

فصل سوم

شناسایی کنندةها

3-1: مقدمه

در دنیای امروز شروع هر فعالیت و کسب و کار نیازمند مطالعات دقیق در زمینه بازار، تکنولوژی، اقتصاد، مکان اجرا، محیط، استراتژی، نظام اقتصادی حاکم بر عملیات تولید و بازرگانی و محیط زیست می باشد که به دلیل حجم بالای اطلاعات مورد بررسی می بایستی طبقه بندی، فیلتر و در نهایت گزارشات اختیاری اخذ گردد.

اطلاعات بدست آمده از یک تحقیق غالباً توده ای از اطلاعات خام و بدون نظم هستند که هر نوع نتیجه گیری و تفسیر آن ها غیرممکن است. بنابراین برای هر نوع تجزیه و تحلیل اطلاعات لازم است داده ها بر اساس یک نظم منطقی طبقه بندی شوند تا به صورت معنی دار و قابل تفسیر در آیند. هدف دسته بندی داده ها، سازماندهی و تخصیص داده ها به کلاس های مجزا است. در این فصل به بررسی انواع شناسایی کننده ها و نحوه عملکرد و تعدادی از الگوریتم های موجود در هر یک از آن ها می پردازیم.

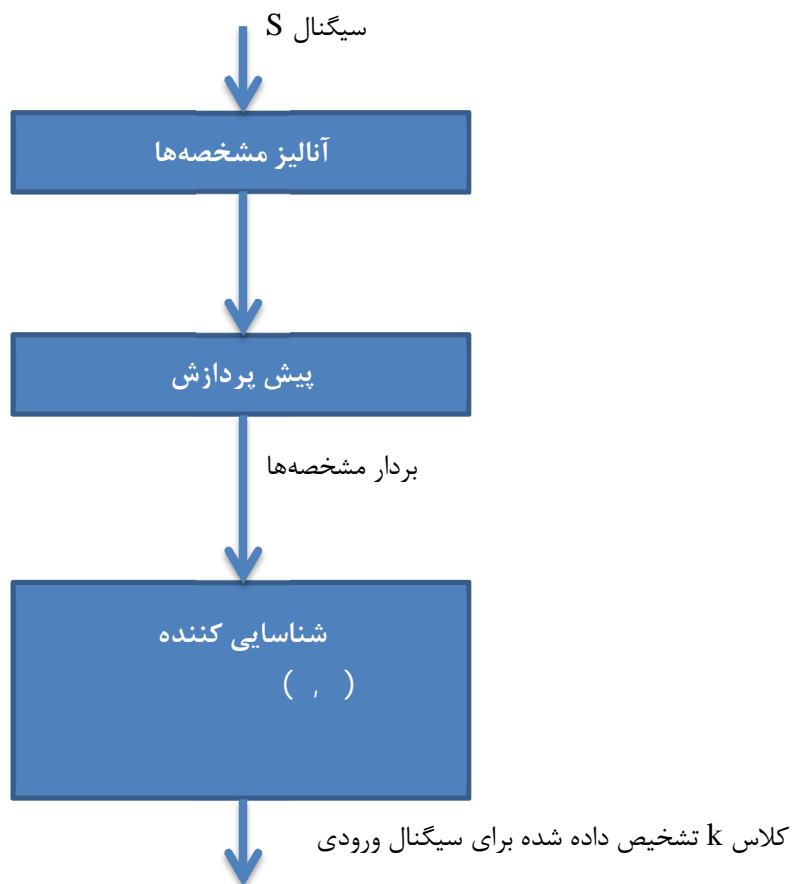
همان گونه که در دو پاراگراف اول این فصل مشاهده نمودید برای واژه Classification معانی مختلفی در زبان فارسی استفاده شده است. این واژه توسط محققین مختلف با نام های شناسایی کننده، طبقه بندی کننده، کلاسه بند، دسته بندی کننده و نام هایی از این قبیل نام گذاری شده است. از این رو در بخش های مختلف این فصل از این نام ها استفاده شده است تا در صورت برخورد با این نام ها در مراجع دیگر با آن ها آشنا باشید.

3-2: عملکرد شناسایی کننده ها

سیستم های شناسایی کننده یک سیگنال s (مانند یک سیگنال صوتی، یک تصویر دیجیتال، یک تصویر ماهواره ای یا یک متن خبری) را به عنوان ورودی دریافت نموده و با آنالیز سیگنال ورودی و عملیات پیش پردازش، یک بردار X به عنوان بردار ویژگی های سیگنال ورودی را تشکیل می دهند که به این مرحله استخراج ویژگیها¹ گفته می شود. در ادامه یک تابع

¹ Feature extraction

تشخیص‌دهنده به نام $g(x,k)$ تعیین می‌کند که احتمال اینکه سیگنال ورودی S به کلاس k تعلق داشته باشد، چقدر است. دسته k که بتواند بیشترین مقدار را برای تابع تشخیص‌دهنده g با بردار ویژگی X به وجود آورد، کلاس مربوط به سیگنال ورودی خواهد بود.



