه تمام شد، می‌توان از شورا برای آزمون نمونه‌ها استفاده کرد. نمونه‌ای از این نوع قواعد ترکیب، قاعدهٔ رأی حداکثر[4] و شمارش بورد[5] است. در برخی از روش‌های ترکیب، علاوه‌بر آموزش دسته‌بندهای پایه، پارامترهای روش نیز از طریق الگوریتم یادگیریِ دیگری تعیین می‌شود. به این روش‌ها روش‌های آموزش‌پذیر[6] گفته می‌شود.

۳-۲- انواع خروجی دسته‌بندها

خروجی الگوریتم‌های مختلف دسته‌بندی به یکی از شکل‌های زیر بیان می‌شود (Xu, Krzyzak, and Suen 1992):

الف. خروجی تک‌کلاسه[7] (مطلق): در این شکل، دسته‌بند فقط یک برچسب برای الگوی ورودی (x) ارائه می‌کند. در این نوع، اطّلاعاتی درخصوص قطعیت برچسب پیشنهادی وجود ندارد. به‌طور معمول، تمامی دسته‌بندها می‌توانند یک برچسب را برای الگوریتم ورودی تعیین کنند؛ لذا خروجی مطلق بیشترین کاربرد را دارد.

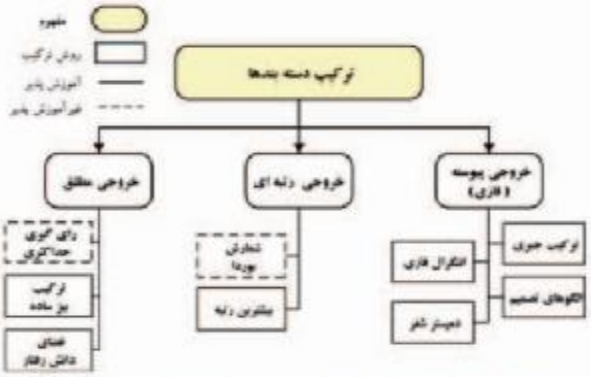
ب. خروجی رتبه‌ای[8]: در این حالت، دسته‌بند تعلّق الگوی ورودی به دسته‌های مختلف را به‌صورت فهرستی مرتب‌شده ارائه می کند. دستهٔ با بیشترین تعلّق در ابتدای فهرست و دستهٔ با کمترین تعلّق در انتهای آن قرار دارد. خروجی رتبه‌ای به‌ویژه برای مسائل با تعداد دسته‌های زیاد مانند تشخیص حرف، صورت و صدا مناسب است.

ج. خروجی امتیازدار[9] (پیوسته) : در این حالت، دسته‌بند برای تعلق الگوی ورودی به هر کدام از دسته‌ها یک مقدار عددی تخصیص می دهد. به‌عبارتی خروجی هر دسته‌بندِ D_i، برداری c- بعدی $[d_{i,1},\dots,d_{i,c}]$ خواهد بود؛ مقدار $d_{i,j}$ نشان‌دهندهٔ امتیازی[10] است که نمونهٔ x متعلّق به دستهٔ j-اُم است.

واضح است که خروجی تک‌کلاسه شامل کمترین اطّلاعات و خروجی امتیازدار حاوی بیشترین اطّلاعات در فرایند دسته‌بندی است. متداول‌ترین تقسیم‌بندی برای روش‌های ترکیب خروجی دسته‌بندها، تقسیم‌بندی آنها بر حسب نوع خروجی الگوریتم دسته‌بندی است که بر اساس آن روش‌های ترکیب به سه بخش تقسیم می شوند. (شکل ۸) روش‌های مختلف ترکیب دسته‌بندها را به‌خوبی نشان می‌دهد. بر اساس این تقسیم‌بندی، روش‌های مهم هر بخش در ادامه آورده شده است.

---

[1] validation
[2] over-estimation
[3] non-trainable
[4] majority vote
[5] Borda count
[6] trainable
[7] abstract level
[8] rank level
[9] measurement level
[10] support



(شکل ۸):روش‌های مختلف ترکیب دسته‌بندها، برگرفته از
(Ruta and Gabrys, 2000)

۳-۳- روش‌های خروجی مطلق (خروجی تک‌دسته‌ای)

۳-۳-۱- رأی اکثریت

رأی اکثریت ساده‌ترین روش ترکیب خروجی دسته‌بندها است. در این روش، خروجی نمونهٔ آزمون دسته‌ای است که بیشترین رأی را در میان دسته‌بندها داراست. زمانی که مسأله تنها شامل دو دسته است، به این روش رأی‌گیری حداکثری ساده[1] نیز گفته می‌شود. در این حالت باید حداقل ۵۰ + ۱٪ دسته‌بندها بر یک دستهٔ خاص توافق داشته باشند. توافق کامل[2] از دیگر روش‌های مرسوم در این حوزه است. در این روش، تنها زمانی که تمامی دسته‌بندها دستهٔ مشترکی برای نمونهٔ آزمون تعیین کنند، تصمیم گرفته می‌شود. این سه رویکرد در (شکل ۹) نشان داده شده است (*Kuncheva, 2004; Lam and Suen., 1997*).

(شکل ۹): الگوهای اتفاق آرا در میان ۱۰ تصمیم‌گیر
(در تمام این سه روش تصمیم نهایی گروه، مشکی است)

برای درک علّت محبوبیت روش رأی‌گیری حداکثری فرض کنید:

- تعداد دسته‌بندها، L، فرد است.
- احتمال اینکه هر دسته‌بند دستهٔ صحیح را برای هر $x \in R^n$ انتخاب کند برابر p است.
- خروجی دسته‌بندها مستقل هستند؛ یعنی برای هر زیرمجموعهٔ دسته‌بندها $A = \{D_{i,1}, \ldots, D_{i,k}\}$، $A \in D$، احتمال توأم آنها قابل تجزیه به‌صورت زیر است:

$$P(D_{i_1} = s_{i_1}, \ldots, D_{i_k} = s_{i_k}) = P(D_{i_1} = s_{i_1}) \times \ldots \times P(D_{i_k} = s_{i_k}) \quad (۵)$$

که $S_{i,j}$ برچسب خروجی دسته‌بند $D_{i,j}$ است.

اگر حداقل $\lfloor L/2 \rfloor + 1$[3] دسته‌بند تصمیم صحیح بگیرند، رأی اکثریت دستهٔ صحیح را انتخاب می‌کند. لذا احتمال دسته‌بندی صحیح شورا برابر است با:

$$P_{maj} = \sum_{m=\lfloor L/2 \rfloor + 1}^{L} \binom{L}{m} p^m (1-p)^{L-m} \quad (۶)$$

این احتمال به‌ازای $p = 0.6,\ 0.7,\ 0.8,\ 0.9$ و $L=3,5,7,9$ در جدول ۲ نشان داده شده است. این نتایج «نظریهٔ هیئت منصفهٔ کندرست»[4] نیز خوانده می‌شود:

(جدول ۱): مقادیر صحت دسته‌بندی به‌ازای مقادیر مختلف L دسته‌بند مستقل و صحت دسته‌بندی p

	$L=3$	$L=5$	$L=7$	$L=9$
$p=0.6$	0.6480	0.6826	0.7102	0.7334
$p=0.7$	0.7840	0.8369	0.8740	0.9012
$p=0.8$	0.8960	0.9421	0.9667	0.9804
$p=0.9$	0.9720	0.9914	0.9973	0.9991

از رابطهٔ فوق چند نکته نتیجه می‌شود:

۱. اگر $p > 0.5$، آنگاه P_{maj} در رابطهٔ ۶ به‌طور یکنواخت افزایش می‌یابد: $L \to \infty : P_{maj} \to 1$

۲. اگر $p < 0.5$، آنگاه P_{maj} در رابطهٔ ۶ به‌طور یکنواخت کاهش می‌یابد: $L \to \infty : P_{maj} \to 0$

۳. اگر $p = 0.5$، آنگاه به‌ازای هر تعداد دسته‌بند: $P_{maj} = 0.5$

این نتایج نشان می‌دهد، تنها زمانی که صحت دسته‌بندی p بیش از ۰/۵ است، می‌توان انتظار بهبود دسته‌بندی شورا را داشت. در این حالت، با شرط مستقل بودن خطای دسته‌بندها صحت شورای دسته‌بندها می‌تواند به ۱ میل کند.

۳-۳-۲- رأی‌گیری حداکثری وزن‌دار

اگر دسته‌بندها در شورا کارایی یکسانی نداشته باشند، منطقی است که برای تصمیم نهایی شورا به دسته‌بندهای برتر وزن بیشتری داده شود. مشابه پیش، فرض کنیم تصمیم دسته‌بند D_i برای دستهٔ ω_j، $d_{i,j}$ باشد؛ به‌طوری که اگر D_i

[1] simple majority
[2] unanimity

[3] $\lfloor . \rfloor$ جزء صحیح عدد را نشان می‌دهد

[4] Condorcet Jury Theorem (1785)

دستهٔ ω_j را انتخاب کند d_{ij} مقدار ۱ و درغیراین‌صورت ۰ است. همچنین فرض می‌کنیم روشی برای اندازه‌گیری کارایی هر دسته‌بند وجود دارد و براساس این کارایی به هر دسته‌بند وزن w_i می‌دهیم. لذا تصمیم شورا برای انتخاب دستهٔ jام بر اساس رابطهٔ زیر است:

$$\sum_{i=1}^{L} w_i d_{i,j} = \max_{j=1:C} \sum_{i=1}^{L} w_i j \qquad (۷)$$

یعنی، دسته‌ای که بیشترین امتیاز وزن‌دار را بگیرد، تصمیم شورا خواهد بود. برای تفسیر راحت‌تر، می‌توان وزن‌ها را هنجار کرد؛ به‌طوری که مجموع آنها یک شود (*Polikar, 2006*).