شکل 3-1: شناسایی‌کننده آماری الگو

عموم روش‌های دسته‌بندی، از یک الگوریتم آموزش و آزمون کمک می‌گیرند. به این ترتیب که داده‌های دسته‌های مختلف را به دو بخش آموزش و آزمون تقسیم می‌کنند. این دو مجموعه

کاملاً از هم مجزا هستند و نباید هیچ‌گونه اشتراکی با هم داشته باشند. سپس بر اساس یک الگوریتم مشخص و بر اساس داده‌های آموزش، الگوریتم مشخص شده آموزش می‌بیند و در نهایت مطلوبیت کارکرد الگوریتم با داده‌های آزمون محاسبه می‌شود.

3-2-1: آموزش

مجموعه آموزش²، مجموعه‌ای از داده‌ها است که در زمینه‌های مختلف علم اطلاعات برای کشف ارتباطات قابل پیش‌بینی مورد استفاده قرار می‌گیرد. مجموعه‌های آموزش در عرصه‌های مختلف مانند هوش مصنوعی، یادگیری ماشین، برنامه نویسی ژنتیک، سیستم‌های هوشمند و آمار استفاده می‌شود. در همه این زمینه‌ها، مجموعه آموزش نقش مشابهی دارد و در کنار مجموعه تست و اعتبارسنجی مورد استفاده قرار می‌گیرد.

در مسائل دسته‌بندی، اندازه‌گیری کارایی شناسایی‌کننده در رابطه با میزان خطا صورت می‌گیرد. شناسایی‌کننده کلاس هر نمونه را پیش‌بینی می‌کند. اگر دسته‌بندی درست باشد عمل موفقیت آمیز است وگرنه خطا محسوب می‌شود. اما باید توجه داشته باشیم که شناسایی‌کننده پیش از آنکه وارد مرحله تست یا آزمون شود نیاز دارد به اینکه برای عملیات شناسایی تعریف و آماده شود. مسلماً نمی‌توان یک الگوریتم کلی برای یک شناسایی‌کننده تعریف کرد به گونه‌ای که برای دسته‌بندی داده‌های مختلف مورد استفاده قرار گیرد. از طرف دیگر، برای هر مساله شناسایی یا دسته‌بندی نمی‌توان یک سیستم شناسایی‌کننده خاص منظوره مجزا با شرایط ویژه تعریف کنیم. لذا یک سیستم شناسایی‌کننده با یک الگوریتم و قالب کلی تعریف و طراحی می‌شود به گونه‌ای که با تغییر پارامترهای آن بتواند برای مسائل شناسایی و دسته‌بندی مختلف مورد استفاده قرار گیرد. تعیین این پارامترها آموزش نامیده می‌شود که این مرحله شامل تخمین پارامترهای سیستم با استفاده از داده‌هایی است که با نام داده‌های آموزش شناخته می‌شوند.

² Train set

3-2-2: تست

یک مدل معمولاً توسط ماکزیمم کردن کارایی روی مجموعه‌ای از داده‌های آموزشی، آموزش می‌بیند اما تاثیر آن بر اساس کارایی روی داده‌های آموزشی تعیین نمی‌شود، بلکه بر اساس توانایی انجام آن روی داده‌های جدید بررسی می‌شود. میزان خطا روی مجموعه آموزش، شاخص خوبی برای کارایی آینده مدل نیست. زیرا طبقه‌بندی‌کننده از داده‌های آموزشی، آموزش دیده است اما داده‌های جدید دقیقاً همان داده‌های آموزشی نخواهند بود. هر تخمینی درباره کارایی بر اساس آن داده‌ها خوش‌بینانه خواهد بود و ممکن است مایوس‌کننده باشد. به همین دلیل مجموعه‌های آموزش و تست، باید از دو مجموعه داده مستقل انتخاب شوند. مجموعه تست³ مجموعه‌ای از داده‌هاست که از مجموعه آموزش مستقل است اما همان توزیع احتمال داده‌های آموزش را پیگیری می‌کند و برای ارزیابی کارایی یک طبقه‌بندی‌کننده که به صورت کامل آموزش دیده، استفاده می‌شود.