پرسش اصلی این است که «چگونه وزن‌ها تعیین شوند؟». به‌طور مشخّص اگر می‌دانستیم کدام دسته‌بندها بهتر عمل می‌کنند، بیشترین وزن‌ها را به آنها می دادیم یا شاید تنها آن دسته‌بندها را انتخاب می‌کردیم. زمانی که این دانش را نداریم، راهکار معقول استفاده از کارایی دسته‌بند بر روی مجموعه داده‌های اعتباریابی یا حتّی بر روی داده‌های آموزش، به‌عنوان تخمینی از کارایی آتی دسته‌بند، است. این راهکار در الگوریتم AdaBoost (روابط (۲) و (۴)) استفاده شده است: AdaBoost وزنی برابر $ln\left(\frac{1}{\beta_k}\right)$ به هر دسته‌بند تخصیص می‌دهد که $\beta_k = \frac{\varepsilon_k}{1-\varepsilon_k}$، و ε_k خطای وزن‌دار دسته‌بند D_k است. همانگونه که بیان کردیم، هنجارسازی خطا این امکان را می‌دهد که از معیار خطای β استفاده کنیم که در بازهٔ [۱،۰] قرار دارد (به‌جای ε که بین ۰ و ½ است). امّا دلیل مهم دیگر برای هنجارسازی این است که می‌توان نشان داد که اگر L دسته‌بند مستقل وجود داشته باشند، صحّت شورای دسته‌بندها زمانی حداکثر می‌شود که وزن دسته‌بندها تناسب زیر را داشته باشند (*Kuncheva, 2004*):

$$w_k \propto \log\frac{p_k}{1-p_k} \qquad (۸)$$

ازآنجا که صحّت هر دسته‌بند $p_k = 1 - \varepsilon_k$ است، مشاهده می‌شود که در الگوریتم AdaBoost وزن هر دسته‌بند به‌طور دقیق براساس رابطهٔ ۸ تخصیص داده شده است.

۳-۳-۳- ترکیب بیز ساده[1]

این روش فرض می‌کند که دسته‌بندها مستقل از یکدیگرند و لذا بیز ساده نیز نامیده می‌شود. در این روش، برای هر دسته‌بند D_i، ماتریس ابهام[2] (CM^i) با ابعاد $c \times c$ با استفاده از نمونه‌های آموزش تشکیل می‌شود. عنصر(k,s)ام این ماتریس ($CM_{k,s}^i$)، بیانگر تعداد نمونه‌های در مجموعهٔ آموزش است که برچسب واقعی آنها k است و توسط دسته‌بند D_i به دستهٔ s تخصیص داده شده‌اند. با جمع کردن مقادیر عناصر ستون s ام ماتریس ابهام این دسته‌بند، می‌توان $CM_{.s}^i$ (یعنی تعداد کلّ عناصری که این دسته‌بند به دستهٔ ω_s نسبت داده است) را مشخص کرد:

$$CM_{.s}^i = \sum_{c=1}^{C} CM_{k,s}^i \qquad (۹)$$

با استفاده از مقادیر $CM_{.s}^i$ و $CM_{k,s}^i$، می توان ماتریس برچسب[3]، LM^i، را محاسبه کرد. این ماتریس، ماتریسی با ابعاد $c \times c$ است که عنصر(k,s)ام آن، تخمینی از احتمال آن است که دستهٔ واقعی k باشد، به‌شرطی که دسته‌بند D_i دستهٔ قطعی s را تخصیص داده باشد (رابطهٔ ۱۰):

$$LM_{.s}^i = \hat{P}(\omega_k|D_i(x) = \omega_s) = \frac{CM_{k,s}^i}{CM_{.s}^i} \qquad (۱۰)$$

اگر $S_1, S_2, \ldots, S_L$ برچسب‌هایی باشد که دسته‌بندهای D_1 تا D_L به الگوی $\mathbf{x}$ تخصیص داده‌اند، با فرض استقلال دسته‌بندها، تخمین احتمال اینکه برچسب واقعی الگوی $\mathbf{x}$، ω_s باشد از رابطهٔ ۵ به‌دست می‌آید:

$$\mu_D{}^c(x) = \prod_{i=1}^{L} \hat{P}(\omega_k|D_i(x) = S_i) = \prod_{i=1}^{L} LM_{cs_i}{}^i \quad t = 1, \ldots, c \qquad (۱۱)$$

که درآن $\hat{P}(\omega_k|D_i(x) = S_i)$ با توجّه به نمونه‌های آموزشی تخمین زده می‌شوند. برای این کار، تعداد الگوهایی که توسط D_i برچسب S_i خورده‌اند ولی برچسب واقعی آنها ω_c است برتعداد کل الگوهای نسبت داده شده توسط D_i به دستهٔ ω_i تقسیم می‌شوند. توضیحات بیشتر این روش در (*سیدحسن نبوی کریزی احسان اله کبیری پائیز ۱۳۸۴؛ Kuncheva, 2004*) آورده شده است.

۳-۳-۴- فضای دانش‌رفتار[5]

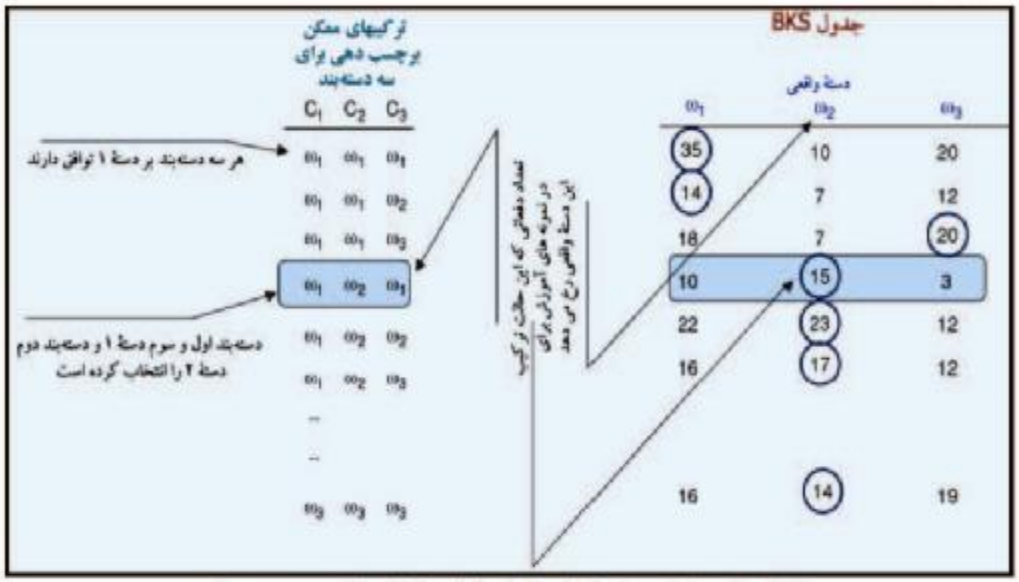
بیشتر روش‌های ترکیب از فرض استقلال دسته‌بندها برای تصمیم گیری نهایی استفاده می‌کنند. این فرض همواره صحیح نیست. روش فضای دانش‌رفتار، BKS به چنین فرضی برای تصمیم گیری نهایی نیاز ندارد. این روش که توسط «هانگ»[6] و «سوئن»[7] (*Huang and Suen, 1993*) ارائه شده

[1] naïve Bayes

[2] confusion matrix
[4] label matrix
[5] Behavior Knowledge Space
[6] Huang
[7] Suen

است، از جدول جستجو[1] برای تعیین برچسب نمونهٔ آزمون استفاده می‌کند (Huang and Suen, 1995).

با ثبت نظر دسته‌بندها در مورد الگوهایی که دستهٔ آنها معلوم است، روش BKS رفتار جمعی دسته‌بندها را مدل کرده و بر اساس این مدل برچسب دستهٔ الگوی ناشناخته x را مشخص می‌کند. به‌عبارت دیگر، در این روش تصمیم گیری نهایی برای دستهٔ یک الگو با استفاده از رفتاری که دسته‌بندهای مختلف در هنگام یادگیری الگوها از خود نشان داده‌اند صورت می گیرد.



(شکل ۱۰): نمایش روش فضای دانش‌رفتار؛ برگرفته از (Polikar 2006)

در (شکل ۱۰)، این روش با استفاده از یک مثال نشان داده شده است. فرض کنید سه دسته‌بند D_1, D_2, D_3 داریم و مسأله شامل سه دسته است. ۲۷ ترکیب ممکن برای برچسب‌دهی یک الگو وجود دارد که از $\{\omega_1\ \omega_1\ \omega_1\}$ تا $\{\omega_3\ \omega_3\ \omega_3\}$ مرتّب شده‌اند. درطول فرایند آموزش، سابقهٔ رخ دادن هر ترکیب و نیز دستهٔ صحیح متناظر با آن ترکیب را ثبت می‌کنیم. تعداد پیشامدهای هر ترکیب و هر دسته آورده شده است و حداکثر آن با دایره مشخص شده است. برای مثال، فرض کنید ترکیب $\{\omega_1\ \omega_2\ \omega_1\}$ در طول فرایند آموزش در مجموع ۲۸ بار رخ داده است؛ ۱۰ بار از آن دستهٔ صحیح ω_1 ، در ۱۵ بار از آن دستهٔ صحیح ω_2 و در ۳ بار از آن دستهٔ صحیح ω_3 بوده است. لذا برندهٔ این حالت ترکیب، ω_2 است: دسته‌ای که بیشترین مشاهدهٔ صحیح را برای این حالت ترکیب برچسب‌ها داشته است. بدین‌ترتیب، در فرایند آزمون هر زمان که ترکیب $\{\omega_1\ \omega_2\ \omega_1\}$ رخ دهد، این روش دستهٔ ω_2 را انتخاب می‌کند. قابل توجّه است که برای این حالت، اگر ترکیب دسته‌بندها با روش رأی‌گیری حداکثری باشد ω_1 انتخاب خواهد شد. اگر آموزش دسته‌بندها با تعداد زیاد نمونه‌های آموزش انجام شود و جدول به روش مناسبی هنجار شود، رأی حداکثر با استفاده از جدول BKS می‌تواند احتمال‌های پسین دسته را دقّت خوبی تخمین بزند.

اگر L دسته‌بند و C کلاس داشته باشیم، با احتساب دستهٔ رد[2] برای الگوها، جدول BKS $(C+1)^L$ مقدار خواهد داشت. برای محاسبهٔ جدول BKS به مجموعه نمونه‌های زیادی نیاز است. به‌عنوان نمونه اگر برای هر حالت ترکیبِ برچسب‌ها به ده نمونه نیاز باشد، برای حالت پنج دسته‌بند و سه دسته الگو، به تعداد ۱۰۲۴ نمونه آموزشی $(10 \times (C+1)^L = 10 \times (3+1)^5 = 1024)$ نیاز خواهد بود. هر چه تعداد داده بیشتر باشد، دقّت روش بیشتر خواهد بود. واضح است که ترکیب با این روش زمانی موفّق خواهد بود که دسته‌بندها، الگوها را تحت تمامی شرایط ممکن یاد گرفته باشند.

۳-۴- روش‌های ترکیب برای خروجی رتبه‌ای

۳-۴-۱- شمارش بوردا

شمارش بوردا از یک نگاه مهم، با قواعدی که تاکنون معرّفی شده متفاوت است: این روش، دسته‌های بازنده[3] را نادیده

[1] look up table

[2] reject

[3] دسته‌ای که دسته‌بند از دسته‌بندی نمونه‌های آن ناتوان است

نمی‌گیرد. در این روش برای هر دستهٔ c، «شمارش بوردا» برابر با مجموع تعداد دسته‌هایی که رتبهٔ آنها پایین‌تر از دستهٔ c است. اگر $B_{k,c}(x)$ تعداد دسته‌هایی باشد که برای نمونهٔ آزمون x توسط دسته‌بند kام پایین‌تر از دستهٔ c مرتب شده‌اند، تصمیم نهایی شورا دسته‌ای است که بیشترین شمارش بوردا را دارد (Wanas and Kamel 2001):

$$\mu(x) = \underset{c}{\operatorname{argmax}} \sum_{k=1}^{L} B_{k,c}(x) \qquad (۱۲)$$

پیاده‌سازی این روش ساده است و نیازمند آموزش نیست. بااین‌حال، این روش بین کارایی دسته‌بندها تفاوت قائل نمی‌شود و همهٔ دسته‌بند یکسان درنظر گرفته می‌شوند، لذا زمانی که می‌دانیم برخی دسته‌بندها به‌احتمال برتر هستند این روش ترجیح داده نمی‌شود. برای مسألهٔ با دو دسته، روش شمارش بوردا معادل روش رأی حداکثر است. روش شمارش بوردا به‌طور معمول زمانی که خروجی دسته‌بند به‌شکل رتبه‌ای است استفاده می‌شود. زمانی که خروجی دسته‌بند از نوع پیوسته است، می‌توان مقادیر خروجی را رتبه‌بندی کرده و از این روش برای ترکیب خروجی دسته‌بندها استفاده کرد.

۳-۵- ترکیب خروجی‌های امتیازدار

فرض کنید $\mathcal{D}$ و Ω به‌ترتیب بیان‌گر مجموعهٔ دسته‌بندها و مجموعهٔ دسته‌ها باشند:

$$\mathcal{D} = \{D_1, \dots, D_L\}, \qquad \Omega = \{\omega_1, \omega_2, \dots, \omega_c\} \qquad (۱۳)$$

هر دسته‌بند D_i به‌ازای بردار ویژگی $x \in \Re^n$ یک بُردار خروجی به‌صورت:

$$D_i(x) = [d_{i,1}(x) \quad d_{i,2}(x) \quad \dots \quad d_{i,c}(x)] \qquad (۱۴)$$

تولید می‌کند که در آن $d_{i,j}(x)$ امتیاز تعلّق الگوی x به دستهٔ ω_j است که توسط دسته‌بند D_i گزارش شده است. ادغام خروجی دسته‌بندها به‌این معناست که با ترکیب بردارهای خروجی L دسته‌بند، بردار تعلّق الگوی x به دسته‌های c گانه را به‌صورت $D(x) = F(D_1(x), \dots, D_L(x))^T$ تعیین کنیم که در آن F قاعدهٔ ترکیب است. اگر $D(x)$ به‌صورت زیر فرض شود:

$$D(x) = [\mu_D^1(x) \quad \mu_D^2(x) \quad \dots \quad \mu_D^c(x)] \qquad (۱۵)$$

پس از تعیین این بردار، دستهٔ الگوی x متناظر با دُرایه‌ای از بُردار $D(x)$ خواهد بود که بیشترین مقدار را دارد. خروجی دسته‌بندهای مختلف می‌توانند به‌صورت زیر در ساختار ماتریسی به‌نام ماتریس نیم‌رخ (پروفایل) تصمیم[1]، $DP(x)$، مرتب شود (Kuncheva, Bezdek, and Duin, 2001):

$$DP(x) = \begin{bmatrix} d_{1,1}(x) & \dots & d_{1,j}(x) & \dots & d_{1,c}(x) \\ \vdots & & \vdots & & \vdots \\ d_{i,1}(x) & \dots & d_{i,j}(x) & \dots & d_{i,c}(x) \\ \vdots & & \vdots & & \vdots \\ d_{L,1}(x) & \dots & d_{L,j}(x) & \dots & d_{L,c}(x) \end{bmatrix} \qquad (۱۶)$$

سطر iام ماتریس، خروجی دسته‌بند D_i برای الگوی x و ستون jام ماتریس، بردار تعلّق الگوی x به دستهٔ ω_j است که به کمک دسته‌بندهای D_1 تا D_L ارائه شده است. مقدار $d_{i,j}(x)$ با توجّه به مقدار شباهت الگو به دسته‌های مختلف مقداری بین صفر و یک است. شایان ذکر است ماتریس پروفایل تصمیم برای دسته‌بندهای با خروجی مطلق و خروجی رتبه‌ای نیز می‌تواند به‌کارگرفته شود. در دسته‌بندهای با خروجی مطلق (تک‌دسته‌ای)، مقدار $d_{i,j}(x)$ صفر یا یک و برای دسته‌بندهای با خروجی رتبه‌ای، $d_{i,j}(x)$ یک عدد صحیح است که رتبهٔ تعلّق الگوی x به دستهٔ ω_j را نشان می‌دهد.

۳-۵-۱- روش‌های ترکیب جبری

به‌طور کلّی روش‌های ترکیب جبری ساده، ترکیب‌کننده‌هایی «غیرآموزش‌پذیر» برای خروجی‌های پیوسته هستند. امتیاز کلّی هر دسته با استفاده از تابعی ساده از خروجی دسته‌بندها به‌دست می‌آید. قواعد ترکیب ساده امتیاز هر دستهٔ ω_j را تنها با استفاده از ستون jام ماتریس $DP(x)$ محاسبه می‌کنند:

$$\mu_j(x) = \mathcal{F}\left[d_{1,j}(x), \dots, d_{L,j}(x)\right] \qquad (۱۷)$$

که $\mathcal{F}$ تابع ترکیب است. دستهٔ الگوی x، اندیس بیشترین $\mu_j(x)$ است. تابع ترکیب $\mathcal{F}$ می‌تواند به روش‌های مختلفی تعیین شود. مرسوم‌ترین این روش‌ها عبارتند از (Kuncheva, Bezdek, and Duin 2001):

□ میانگین ساده ($\mathcal{F}$ =Average)

$$\mu_j(x) = \frac{1}{L}\sum_{i=1}^{L} d_{i,j}(x) \qquad (۱۸)$$

قاعدهٔ میانگین معادل با قاعدهٔ مجموع (همراه با ضریب هنجارسازی $\frac{1}{L}$) است. در هر دو روش، تصمیم شورا دسته‌ای خواهد بود که امتیاز $\mu_j(x)$ آن بیشترین مقدار است.

□ کمینه/ بیشینه / میانه: این روش‌ها، مقدار کمینه، بیشینه یا میانه را ازمیان خروجی دسته‌بندها انتخاب می‌کند:

[1] decision profile (DP)

$$\mu_j(x) = \max_{i=1..L}\{d_{i,j(x)}\} \quad (۱۹)$$

$$\mu_j(x) = \min_{i=1..L}\{d_{i,j(x)}\} \quad (۲۰)$$

$$\mu_j(x) = \underset{i=1..L}{median}\{d_{i,j(x)}\} \quad (۲۱)$$

در همهٔ این شیوه‌ها تصمیم نهایی، دسته‌ای خواهد بود که بیشترین امتیاز $\mu_j(x)$ را داشته باشد. روش کمینه، محافظه‌کارانه‌ترین روش ترکیب است؛ چرا که دسته‌ای را انتخاب می‌کند که کمترین امتیازی که تمام دسته‌بندها به آن دسته داده‌اند، بیشترین مقدار باشد.

□ **میانگین اصلاح‌شده**[1]: زمانی که برخی دسته‌بندها امتیاز نامرسوم کم یا زیادی را به دسته‌ها بدهند، روش ترکیب میانگین ممکن است نتایج خوبی را ایجاد نکند. برای حلّ این مشکل، بدبین‌ترین و خوش‌بین‌ترین دسته‌بندها کنار گذاشته می‌شوند. ازاین‌رو، در این روش P٪ کمترین امتیاز و بیشترین امتیاز کنار گذاشته می‌شود و سپس میانگین امتیازهای باقی‌مانده محاسبه می‌شود. شایان توجه است که روش میانگین اصلاح‌شده با کران ۵۰٪ معادل با روش میانه است.

□ **ضرب**: در این روش امتیاز دسته‌بندها درهم ضرب می‌شوند. این روش، به بدبینانه‌ترین دسته‌بند بسیار حساس است؛ یعنی امتیاز کم (نزدیک به صفر) برای یک دسته از سوی هرکدام از دسته‌بندها درعمل می‌تواند احتمال انتخاب آن دسته را از بین ببرد. با این‌حال، زمانی که خروجی دسته‌بندها تخمین خوبی از احتمال‌های پسین هر دسته فراهم کنند، این قاعده بهترین تخمین احتمال پسین کلی دستهٔ انتخاب‌شده را ارائه می‌کند.