3-2-3: اعتبار سنجی⁴

همانگونه که تاکنون مشاهده کردید، تأکیدی وجود دارد که داده‌های آموزش و تست از یکدیگر مستقل بوده و به صورت مشترک از یک داده مشخص برای دو مجموعه استفاده نشود. البته داده‌های آموزش باید تا حد امکان کامل بوده و حالت‌های مختلف جامعه آماری را شامل شوند تا سیستم به صورت کامل آموزش ببیند. حال گاهی اوقات اتفاق می‌افتد که پس از آموزش سیستم شناسایی‌کننده توسط داده‌های آموزشی، وقتی ارزیابی را بر روی داده‌های تست انجام می‌دهیم نتیجه بسیار ضعیفی را شاهد هستیم که این می‌تواند به این علت باشد که یک یا چند پارامتر تعیین‌کننده دیگر در سیستم وجود دارند که با استفاده از داده‌های آموزشی، آموزش ندیده‌اند. در چنین حالتی از دسته دیگری از داده‌ها به نام اعتبارسنجی استفاده می‌شود تا پارامترهای باقیمانده را با استفاده از آنها تنظیم کنند. داده‌های اعتبارسنجی مانند داده‌های تست برای ارزیابی استفاده شده و با هدف بهینه کردن نتیجه عملیات شناسایی بر روی مجموعه

³ Test set or evaluation set

⁴ Validation

اعتبارسنجی پارامترها را تنظیم می‌کنیم. به این مجموعه از داده‌ها است که برای سازگار کردن پارامترهای طبقه‌بندی‌کننده استفاده می‌شوند مجموعه اعتبارسنجی یا توسعه⁵ می‌گوییم. به عبارت دیگر، اگر مدل برای داده‌های آموزشی خیلی مناسب‌تر از داده‌های تست ارزیابی شود، احتمالاً علت آن این است که روی هم افتادگی⁶ اتفاق افتاده است. به منظور اجتناب از این رویداد، وقتی پارامترهای دسته‌بندی نیاز به تنظیم دارد، لازم است علاوه بر مجموعه آموزشی و تست، مجموعه اعتبارسنجی نیز داشته باشیم. مثلاً اگر به دنبال مناسب‌ترین طبقه‌بندی‌کننده برای مساله هستیم، مجموعه آموزشی برای آموزش الگوریتم‌های کاندید استفاده می‌شود و مجموعه اعتبارسنجی برای مقایسه کارایی آن‌ها و تصمیم‌گیری برای انتخاب یکی از آن‌ها استفاده می‌شود و بالاخره مجموعه تست برای به دست آوردن مشخصات کارایی مانند صحت، حساسیت و مانند آن استفاده می‌شود.

3-3: انواع شناسایی‌کننده‌ها⁷

در این فصل به بررسی دو نوع از شناسایی‌کننده‌های بر مبنای مدل⁸ و بدون مدل⁹ می‌پردازیم. شناسایی‌کننده‌های مبتنی بر مدل با فرض قرار دادن یک مدل برای سیستم شناسایی‌کننده از داده‌های آموزشی استفاده می‌کنند تا پارامترهای مدل را تخمین زده و با استفاده از داده‌های تست مدل به دست آمده را ارزیابی می‌کنند، در حالی که در شناسایی‌کننده‌های بدون مدل از تک تک داده‌های آموزشی برای شناسایی نمونه‌های آزمایشی استفاده می‌شود.

3-3-1: شناسایی‌کننده بر مبنای مدل

به طور کلی در این نوع دسته‌بندی‌کننده‌ها، تعدادی دسته با اعضای مشخص وجود دارد که بعضی ویژگی‌های این اعضا نیز مشخص است. در دسته‌بندی‌ها به طور معمول به دنبال یافتن

⁵ Development

⁶ Over-fitting

¹ Classification

² Model based

³ Model free

الگوریتمی هستیم که بر اساس اطلاعات اولیه از دسته‌های مختلف، در صورت ورود یک عضو جدید، تعلق آن به یکی از دسته‌های موجود شناسایی می‌شود. در این پروسه بر اساس مجموعه داده‌های آموزشی، مدل اولیه‌ای ایجاد می‌شود، سپس این مدل برای دسته‌بندی داده‌های جدید مورد استفاده قرار می‌گیرد، به این ترتیب با بکارگیری مدل بدست آمده تعلق داده‌های جدید به دسته معین قابل پیش‌بینی است. به عبارت دیگر دسته‌بندی شامل بررسی ویژگی‌های یک شیء جدید و تخصیص آن به یکی از مجموعه‌های از قبل تعیین شده است.

در زیر به بررسی معروفترین نوع از انواع این نوع شناسایی‌کننده‌ها پرداخته شده است.

3-1-3-1: شناسایی‌کننده بیز¹⁰

الگوریتم‌های یادگیری بیزی به طور صریح بر روی احتمالات فرض‌های مختلف کار می‌کنند. شناسایی‌کننده‌های بیزی شناسایی‌کننده‌های آماری هستند. آن‌ها اعضای کلاس را به صورت احتمالی پیشگویی می‌کنند. مثلاً میزان احتمال این که یک نمونه داده شده متعلق به یک کلاس خاص باشد. شناسایی‌کننده بیزی بر مبنای تئوری بیز است. مقایسه الگوریتم‌های کلاسه‌بندی نشان داده است که یک کلاسه‌بند بیزی ساده از نظر کارایی با کلاسه‌بندهای درخت تصمیم و سایر شناسایی‌کننده‌های دیگر قابل رقابت است و در برخی موارد بهتر از آن‌ها عمل می‌کند. همچنین شناسایی‌کننده‌های بیزی میزان دقت و سرعت بالایی را هنگامی که در حجم داده بزرگ به کار برده می‌شوند، ارائه می‌دهند.

فرض کنید بردار \mathbf{x} در اختیار باشد و هدف ما انتساب آن به یکی از دو دسته ω_1 و ω_2 باشد. از دیدگاه آماری ما به دنبال معیاری برای دسته‌بندی به محتمل‌ترین دسته هستیم. با این تعبیر معیار معقول برای انجام این دسته‌بندی، استفاده از $P(\omega_1 | \mathbf{x})$ و $P(\omega_2 | \mathbf{x})$ یا احتمالات پسین دو دسته ω_1 و ω_2 است. $P(\omega_1 | \mathbf{x})$ احتمال تعلق به دسته ω_1 است وقتی که بردار \mathbf{x} مشاهده شده باشد. بنابراین برای انتخاب محتمل‌ترین دسته کافی است این احتمال‌ها را با یکدیگر مقایسه کنیم:

¹ Bayes

$$\begin{aligned} (k|x) > (k|y) &\rightarrow \epsilon \\ (k|x) < (k|y) &\rightarrow \epsilon \end{aligned} \quad (1)$$

بنا بر قضیه بیز احتمال‌های پسین $(k|x)$ و $(k|y)$ را می‌توان به احتمال‌های پیشین¹¹ و تابع درست‌نمایی¹² دسته‌ها مرتبط ساخت:

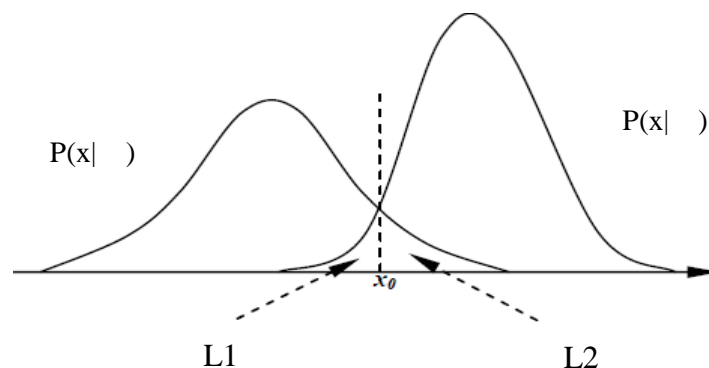
$$P(k|x) = \frac{P(k) P(x|k)}{P(x)} = 1,2 \quad (2)$$

که در این عبارت $P(k)$ احتمال وقوع دسته k یا احتمال پیشین دسته k بدون در نظر گرفتن بردار x است. $P(x|k)$ احتمال وقوع بردار x و $P(k|x)$ احتمال مشاهده بردار x در میان نمونه‌های مربوط به دسته k است. به عبارت دیگر، وقتی که می‌دانیم بردار مذکور به دسته k تعلق دارد احتمال $P(k|x)$ که به آن اصطلاحاً تابع درست‌نمایی دسته k نسبت به x گفته می‌شود نشان دهنده احتمال رویت بردار x است. حال اگر روابط (1) و (2) را با هم ترکیب کنیم، به معیار مقایسه زیر می‌رسیم:

$$L(x) \equiv \frac{P(k|x)}{P(y|x)}, \quad \begin{aligned} (k|x) &> \frac{P(k)}{P(y)} \rightarrow \\ (k|x) &< \frac{P(k)}{P(y)} \rightarrow \end{aligned} \quad (3)$$

که $P(k|x)$ را نسبت درست‌نمایی¹³ و نسبت $\frac{P(k)}{P(y)}$ را مرز مقایسه¹⁴ می‌نامند. در شکل 2-3 نمونه یک بعدی از این معیار مشاهده می‌شود.

¹ Priori probability
² Likelihood function
³ Likelihood ratio
⁴ Threshold



شکل 3-1: نمونه‌ای از 2 دسته توابع درست‌نمایی و مرز بین آنها

مطابق این شکل x مرز تساوی توابع درست‌نمایی $(\cdot | L1)$ و $(\cdot | L2)$ می‌باشد. بنا به معیار (3) اگر $x > x_0$ ، جزء دسته k محسوب می‌شود و اگر $x < x_0$ ، جزء دسته k به حساب می‌آید (در حالت کلی برداری از ویژگی‌هاست که در این مثال یک بعدی، استثنائاً از نماد بردار استفاده نشده است). در عمل راحت‌تر است که با منفی لگاریتم $(\cdot | L1)$ کار کنیم که تابع تفکیک¹⁵ خوانده می‌شود:

$$h(\cdot) \equiv - \ln \left(\frac{P(\cdot | L1)}{P(\cdot | L2)} \right) = \ln \left(\frac{P(\cdot | L2)}{P(\cdot | L1)} \right) \quad (4)$$