□ **میانگین تعمیم‌یافته**[2]: بسیاری از قواعد بالا درحقیقت نوع خاصّی از روش میانگین عمومی زیر هستند:

$$\mu_j(x,\alpha) = \left(\frac{1}{L}\sum_{i=1}^{L} d_{i,j(x)}{}^{\alpha}\right)^{1/\alpha} \quad (۲۲)$$

که به‌ازای مقادیر خاص α قواعد فوق را می‌سازند. برای مثال، اگر

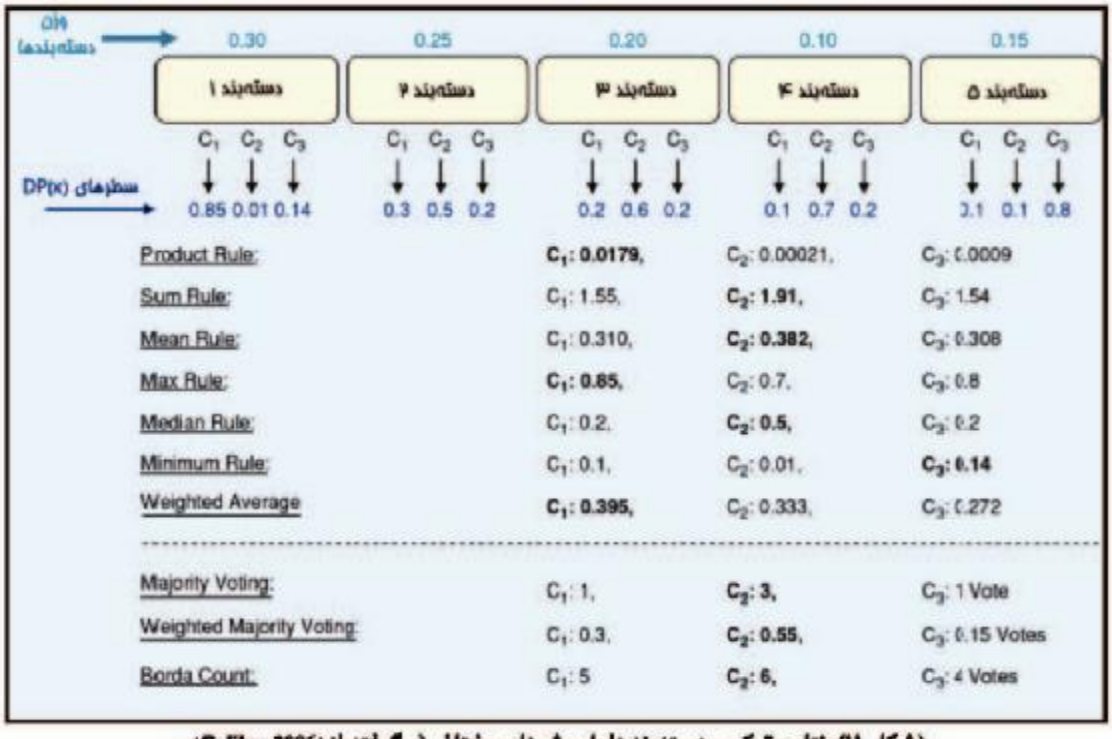
$\alpha \to -\infty \Rightarrow \mu_j(x,\alpha) = \min_i\{d_{i,j(x)}\}$ (۲۳) قاعدهٔ کمینه

$\alpha = 1 \Rightarrow \mu_j(x,\alpha) = \frac{1}{L}\sum_{i=1}^{L} d_{i,j(x)}$ (۲۴) قاعدهٔ میانگین

$\alpha = 0 \Rightarrow \mu_j(x,\alpha) = \left(\prod_{i=1}^{L} d_{i,j(x)}\right)^{1/L}$ (۲۵) قاعدهٔ میانگین هندسی

مثال. مسأله‌ای با ۵ دسته‌بند و ۳ دسته درنظر بگیرید. فرض کنید ماتریس پروفایل زیر برای الگوی x به‌دست آمده است:

$$DP(x) = \begin{bmatrix} 0.85 & 0.01 & 0.14 \\ 0.30 & 0.50 & 0.20 \\ 0.20 & 0.60 & 0.20 \\ 0.10 & 0.70 & 0.20 \\ 0.10 & 0.10 & 0.80 \end{bmatrix}$$



(شکل ۱۱): نتایج ترکیب دسته‌بندها با روش‌های مختلف (برگرفته از *(Polikar 2006)*)

به‌کارگیری روش‌های فوق، نتایج مختلفی را برای عضویت نمونهٔ x به دسته‌های ω_1، ω_2 و ω_3 حاصل می‌کند. این نتایج در (شکل ۱۱) نشان داده شده است؛ و دستهٔ

[1] trimmed mean

[2] generalized mean

منتخب در هر روش پررنگ نشان داده شده است.

از (شکل ۱۱) چند نکته نتیجه می‌شود:

۱) با روش‌های مختلف، دسته‌های مختلف می‌توانند برای الگوی x انتخاب شوند. در این مثال، دستهٔ ۲ در اغلب روش‌ها انتخاب شده است؛ دستهٔ ۱ در سه روش و دستهٔ ۳ تنها در یک روش (روش کمینه) انتخاب شده است.

۲) قاعدهٔ حاصل‌ضرب و قاعدهٔ کمینه امتیاز دستهٔ ۲ را به‌شدت می‌کاهند؛ چرا که این دسته امتیاز کمی از دسته‌بند C_1 گرفته است.

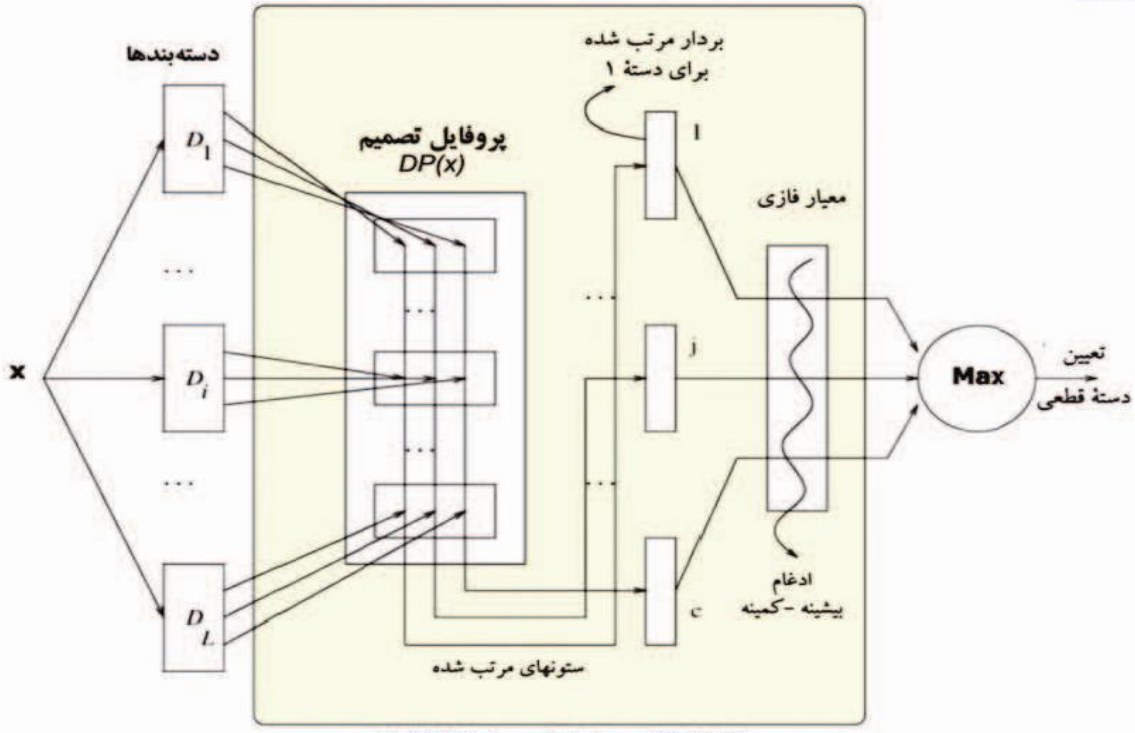
۳) همان‌گونه که انتظار داشتیم، قاعدهٔ میانگین و مجموع نتایج یکسانی را ارائه می‌کنند.

۴) قاعدهٔ میانگین وزن‌دار دستهٔ ۱ را انتخاب کرده است، چرا که دسته‌بند ۱ که وزن زیادی دارد و امتیاز بالایی به دستهٔ ۱ داده است.

همچنین سه روش ترکیب برای خروجی‌های مطلق در این شکل آورده شده است و هر سه دستهٔ ۲ را انتخاب کرده‌اند (در روش شمارش بورد، زمانی‌که وزن دسته‌بند ۱ با دیگر دسته‌بندها برابر است، رتبه‌بندی به نفع دستهٔ ۱ لحاظ شده است).

۳-۵-۲- انتگرال فازی[1]

در روش انتگرال فازی (*FI*) برای نمونهٔ x، c بردار با طول L محاسبه می‌شود که هر بردار متناظر با یک دسته است، و شامل L مقدار از یک معیار فازی است. سپس، اُمین ستون ماتریس پروفایل تصمیم، شامل مقادیر امتیاز L دسته‌بند به دستهٔ ω_j، مرتب می‌شود و با معیار فازی ادغام می‌شود تا میزان تعلق نمونهٔ x به دستهٔ ω_j، $\mu_j(x)$، تعیین شود. لذا، ترکیب فازی نوعی جستجوی بیشترین نمرهٔ توافق بین شواهد عینی[2] (مقادیر خروجی مرتب‌شدهٔ دسته‌بندها برای دستهٔ j) و مقادیر موردانتظار (L مقدار معیار فازی) است. توضیحات بیشتر این روش در (*سیدحسن نبوی کریزی احسان اله کبیری پائیز ۱۳۸۴؛ Kuncheva, 2004*) آورده شده است. نمودار بلوکی روش انتگرال فازی در (شکل ۱۲) نشان داده شده است.



(شکل ۱۲): نمودار بلوکی روش انتگرال فازی

۳-۵-۳- الگوهای تصمیم[3]

ایدهٔ روش ترکیب الگوهای تصمیم (*DT*)، به‌یاد داشتن مرسوم‌ترین پروفایل تصمیم برای هر دسته است که الگوی تصمیم دسته (DT_j) گفته می‌شود؛ و سپس مقایسهٔ آن با پروفایل تصمیم DP با استفاده از یک معیار شباهت است (*Kuncheva, Bezdek, and Duin 2001*). نزدیک‌ترین شباهت، برچسب نمونهٔ ناشناختهٔ x را تعیین می‌کند. برای

[1] fuzzy integral (FI)

[2] objective evidence
[3] decision templates (DT)

تعیین شباهت از معیارهای متفاوتی استفاده شده است. دو معیار شباهت مهم عبارتند از:

۱. فاصلهٔ مربع اقلیدسی: امتیاز شورا برای دستهٔ ω_j برابر است با:

$$\mu_j(x) = 1 - \frac{1}{L \times C}\sum_{i=1}^{L}\sum_{k=1}^{C}\left[DT_j(i,k) - d_{i,k}(x)\right]^2 \quad (۲۶)$$

که $DT_j(i,k)$ ، (i,k)اُمین دَرایهٔ الگوی تصمیم DT_j است. دستهٔ با بیشترین امتیاز $(\mu_{j(x)})$ تصمیم شورا خواهد بود. شایان ذکر است معیارهای دیگر فاصله مانند فاصلهٔ ماهالونبیس[1] نیز می‌تواند به‌کار گرفته شود.