در ادامه خواهیم دید که این فرم از رابطه در صورت گوسی بودن توابع توزیع، ما را به یک تابع تفکیک خطی می‌رساند. طبیعتاً در مواردی که دو دسته مورد نظر هم احتمال باشند، $h(k) = \ln \left(\frac{P(\cdot | L2)}{P(\cdot | L1)} \right)$ می‌گردد و مرز مقایسه $h(\cdot)$ عدد صفر خواهد بود. بدین ترتیب $h(\cdot)$ نمونه‌ای از یک تابع تفکیک است. روابط (1) و (3) و (4) صورت‌های مختلف معیار بیز محسوب می‌شوند و تفکیک‌کننده‌ای که بر اساس آن طراحی می‌شود را تفکیک‌کننده بیز¹⁶ گویند.

اگر مجدداً به شکل 3-1 مراجعه کنیم، ملاحظه می‌شود که در انتساب به یکی از دو دسته k و k چهار حالت زیر متصور است:

¹ Discrimination function

² Bayes classifier

- (1) متعلق به دسته k باشد و ما هم آن را به k نسبت دهیم.
- (2) متعلق به دسته k باشد اما ما آن را به k نسبت دهیم.
- (3) متعلق به دسته k باشد اما ما آن را به k نسبت دهیم.
- (4) متعلق به دسته k باشد و ما هم آن را به k نسبت دهیم.

از این میان تنها حالات (1) و (4) مطلوب ما هستند که در آن‌ها دسته‌بندی درست¹⁷ انجام شده است. اما وقتی از معیارهای آماری همچون معیار بیز برای دسته‌بندی استفاده می‌کنیم بسته به شکل $(\cdot | \cdot)$ و $(\cdot | \cdot)$ و میزان همپوشانی آن‌ها، بروز حالات (2) و (3) اجتناب ناپذیر هستند.

در واقع برای محاسبه خطای ناشی از این دسته‌بندی نادرست، کفایست معیار خطایی بصورت زیر تعریف گردد:

$$= (\cdot | \cdot) + (\cdot | \cdot) \quad (5)$$

که در حالت یک بعدی L و L نواحی مشخص شده در شکل 3-1 می‌باشند. اگر چه ثابت می‌شود که تفکیک‌کننده بیز کمترین احتمال خطا را داراست، اما در عمل به علت دشواری محاسبه توابع درست‌نمایی $(\cdot | \cdot)$ و $(\cdot | \cdot)$ ، تشکیل $(\cdot | \cdot)$ و $(\cdot | \cdot)$ میسر نمی‌باشد و تنها در حالتی که $(\cdot | \cdot)$ ها دارای توزیع گوسی با متوسطهای M و ماتریس‌های کواریانس Σ باشند، $h(\cdot)$ به فرم زیر خواهد بود:

$$h(x) = -(x - M) \Sigma (x - M) - (x - M) \Sigma (x - M) + \ln \frac{|\Sigma|}{|\Sigma|} \quad (6)$$

$$\begin{aligned} h(x) < \ln \frac{(\cdot)}{(\cdot)} &\rightarrow x \in k \\ h(x) > \ln \frac{(\cdot)}{(\cdot)} &\rightarrow x \in k \end{aligned} \quad (7)$$

¹ Correct classification (CC)

برای یافتن درک بهتری نسبت به رابطه (6) حالت ساده‌ای را فرض می‌کنیم که $\Sigma = \Sigma = I$ که فرض تساوی Σ و Σ به معنی ناهمبسته فرض کردن ویژگی‌هاست و تساوی آن‌ها با ماتریس I ناشی از نرمالیزه کردن آن‌هاست. در این حالت می‌توان نشان داد که از ساده‌سازی رابطه (6) و (7) به رابطه زیر خواهیم رسید:

$$\begin{cases} ||x - M|| - ||x - M|| < 2 \ln \frac{P(k)}{P(k)} \rightarrow x \in k \\ ||x - M|| - ||x - M|| > 2 \ln \frac{P(k)}{P(k)} \rightarrow x \in k \end{cases} \quad (8)$$

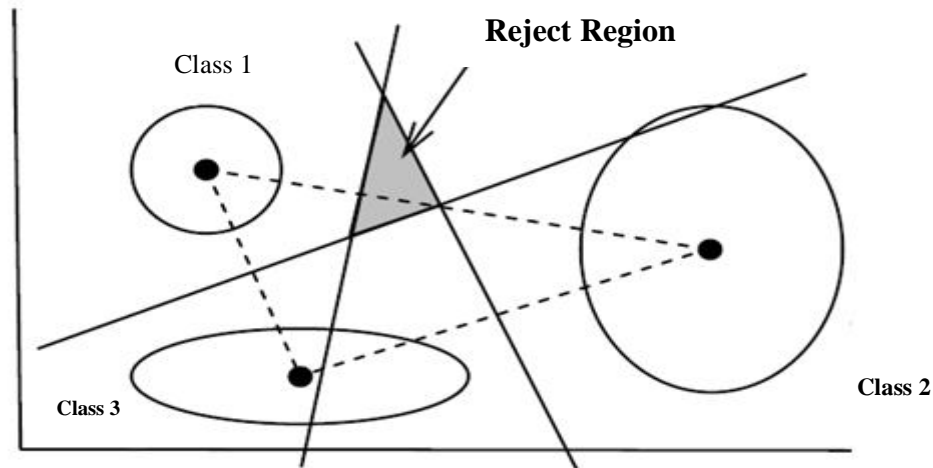
مطابق این رابطه فضای ویژگی به دو دسته تقسیم می‌شود. نقاطی که به M نزدیک‌ترند جزء دسته اول و نقاطی که به M نزدیک‌ترند جزء دسته دوم به حساب می‌آیند. در حالتی که $\Sigma \neq \Sigma$ تعبیر رابطه (6) قدری دشوارتر است، اما می‌توان گفت که باز معیاری از فاصله نقاط از M و M است که پراکندگی دسته‌ها نیز در آن دخیل گشته است.

برای تعمیم روش دسته‌بندی تفکیک‌کننده بیز به بیش از دو دسته، از ایده تفکیک‌کننده‌های تکه‌ای خطی استفاده می‌کنیم. برای این منظور فرض کنید که بخواهیم بردار ویژگی را در یکی از N دسته ($i = 1, 2, \dots, N$) قرار دهیم. در این صورت مطابق رابطه (6) و یا هر معیار فاصله دیگر $h_i(x)$ را به ازای کلیه مقادیر i و تشکیل می‌دهیم و در صورت وجود یک که به ازای آن شرط $h_i(x) < 0$ ($i \neq k, i = 1, 2, \dots, N$) برقرار باشد، را در دسته ام جای می‌دهیم. به عبارت دیگر:

$$h_1(x) < 0, h_2(x) < 0, \dots, h_N(x) < 0 \Rightarrow \in k \quad (9)$$

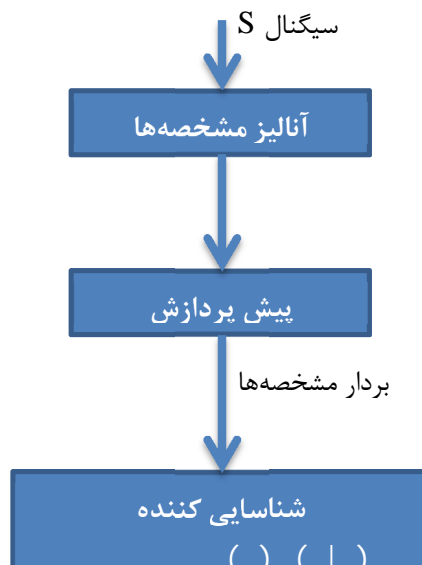
طبیعتاً ممکن است برای برخی از h_i نتوانیم آن را مطابق این رابطه در هیچ یک از دسته‌ها جای دهیم. بدین ترتیب مطابق شکل 2-3 نواحی ایجاد می‌شوند که با استفاده از معیار فوق به هیچ دسته‌ای تعلق ندارند. در مبحث تفکیک‌کننده‌های تکه‌ای خطی به این نواحی، نواحی طرد¹ طرد¹ گفته می‌شود.

¹ Reject region



شکل 3-3: دسته‌بندی با بیش از 2 دسته و پدید آمدن ناحیه طرد

با توجه به موارد مطرح شده برای شناسایی‌کننده بیز، می‌توان نمودار مربوط به شناسایی‌کننده‌های آماری را برای این نوع طبقه‌بندی‌کننده به صورت شکل 3-4 نمایش داد. به منظور آموزش و تشکیل یک شناسایی‌کننده بیز نیاز است پارامترهای مربوط به دو تابع $()$ و $(|)$ محاسبه شوند. از آنجا که $()$ مربوط به کلاس بوده و مجموع احتمالات اولیه برای کلاسهای مختلف برابر با یک است، نیاز به برآورد 1- پارامتر برای محاسبه احتمالات اولیه مربوط به دسته‌های مختلف داریم. علاوه بر این، برای هر تابع احتمالاتی $(|)$ نیاز به محاسبه یک مقدار میانگین و مقدار انحراف معیار هر ویژگی داریم. حال اگر بردار ویژگی دارای عنصر باشد، در این بخش نیز نیاز به برآورد $2 * *$ پارامتر مشهود است.