۲. تفاوت متقارن[2]: در این معیار که از نظریهٔ فازی منشأ گرفته است *(Kuncheva 2001; Kuncheva 2003)* امتیاز دستهٔ ω_j برابر است با:

$$\mu_j(x) = 1 - \frac{1}{L \times C}\sum_{i=1}^{L}\sum_{k=1}^{C} max\{min\{DT_j(i,k), 1 - d_{i,k}(x)\}, min\{1 - DT_j(i,k), d_{i,k}(x)\}\} \quad (۲۷)$$

(شکل ۱۲) الگوریتم روش الگوهای تصمیم را نشان می‌دهد و (شکل ۱۴) دیاگرام بلوکی روش را نشان می‌دهد. توضیحات بیشتر این روش در *(نبوی کریزی و کبیر ۱۳۸۴؛ Kuncheva 2004)* آورده شده است.

۱. آموزش

به‌ازای $j = 1, \ldots, c$ میانگین ماتریسهای پروفایل متعلّق به دستهٔ ω_j را برای تمام نمونه‌های آموزش (Z) محاسبه کن. ماتریس میانگین هر دسته را الگوی تصمیم DT_j می‌نامیم:

$$DT_j = \frac{1}{N_j}\sum_{z_k \in \omega_j} DP(z_k) \quad (۲۸)$$

که N_j تعداد عناصر (الگوهای) دستهٔ ω_j است.

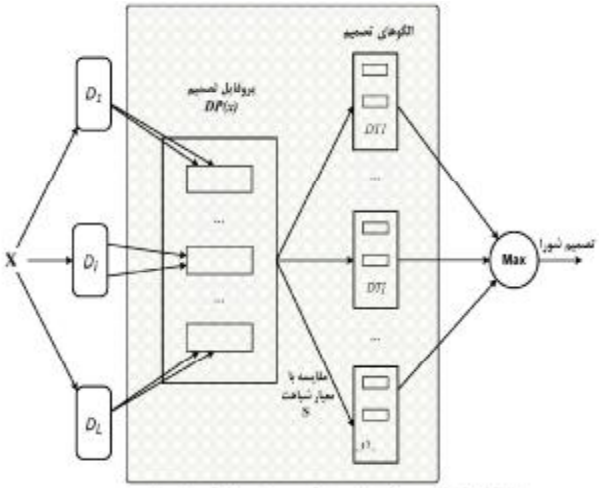
۲. آزمون

برای دادهٔ آزمون $x \in \Re^n$ ، ماتریس $DP(x)$ را ایجاد کند. شباهت بین $DP(x)$ و تمام DT_j ها را محاسبه کن؛

$$\mu_j(x) = S\left(DP(x), DT_j\right) \quad j = 1, \ldots, c \quad (۲۹)$$

(شکل ۱۳): الگوریتم روش ترکیب الگوهای تصمیم

[1] mahalanobis distance
[2] symmetric difference



(شکل ۱۴): دیاگرام بلوکی روش الگوهای تصمیم

۳-۵-۴- ترکیب دمپستر- شَفر[3]

این روش از مفهوم هم‌جوشی داده[4] سرچشمه گرفته است. بسیاری از شیوه‌های همجوشی داده براساس نظریهٔ شواهد دمپستر- شَفر *(Dempster 1967; Shafer 1976)* هستند که از توابع باور[5] برای کمّی‌کردن شواهد موجود در منابع اطّلاعاتی مختلف استفاده می‌کند و سپس با قاعدهٔ ترکیب دمپستر- شَفر ادغام می‌شوند. توضیحات کامل این روش در *(نبوی کریزی و کبیر ۱۳۸۴؛ Kuncheva 2004)* آورده شده است.

۳-۵-۵- الگوریتم‌های تکاملی

از الگوریتم‌های تکاملی برای ترکیب دسته‌بندها در برخی پژوهش‌ها استفاده شده است. در پژوهش *(Lin, Wang, and Yeung 2005)* با هدف بهبود شناسایی دست‌خط چینی دو روش ترکیب بر مبنای الگوریتم ژنتیک ارائه شده است. «نبوی کریزی» و همکارانش روش جدیدی بر پایهٔ الگوریتم بهینه‌سازی تجمّع ذرات[6] برای ترکیب چند دسته‌بند شبکهٔ عصبی ارائه کردند *(Nabavi-Kerizi, Abadi, and Kabir 2010)*. نتایج تجربی پژوهش بر روی چند مجموعه داده، توانایی این روش ترکیب را نشان می‌داد. در پژوهشی دیگر *(نبوی کریزی و کبیر ۱۳۸۷)*، یک روش دومرحله‌ای برای ترکیب دسته‌بندها با ترکیب روش اختلاط خبرگان و PSO ارائه شده است.

[3] Dempster-Shafer
[4] حوزه‌ای از تحلیل داده که اصولاً برای ادغام شواهدی از منابع مختلف شکل گرفته است.
[5] belief functions
[6] particle swarm optimaztion (PSO)

۴- اندازه‌گیری گوناگونی

همان‌گونه که بیان شد، کارایی بهتر سیستم‌های شورایی به گوناگونی دسته‌بندهای پایه در شورا وابسته است. به‌گونه‌ای که اگر دسته‌بندهای پایه گوناگون در خطا نباشند، صحت دسته‌بندی شورا بهتر از دسته‌های مجزا نخواهد بود. در مرجع *(Brown et al., 2005)*، اهمیت گوناگونی دسته‌بندها به‌طور نظری نشان داده شده که با توجه به محدودیت مقاله، به آن پرداخته نشده است. اغلب روش‌های طراحی شورای دسته‌بندها (که در بخش ۲ معرفی شد)، گوناگونی را به‌طور مستقیم در طراحی روش درنظر نمی‌گیرند؛ بلکه گوناگونی به‌طور ضمنی با استفاده از مجموعه‌های داده یا ویژگی متفاوت حاصل می‌شود. با این حال، برای ارزیابی گوناگونی نهایی شورای طراحی‌شده می‌توان از معیارهای گوناگونی مختلف استفاده کرد. در این بخش، معیارهای ارزیابی گوناگونی به‌اجمال معرّفی شده است. مطالعهٔ گستردهٔ مفهوم و معیارهای گوناگونی در حوزهٔ دسته‌بندی و رگرسیون در *(Brown et al., 2005; Windeatt, 2005)* آورده شده است.

معیارهای مختلفی برای ارزیابی کمّی گوناگونی تعریف شده است. ساده‌ترین آنها معیارهای زوجی[1] هستند که بین دو دسته‌بند تعریف می‌شوند. زمانی که تعداد دسته‌بندها L باشد ($L \geq 2$)، تعداد $L(L-1)/2$ معیار جفتی بین دو دسته‌بند محاسبه می‌شود و گوناگونی کلّ سیستم دسته‌بند چندگانه برابر با میانگین این معیارها خواهد بود.

برای دو دسته‌بند h_i و h_j از نمادهای جدول زیر استفاده می‌کنیم:

(جدول ۲): نمادهای دسته‌بندی برای دو دسته‌بند h_i و h_j

	h_j صحیح است	h_j اشتباه است
h_i صحیح است	a	b
h_i اشتباه است	c	d

که a نسبتی از نمونه‌ها است که با هر دو دسته‌بند صحیح دسته‌بندی شده‌اند، b نسبتی از نمونه‌ها است که با دسته‌بند h_i صحیح دسته‌بندی شده‌اند و با دسته‌بند h_j اشتباه دسته‌بندی شده‌اند. نمادهای c و d نیز مطابق جدول به طرز مشابه تعریف می‌شوند. مشخصاً $a+b+c+d=1$ خواهد بود. بر اساس این نمادها معیارهای گوناگونی زیر ارائه شده است:

□ همبستگی[2]: گوناگونی بر اساس همبستگی بین خروجی دو دسته‌بند سنجیده می‌شود که طبق رابطهٔ ۳۰ تعریف شده است:

$$\rho_{i,j} = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}, \quad 0 \leq \rho \leq 1 \qquad (۳۰)$$

بیشترین گوناگونی برای $\rho=0$ به‌دست می‌آید که نشان می‌دهد دو دسته‌بند ناهمبسته‌اند.

□ آمارهٔ Q[3]: این معیار طبق رابطهٔ زیر تعریف می‌شود:

$$Q_{i,j} = \frac{ad - bc}{ad + bc} \qquad (۳۱)$$

زمانی‌که نمونه‌های یکسان با هر دو دسته‌بند صحیح دسته‌بندی شده باشند این آماره مقادیر مثبت و درغیراین‌صورت مقادیر منفی را اختیار می‌کند. بیشترین گوناگونی برای $Q=0$ به‌دست می‌آید.

□ معیار آنتروپی: این معیار فرض می‌کند زمانی گوناگونی حداکثر است که برای هر نمونه، نیمی از دسته‌بندها صحیح و نیمی از آنها اشتباه دسته‌بندی کنند. براساس این فرض و تعریف ζ_i به‌عنوان تعدادی از دسته‌بندها از میان L دسته‌بند که نمونهٔ x_i را اشتباه دسته‌بندی می‌کنند، معیار آنتروپی به‌صورت زیر تعریف می‌شود:

$$E = \frac{1}{N}\sum_{i=1}^{N} \frac{1}{L - \lceil L/2 \rceil} min\left\{\zeta_i, (T - \zeta_i)\right\} \qquad (۳۲)$$

که $\lceil . \rceil$ جزء صحیح عدد است؛ و N بُعد مجموعه داده است. معیار آنتروپی بین ۰ و ۱ است؛ که ۰ نشان‌دهندهٔ یکسان بودن تمام دسته‌بندها و ۱ نشان‌دهندهٔ بیشترین گوناگونی است *(Kuncheva and Whitaker, 2003)*.