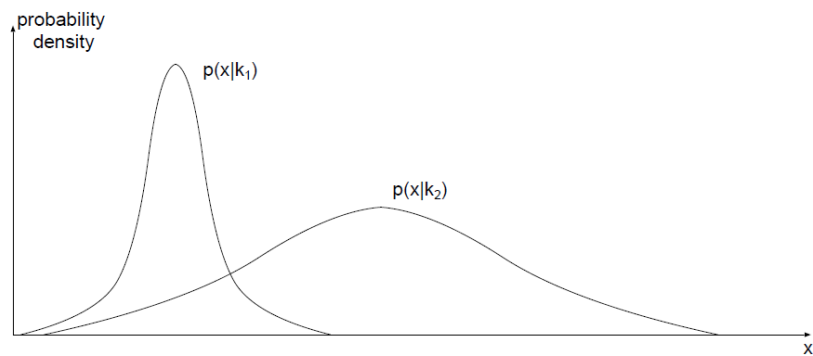


شکل 3-4: شناسایی کننده بیز

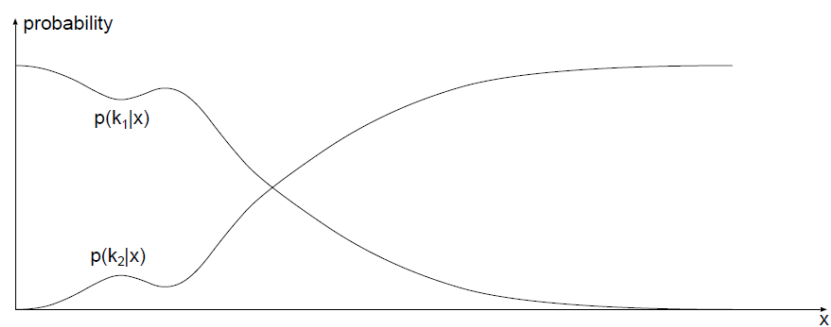
شناسایی کننده بیز را می توان به صورت زیر نمایش داد:

$$\text{---} \quad () = \operatorname{argmax} \quad () \quad (|)$$

با در نظر گرفتن یک مساله شناسایی برای دو کلاس و در حالی که میدانیم احتمال پیشین برای این دو کلاس به ترتیب برابر با $2/3$ و $1/3$ است، در صورتی که تابع درست‌نمایی مربوط به این دو دسته مطابق شکل عععع نمایش داده شود، تابع احتمالاتی پسین آن به صورت شکل عععع خواهد بود.

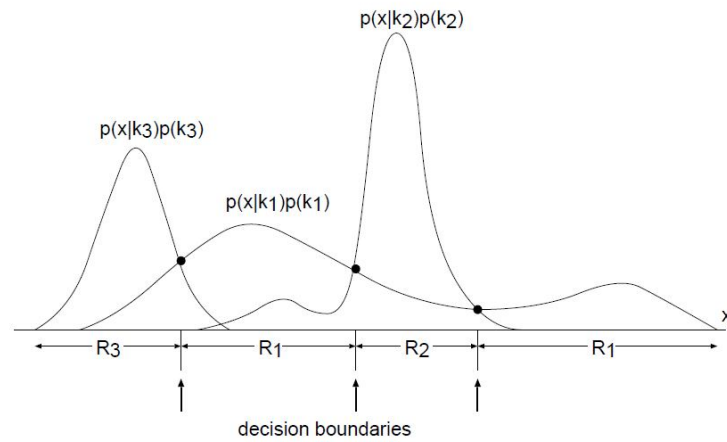


شکل 3-5: تابع راست نمایی برای دو کلاس



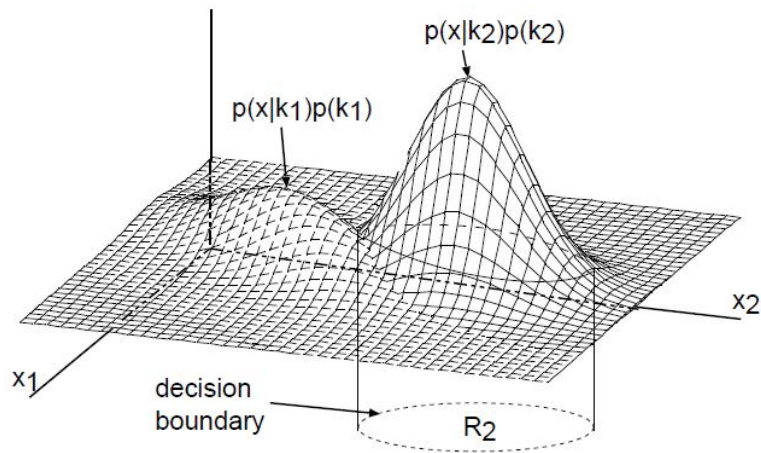
شکل 3-6: تابع احتمالاتی پسین

در صفحات قبل به مرز بین دو کلاس که با توجه به عملکرد دسته‌بندی کننده بیز به دست می‌آمد اشاره شده است. در شکل زیر شما می‌توانید نقاط مرزی مربوط به سه کلاس را مشاهده نمایید.