۵- حوزه‌های پژوهشی فعّال

با وجود دو دهه پژوهش، برخی از پژوهشگران معتقدند این حوزه از یادگیری ماشینی، به بلوغ خود رسیده است. با این حال، به‌نظر می‌رسد سیستم‌های شورایی توجه زیادی را به خود جلب خواهد کرد. این موضوع تا حدّ زیادی به علّت کاربردهای جدیدی است که از سیستم‌های شورایی بهره می‌برند. اخیراً کاربردها و زمینه‌های پژوهشی نوینی در حوزهٔ سیستم‌های دسته‌بند چندگانه شکل گرفته است. در این بخش، برخی از مسیرهای آتی پژوهش و زمینه‌های پژوهشی فعال در حوزهٔ سیستم‌های شورایی که به نظر مؤلفان مقاله فراخور پژوهش بیشتر هستند، آورده شده است:

[1] pair-wise

[2] correlation

[3] Q-Statistic

۵-۱- یادگیری تدریجی[1]

در برخی مسائل کاربردی، مجموعه دادهٔ مسأله در ابتدا موجود نیست و به‌تدریج در بسته‌های مختلف دردسترس قرار می‌گیرد. علاوه‌براین ممکن است نمونه‌های بعدی شامل نمونه‌هایی از دسته‌ای جدید باشد. لذا لازم است بدون فراموش کردن دانش پیشین و بدون نیاز به نمونه‌های قبل، اطلاعات نمونه‌های جدید نیز یادگرفته شود. توانایی دسته‌بند برای یادگیری تحت این شرایط، به‌طور معمول یادگیری تدریجی خوانده می‌شود. رویکردی عملی برای یادگرفتن از نمونه‌های جدید صرف‌نظر کردن از دسته‌بند موجود و استفاده از دسته‌بندی جدید با استفاده از تمام نمونه‌های است که تاکنون به‌دست آمده است. بااین‌حال، این رویکرد منجر به ازدست دادن تمام دانش به‌دست‌آمده تا آن مرحله می‌شود؛ مشکلی که فراموشی فاجعه‌آمیز[2] خوانده می‌شود. درصورتیکه که زمان یا هزینهٔ محاسباتی زیادی برای آموزش دوباره دسته‌بند لازم باشد، این رویکرد مطلوب نیست. مهم‌تر از این، اگر مجموعه دادهٔ اولیه حذف شده یا آسیب دیده باشد یا به‌طورکلی دردسترس نباشد[3] این رویکرد عملی نیست.

استفاده از سیستم‌های شورایی در این مسائل منجر به نتایج خوبی شده است. ایدهٔ اصلی، ایجاد شوراهایی جدید از دسته‌بندها با هرمجموعه دادهٔ جدید و سپس ترکیب خروجی آنها با استفاده از یکی از روش‌های معمول است. الگوریتم Learn++[11] مهمترین الگویتم ارائه شده در این زمینه است که در کاربردهای مختلف، حتی زمانی که نمونه‌های بعدی شامل نمونه‌هایی از دسته‌هایی جدید باشد، نتایج خوبی را نشان داده است (*Polikar et al. 2004; R.Polikar et al. 2001*).

۵-۲- توسعهٔ سیستم‌های شورایی در انواع دیگر مدل‌های یادگیری

اغلب پژوهش‌ها در سیستم‌های شورایی در حوزهٔ دسته‌بندی ساده انجام شده است. مطالعات بیشتری برای به‌کارگیری این رویکرد در حوزه‌های دیگر یادگیری ماشین مانند خوشه‌بندی، رگرسیون، سری زمانی و دیگر حوزه‌های یادگیری با نظارت مانند یادگیری فعّال[4] یا دسته‌بندی توالی[5] زمینه‌های پژوهشی مناسبی خواهد بود.

۵-۳- ارتباط گوناگونی با مسائل دسته‌بندی

روابط ریاضی بین کارایی سیستم شورا با گوناگونی آن برای مسائل رگرسیون ارائه شده است (Brown et al. 2005). با این حال، تاکنون این گونه روابط برای مسائل دسته‌بندی به‌دست نیامده است.

۵-۴- انتخاب دسته‌بند

رویکرد مشابه ترکیب دسته‌بندها در سیستم‌های شورایی، انتخاب دسته‌بند[6] است (Woods, Kegelmeyer, and Bowyer 1997). در ادغام دسته‌بندها، هر دسته‌بند از تمام نمونه‌های فضای ویژگی استفاده می‌کند. در این حالت، فرایند ادغام شامل ترکیب دسته‌بندهای پایه و ضعیف‌تر برای دست‌یابی به دسته‌بند باکارایی بیشتر است. در انتخاب دسته‌بند، هر دسته‌بند در شورا بخشی از فضای ویژگی را به‌خوبی یاد می‌گیرد و لذا دسته‌بندی نمونه‌های آن بخش از فضا را انجام می‌دهد. ازاین‌رو، در ادغام دسته‌بندها از روش‌های ترکیب (که در فصل پیش معرّفی شده است) برای تعیین دستهٔ الگوی x استفاده می‌کنیم؛ درحالی‌که در رویکرد انتخاب، به‌طور معمول یک دسته‌بند انتخاب می‌شود (*Woods, Kegelmeyer, and Bowyer, 1997*).

رویکرد انتخاب دسته‌بند همانند ادغام دسته‌بند توجه زیادی را به‌خود جلب نکرده است. این مسأله ممکن است درآینده تغییر کند؛ چرا که روش انتخاب دسته‌بند کارایی خوبی را نشان داده است (*Kuncheva, 2004*). دسته‌بندهای آبشاری[7] (*Zhang, Bui, and Suen, 2007*) نیز از دیگر روش‌های رویکرد انتخاب ویژگی است که باوجود توانایی آنها در کاربردهای عملی، پژوهش‌های بسیار کمی به آن پرداخته‌اند. در این روش، تنها یک دسته‌بند در هر لحظه فعّال است و سایر دسته‌بندها غیرفعّال هستند. برای دسته‌بندی نمونهٔ x، اوّلین دسته‌بند سعی در دسته‌بندی آن می‌کند. اگر دسته‌بند در تصمیم خود، *مطمئن*[8] است، x برچسب‌گذاری می‌شود و فرآیند متوقف می‌شود. درغیراین‌صورت، نمونهٔ x به دسته‌بند بعدی می‌رود و این روّیه ادامه می‌یابد.

۵-۵- تخمین اطمینان[9]

استفاده از سیستم‌های شورایی همچنین می‌تواند برای ارزیابی کمی میزان اطمینان تصمیم به‌کار گرفته شود. به

---

[1] incremental learning
[2] catastrophic forgetting
[3] این حالتها در بسیاری از مسائل پزشکی یا نظامی که دسترسی به مجموعه داده محدود یا محرمانه است، مرسوم است.
[4] active learning
[5] sequence classification
[6] classifier selection
[7] cascade classifier
[8] certain
[9] confidence estimation

زبان ساده، اگر اغلب دسته‌بندها در دسته‌بندیِ یک الگو تصمیم یکسانی داشته باشند، این تصمیم شورا اطمینان بالایی خواهد داشت. متقابلاً، اگر تصمیم نهایی براساس تفاوت زیاد نظرات دسته‌بندها گرفته شده باشد، تصمیم شورا اطمینان کمی خواهد داشت. نشان داده شده که خروجی‌های امتیازدار دسته‌بندها می‌توانند برای تخمین احتمال پسین تصمیم استفاده شود؛ که به نوعی نشان‌دهندهٔ اطمینان تصمیم است *(Muhlbaier, Topalis, and Polikar 2005)*.

۶- جمع‌بندی و نتیجه‌گیری

در این مقاله تلاش شده است سیستم‌های دسته‌بند چندگانه (سیستم‌های شورایی) به‌عنوان یکی از زمینه‌های فعّال در حوزهٔ یادگیری ماشین معرفی شود. دو بخش اساسی در این سیستم‌ها که دو عامل تعیین کنندهٔ موفقیت آنهاست به‌اجمال بیان شده است. عامل مهمّ اوّل، ایجاد گوناگونی در شوراست که با روش‌های مختلف طراحی شورای دسته‌بندها ایجاد می‌شود. رویکردهای زیرنمونه، زیرفضا و رویکرد ایجاد شورا با دسته‌بندهایی از انواع مختلف معرفی شده است. عامل مهمّ دوم، روش ترکیب دسته‌بندها است: روش‌های مختلف ترکیب براساس نوع خروجی دسته‌بندها تقسیم‌بندی شده و به‌اختصار آورده شده است. همچنین سعی شده هر روش با ذکر مثالی ساده توضیح داده شود. این دو عامل مهم به‌گونه‌ای مکمّل یکدیگرند. به‌گونه‌ای که مجموعهٔ ایده‌آلی از دسته‌بندها می‌تواند تا حد زیادی ایجاد قاعدهٔ ترکیب را ساده کند و قاعدهٔ ترکیبی قوی می‌تواند حتّی با شورای دسته‌بندهای ضعیف، نتایج خوبی را بدهد. در پایان با بررسی پژوهش‌های اخیر در این حوزه، زمینه‌های پژوهشی پیشنهادی ارائه شده است. امید است این مقاله چراغ راهی برای شروع پژوهش‌های آتی در حوزهٔ سیستم‌های شورایی باشد.

پیوست الف) نرم افزارهای پیاده‌سازی الگوریتم‌های شورایی

بسته‌های نرم‌افزاری زیر شامل توابع برمبنای MATLAB هستند که می‌توان برای پیاده‌سازی الگوریتم‌های شورا استفاده شوند:

- C. Merkwinh and J. Wichard, "ENTOOL—A Matlab Toolbox for Ensemble Modeling," 2002. Available online at: http://chopin.zet.agh.edu.pl/~wichtel
- R. P.W. Duin, D. Ridder, P. Juszczak, P. Paclik, E. Pekalska and D. Tax, "PRTools," 2005. Available online at: http://www.prtools.org
- D. Stork and E. Yom-Tov, "Computer Manual in Matlab to Accompany Pattern Classification, 2nd Edition, New York: Wiley, 2004.

همچنین دردسترس بودن مهم‌ترین الگوریتم‌های شورایی در نرم‌افزارهای معروف متن‌باز در (جدول ۳) نشان داده شده است:

(جدول ۳): دردسترس بودن الگوریتم‌های مهم ایجاد شورا در نرم‌افزارهای متن‌باز

الگوریتم	نرم‌افزارهای موجود
AdaBoost	Weka, Orange, Tanagra, RapidMiner, R
Bagging	All
Random Forest	All
Wagging	Weka, RapidMiner
Attribute Bagging	Weka
Stacking	Weka, Tanagra, RapidMiner
ECOC	Weka, RapidMiner

شایان ذکر است برای استفاده سایر پژوهش‌گران در این حوزه، گُد پیاده‌سازی ده روش مهم ترکیب شورا نوشته شده که از پایگاه http://www.mathworks.com/matlabcentral/fileexchange/ قابل دسترسی است.

۷- منابع

Allwein, Erin L., Robert E. Schapire, and Yoram Singer. 2001. Reducing multiclass to binary: a unifying approach for margin classifiers. Journal of Machine Learning Research 1:113-141.

Bagheri, Mohammad Ali, and Gholam Ali Montazer. 2011. Ensemble Classifier Strategy Based on Transient Feature Fusion in Electronic Nose. Paper read at 14th international symposium on olfaction and electronic nose, at New York City, NY, USA.

Banfield, R. E., L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer. 2007. A Comparison of Decision Tree Ensemble Creation Techniques. IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (1):173-180.

induced by multivalued mappings. Annals of Mathematical Statistics 38 (2):325–339.

Dietterich, T. G. 1997. Machine learning research: Four current directions. Artificial Intell. Mag. 18(4):97-136.

Dietterich, T.G., and G. Bakiri. 1995. Solving Multiclass Learning Problems via Error-Correcting Output Codes. Journal of Artificial Intelligence Research 2:263–286.

Dietterich, Thomas. 2000. Ensemble Methods in Machine Learning. Lecture Notes in Computer Science 1857:1-15.

Ebrahimpour, Reza, Ehsanollah Kabir, Hossein Esteky, and Mohammad Reza Yousefi. 2008. View-independent face recognition with Mixture of Experts. Neurocomputing 71 (4-6):1103-1107.

Ebrahimpour, Reza, Ehsanollah Kabir, and Mohammad Yousefi. 2011. Improving mixture of experts for view-independent face recognition using teacher-directed learning. Machine Vision and Applications 22 (2):421-432.

Finlay, Steven. 2011. Multiple classifier architectures and their application to credit risk assessment. European Journal of Operational Research 210 (2):368-378.

Freund, Y., and R. Schapire. 1996. Experiments with a new boosting algorithm. Paper read at Proceeding of the Thirteenth International Conference on Machine Learning.

Freund, Y ,.and R.E. Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences 55 (1):119–139.

Fumera, G., F. Roli, and A. Serrau. 2008. A Theoretical Analysis of Bagging as a Linear Combination of Classifiers. Pattern Analysis and Machine Intelligence, IEEE Transactions on 30 (7):1293-1299.

Gey, Servane, and Jean-Michel Poggi. 2006. Boosting and instability for regression trees. Computational Statistics & Data Analysis 50 (2):533-550.

Giacinto, G., and F. Roli. 2001. An approach to the automatic design of multiple classifier systems. Pattern Recognition Letters 22:25–33.

Bao, Y., and N. Ishii. 2002. Combining multiple K-nearest neighbor classifiers for text classification by reducts. Paper read at Proceedings of the 5th International Conference on Discovery Science, Lecture Notes in Computer Science, at Berlin.

Bauer, Eric, and Ron Kohavi. 1999. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. Machine Learning 36 (1):105-139.

Bloch, I. 1996. Information Combination Operators for Data Fusion: a Comparative Review with Classification. IEEE Trans. on Systems, Man and Cybernetics - Part A: Systems and Humans 26:52-67.

Breiman, L. 1996. Bagging Predictors .Machine Learning 24:123-140.

.————2001 .Random Forests. Machine Learning 45 (1): 5–32.

Brodley, arla, and Terran Lane. 1996. Creating and Exploiting Coverage and Diversity. Paper read at Proceedings of AAAI-96 Workshop on Integrating Multiple Learned Models, , at Portland, OR.

Brown, Gavin, Jeremy Wyatt, Rachel Harris, and Xin Yao. 2005. Diversity creation methods: a survey and categorisation. Information Fusion 6 (1):5-20.

Bryll, R., R. Gutierrez-Osuna, and F. Quek. 2003. Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets. Pattern Recognition 36 1291–1302.

.————2003 .Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets. Pattern Recognition 36 1291–1302.

Chen, Shih-Chieh ,Shih-Wei Lin, and Shuo-Yan Chou. 2011. Enhancing the classification accuracy by scatter-search-based ensemble approach. Applied Soft Computing 11 (1):1021-1028.

Cunningham, P., and J. Carney. 2000. Diversity versus quality in classification ensembles based on feature selectio. In R.L. de Mántaras, E. Plaza (Eds.), Proceedings of the ECML 2000. Barcelona, Spain: Springer, Berlin.

Dasarathy, B. V., and B. V. Sheela. 1979. A composite classifier system design: Concepts and methodology. Proceedings of the IEEE 67 (5):708-713.

Dempster, A.P. 1967. Upper and lower probabilities

computation 6(2)-181-214.

Kim, Myoung-Jong, and Dae-Ki Kang. 2010. Ensemble with neural networks for bankruptcy prediction. Expert Systems with Applications 37 (4):3373-3379.

Kotsiantis, S., K. Patriarcheas, and M. Xenos. 2010. A combinational incremental ensemble of classifiers as a technique for predicting students' performance in distance education. Knowledge-Based Systems 23 (6):529-535.

Krogh, A., and J. Vedelsby. 1995. Neural network ensembles, cross validation, and active learning. In Advances in Neural Information Processing Systems, edited by D. Touretzky and T. Leen. Cambridge, MA: MIT Press.

Kuncheva, L. I. 2003. "Fuzzy" versus "nonfuzzy" in combining classifiers designed by Boosting. Fuzzy Systems, IEEE Transactions on 11 (6):729-741.

.————2004 .Combining Pattern Classifiers: Methods and Algorithms. New York, NY: Wiley.

Kuncheva, L. I., J. C. Bezdek, and R. P. W. Duin. 2001. Decision Templates for Multiple Classifier Fusion: An Experimental Comparison. Pattern Recognition 34 (2):299-314.

Kuncheva, L. I., and L. C. Jain. 2000. Designing classifier fusion systems by genetic algorithms. Evolutionary Computation, IEEE Transactions on 4 (4):327-336.

Kuncheva, L. I., M. Skurichina, and R. P. W. Duin. 2000. An experimental study on diversity for bagging and boosting with linear classifiers. Information Fusion 3 (4):245-258.

Kuncheva, L., and C. Whitaker. 2003. "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine Learning 51 (2):181–207.

Kunchcva, Ludmila I. 2001 .Using measures of similarity and inclusion for multiple classifier fusion by decision templates. Fuzzy Sets and Systems 122 (3):401-407.

Lam, L., and C. Y. Suen. 1997. Application of majority voting to pattern recognition: An analysis of its behavior and performance. IEEE Transactions on Systems, Man, and Cybernetics 27 (5):553-568.

Govindarajan, M., and R. M. Chandrasekaran. Intrusion detection using neural based hybrid classification methods. Computer Networks In Press, Corrected Proof.

Hansen, L. K., and P. Salamon. 1990. Neural network ensembles. Pattern Analysis and Machine Intelligence, IEEE Transactions on 12 (10):993-1001. Hansen, L., and P. Salamon. Oct. 1990. Neural network ensembles. IEEE Trans. on Pattern Analysis and Machine Intelligence 12 (10):993-1001.

Hashem, S. 1997. Optimal linear combinations of neural networks: an overview,. Neural Networks 10 (4):599–614.

Haykin, S., and N. Network. 2004. A comprehensive foundation. Neural Networks 2.

Ho, Tin. K. 1998. The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20:832-844.

Hong, Yi, Sam Kwong, Yuchou Chang, and Qingsheng Ren. 2008. Unsupervised feature selection using clustering ensembles and population based incremental learning algorithm. Pattern Recognition 41 (9):2742-2756.

Hu, Q.H., D.R. Yu, and M.Y. Wang. 2005. Constructing rough decision forests. Paper read at D. Slezak et al. (Ed.), RSFDGrC 2005, Lecture Notes in Artificial Intelligence, at Berlin.

Hu, Qinghua, DarenYu, Zongxia Xie, and Xiaodong Li. 2007. EROS: Ensemble rough subspaces. Pattern Recognition 40 3728 – 3739.

Huang, Y.S., and C.Y. Suen. 1993. Behavior-knowledge space method for combination of multiple classifiers. Paper read at IEEE Computer Vision and Pattern Recog.

.————1995 .A method of combining multiple experts for the recognition of unconstrained handwritten numerals. IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (1):90–94.

Jacobs, R.A., M.I. Jordan, S.J. Nowlan, and G.E. Hinton. 1991. Adaptive mixtures of local experts. Neural Computation 3:79-87.

Jordan, M.I., and R.A. Jacobs. 1994. Hierarchical mixtures of experts and the EM algorithm. Neural

signals. IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control 51 (8):990–1001.

R.Polikar, L. Udpa, S.S. Udpa, and V. Honavar. 2001. Learn++: An incremental learning algorithm for supervised neural networks. IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews 31 (4):497–508.

Rodrguez, J.J., L.I. Kuncheva, and C.J. Alonso. 2006. Rotation forest: A new classifier ensemble method. IEEE Trans. Pattern Anal. Machine Intell 28 (10):1619–1630.

Rokach, Lior. 2008. Genetic algorithm-based feature set partitioning for classification problems. Pattern Recognition 41 (5):1676 - 1700.

.————2008 .Mining manufacturing data using genetic algorithm\C45;based feature set decomposition. Int. J. Intell. Syst. Technol. Appl. 4 (1/2):57-78.

Ruta, D., and B. Gabrys. 2000. An Overview of Classifier Fusion Methods. Computing and Information Systems 7 (1):1-10.

Schapire, Robert E. 1990. The strength of weak learnability. Machine Learning 5 (2):197-227.

Shafer, G. 1976. A Mathematical Theory of Evidence. Princeton, NJ: Princeton Univ. Press.

Sharkey, Amanda, Noel Sharkey, Uwe Gerecke, and G. Chandroth. 2000. The "Test and Select" Approach to Ensemble Combination. In Multiple Classifier Systems: Springer Berlin / Heidelberg.

Shi, Lei, Xinming Ma, Lei Xi, Qiguo Duan, and Jingying Zhao. 2011. Rough set and ensemble learning based semi-supervised algorithm for text classification. Expert Systems with Applications 38 (5):6300-6306.

Skurichina, M., and Robert P. W. Duin. 2002. Bagging, Boosting and the Random Subspace Method for Linear Classifiers. Pattern Analysis & Applications 5:121-135.

Tsymbal, A., and S. Puuronen. 2002. Ensemble feature selection with the simple Bayesian classification in medical diagnostics. Paper read at Proceedings of the 15th IEEE Symposium on Computer-Based Medical Systems CBMS'2002, at

Lin, Lei, Xiaolong Wang, and Daniel Yeung. 2005. Combining multiple classifiers based on a statistical method for handwritten chinese character recognition. International Journal of Pattern Recognition and Artificial Intelligence 19 (8):1027-1040.

Meyer, David, Friedrich Leisch, and Kurt Hornik. 2003. The support vector machine under test. Neurocomputing 55 (1-2):169-186.

Muhlbaier, M., A. Topalis, and R. Polikar. 2005. Ensemble confidence estimates posterior probability. In 6th Int. Workshop on Multiple Classifier Sys., edited by N. Oza, R. Polikar, J. Kittler and F. F.Roli.

Nabavi-Kerizi, S. H., M. Abadi, and E. Kabir. 2010. A PSO-based weighting method for linear combination of neural networks. Computers & Electrical Engineering 36 (5):886-894.

Nanni, Loris, and Alessandra Lumini. 2009. Ensemble generation and feature selection for the identification of students with learning disabilities. Expert Systems with Applications 36 (2, Part 2):3896-3900.

Opitz, David W. 1999. Feature selection for ensembles. In Proc. 16th National Conf. on Artificial intelligence, AAAI. Orlando, Florida, United States: American Association for Artificial Intelligence.

Oza, N.C., and K. Tumer. 2001 .Input decimation ensembles: Decorrelation through dimensionality reduction. In 2nd Int. Workshop on Multiple Classifier Systems in Lecture Notes in Computer Science, edited by J. Kittler and F. Roli.

Peng, Yi, Guoxun Wang, Gang Kou, and Yong Shi. 2001 .An empirical study of classification algorithm evaluation for financial risk prediction. Applied Soft Computing 11 (2):2906-2915.

Pino-Mejías, Rafael, María Dolores Cubiles-de-la-Vega, María Anaya-Romero, Antonio Pascual-Acosta, Antonio Jordán-López, and Nicolás Bellinfante-Crocci. 2010. Predicting the potential habitat of oaks with data mining models and the R system. Environmental Modelling & Software 25 (7):826-836.

Polikar, R. 2006. Ensemble based systems in decision making. IEEE Circuits and Systems Magazine 6 (3):21-45.

Polikar, R., L. Udpa, S. Udpa, and V. Honavar. 2004. An incremental learning algorithm with confidence estimation for automated identification of NDE

of combining multiple classifiers and their application to handwriting recognition. IEEE Transactions on Systems, Man, and Cybernetics 22:418–435.

Yang, L.Y., Z. Qin, and R. Huang. 2004. Design of a multiple classifier system. Paper read at IEEE Proceedings of the Third International conference on Machine Learning and Cybernetics, August 26–29, at Shanghai.

Yu, Enzhe, and Sungzoon Cho. 2006. Ensemble based on GA wrapper feature selection. Computers & Industrial Engineering 51 (1):111-116.

Zhang, Chun-Xia, and Jiang-She Zhang. 2008. RotBoost: A technique for combining Rotation Forest and AdaBoost. Pattern Recognition Letters 29(10):1524-1536.

Zhang, Ping, Tien D. Bui, and ChingY. Suen. 2007. A novel cascade ensemble classifier system with a high recognition performance on handwritten digits. Pattern Recognition 40 3415 – 3429.

نبوی‌کریزی، سید حسن، و احسان اله کبیر، ۱۳۸۴، ترکیب طبقه‌بندها: ایجاد گوناگونی و قواعد ترکیب. علوم و مهندسی کامپیوتر مجلد ۳، شماره ۳ (الف)، صفحات ۹۵- ۱۰۷

نبوی‌کریزی، سید حسن، واحسان اله کبیر، ۱۳۸۷. یک روش دو مرحله ای برای ترکیب طبقه‌بندها. نشریه مهندسی برق و مهندسی کامپیوتر ایران ۶ (۱):۶۳-۷۰

Maribor, Slovenia.

Tsymbal, Alexey, Mykola Pechenizkiy, and Pádraig Cunningham. 2005. Diversity in search strategies for ensemble feature selection. Information Fusion 6 (1):83-98.

Tumer, K., and J. Ghosh. August 1996. Classifier combining: analytical results and implications, . Paper read at 13th National Conference on Artifical Intelligence, at Portland, Oregon.

Tumer, K., and N.C. Oza. 2003. Input decimated ensembles. Pattern Anal. Appl. 6:65–77.

Tutz, Gerhard, and Harald Binder. 2007. Boosting ridge regression. Computational Statistics & Data Analysis 51 (12):6044-6059.

Wanas, N. M., and M. S. Kamel. 2001. Decision fusion in neural network ensembles. Paper read at Proceedings of international joint conference on neural networks.

Wang, Shi-jin, Avin Mathew, Yan Chen, Li-feng Xi, Lin Ma, and Jay Lee. 2009. Empirical analysis of support vector machine ensemble classifiers. Expert Systems with Applications 36 (3, Part 2):6466-6476.

Wanga, Shu-Lin, Xueling Li, Shanwen Zhang, Jie Gui, and De-Shuang Huang. 10 .Tumor classification by combining PNN classifier ensemble with neighborhood rough set based gene reduction. Computers in Biology and Medicine 40:179–189.

Windeatt, Terry. 2005. Diversity measures for multiple classifier system analysis and design. Information Fusion 6 (1):21-36.

Windeatt, Terry, and Reza Ghaderi. 2003. Coding and decoding strategies for multi-class learning problems. Information Fusion 4 (1):11-21.

Woods, K., W.P.J. Kegelmeyer, and K. Bowyer. 1997. Combination of multiple classifiers using local accuracy estimates. IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (4):405–410.

Wu, QingXiang, and David Bell. 2003. Multi-knowledge Extraction and Application. In Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing ,edited by G. Wang, Q. Liu, Y. Yao and A. Skowron: Springer Berlin / Heidelberg.

Xu, L., A. Krzyzak, and C. Y. Suen. 1992. Methods

محمّدعلی باقری در سال ۱۳۸۴ مدرک کارشناسی خود را در رشتهٔ مهندسی صنایع دانشگاه صنعتی شریف و در سال ۱۳۸۷ مدرک کارشناسی ارشد خود را در رشتهٔ مهندسی فناوری اطّلاعات از دانشگاه تربیت مدرس دریافت نمود. وی در حال حاضر دانشجوی دکترای مهندسی فناوری اطّلاعات در دانشگاه تربیت مدرس است. زمینه‌های پژوهشی مورد علاقهٔ ایشان شامل شناسایی الگو به‌ویژه سیستم‌های شورایی، بویایی هوشمند (بینی الکترونیکی) و روش‌های بهینه‌سازی تکاملی است.

نشانی رایانامک ایشان عبارتست از:

ma.bagheri@gmail.com

غلامعلی منتظر در سال ۱۳۴۸ در کازرون(فارس) به دنیا آمد. وی در سال ۱۳۷۰ مدرک کارشناسی خود را در رشتهٔ مهندسی برق از دانشگاه صنعتی خواجه نصیرالدین طوسی و سپس در سال ۱۳۷۳ و ۱۳۷۷ مدارک کارشناسی ارشد و دکتری خود را در همین رشته از دانشگاه تربیت مدرس اخذ کرد. پس از اتمام تحصیل به عضویت هیئت علمی دانشگاه تربیت مدرس درآمد و در حال حاضر دانشیار مهندسی فنّاوری اطلاعات در این دانشگاه است. حوزه‌های تخصصی ایشان شامل نرم‌رایانش (نظریهٔ مجموعه‌های فازی، شبکه‌های عصبی مصنوعی، نظریهٔ مجموعه‌های نادقیق) و کاربرد آن در سیستم‌های اطّلاعاتی (همچون سیستم یادگیری الکترونیکی و تجارت الکترونیکی) است. وی تاکنون بیش از ۷۰ مقاله در نشریات معتبر علمی و بیش از ۱۵۰ مقاله در کنفرانس‌های معتبر علمی ملّی و بین‌المللی منتشر کرده است. علاوه‌بر این حائز دریافت جوایزی معتبر علمی از جمله «برگزیدهٔ جشنوارهٔ بین‌المللی خوارزمی»، «برندهٔ کتاب سال دانشگاهی ایران»، «پژوهشگر برگزیدهٔ آیسسکو» و «متخصص برجسته فنّاوری اطّلاعات ایران» شده است.

نشانی رایانامک ایشان عبارتست از:

montazer@modars.ac.ir

احسان‌اله کبیر کارشناسی ارشد پیوستهٔ خود را در مهندسی برق و الکترونیک از دانشکدهٔ فنی دانشگاه تهران و دکترای خود را در مهندسی سیستم‌های الکترونیک از دانشگاه اسکس در انگلستان، به ترتیب در سال‌های ۱۳۶۴ و ۱۳۶۹ دریافت کرد.

وی اکنون استاد دانشکدهٔ مهندسی برق و کامپیوتر دانشگاه تربیت مدرس است. زمینهٔ پژوهشی مورد علاقهٔ ایشان بازشناسی الگو، به ویژه بازشناسی متون چاپی و دست‌نویس است.

نشانی رایانامک ایشان عبارتست از:

kabir@modares.ac.ir