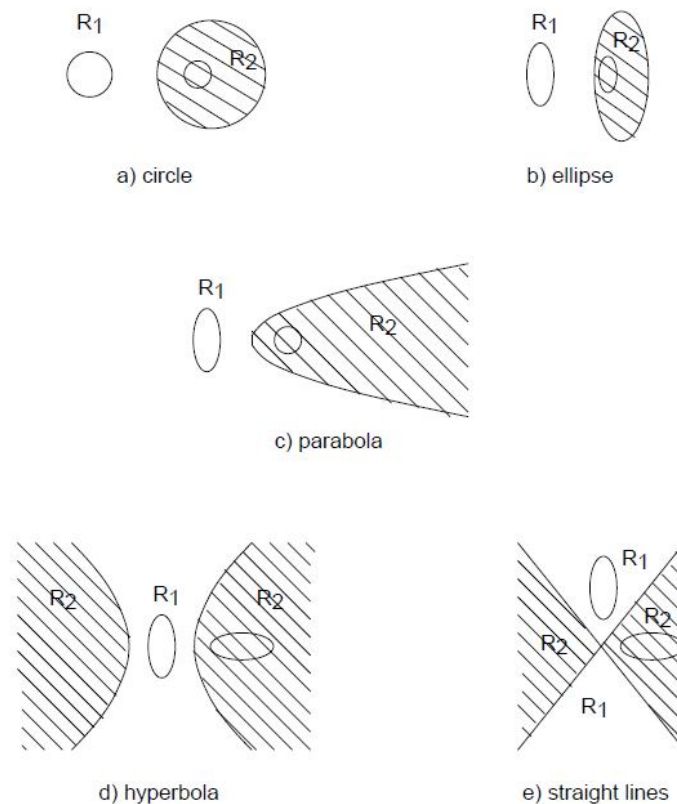
شکل 3-7: حدود مرزی بین سه کلاس در فضای یک بعدی

در صورتی که بردار ویژگیها را به صورت دو بعدی فرض کنیم، حدود مرزی بین دو کلاس را می‌توان به صورت شکل عععع نمایش داد.



شکل 3-8: حد مرزی بین دو کلاس در فضای دو بعدی

در فضای دو بعدی با داشتن دو کلاس که پراکندگی داده‌های هر کلاس به صورت دایره یا بیضی باشد، حدود مرزی می‌تواند به صورت یکی از شکل‌های زیر باشد.



شکل 3-9: حدود مرزی بین دو کلاس در فضای دو بعدی

توجه داشته باشید که کشیدگی شکل بیضی استفاده شده، برای نمایش پراکندگی نمونه‌های یک کلاس است. در نتیجه با در نظر گرفتن نقطه میانگین هر کلاس و پراکندگی مربوط به نمونه‌های آن، حدود مرزی مشخص شده است. به عنوان مثال در بخش a دو کلاس داریم که کلاس اول در مقایسه با کلاس دوم با توجه به دایره نمایش داده شده با شعاع بزرگتر برای آن، پراکندگی داده‌های بیشتری دارد. در نتیجه حدود مرزی بین این دو کلاس به گونه‌ای ترسیم شده است که نمونه‌هایی که فاصله زیادی تا نقطه میانگین کلاس دوم دارند، حتی اگر فاصله

آنها تا نقطه میانگین کلاس اول بیشتر باشد، باز هم جزء کلاس اول در نظر گرفته می شود زیرا کلاس اول در مقایسه با کلاس دوم پراکندگی و فاصله بیشتری را تا مرکز تحمل می کند.

3-1-1: شناسایی کننده مبتنی بر فاصله

پس از آشنایی با شناسایی کننده بیز، به معرفی دسته ای از شناسایی کننده ها به نام شناسایی کننده های مبتنی بر فاصله می پردازیم. اساس این نوع از شناسایی کننده ها نیز همان فرمول بیز است که تکیه بیشتری بر فاصله نمونه آزمایشی تا بردار میانگین یک کلاس دارد. تابع توزیع گاوسی را با ماتری کواریانس Σ در نظر بگیرید. برای تابع تشخیص $(\cdot, \cdot) = \log((\cdot | \cdot))$ از مقادیری که در یک کلاس ثابت بوده و بدون تغییر هستند صرف نظر نموده و تابع را به صورت زیر بسط می دهیم:

$$\begin{aligned} (\cdot, \cdot) &= -\frac{1}{2}(\cdot - \cdot)(\cdot - \cdot) + \log(\cdot) \\ &= -\frac{1}{2}(\cdot - \cdot) + \log(\cdot) \end{aligned}$$

در فرمول فوق، $\Sigma(\cdot - \cdot)$ بیانگر فاصله اقلیدسی بردار ویژگی استخراج شده از نمونه تا بردار میانگین یک کلاس است که با مقدار اندازه آن اصلاح می شود. در صورتی که بخواهیم شناسایی کننده را در سیستم های سریع به کار بگیریم، می توان از میزان احتمال پیشین هر کلاس و پراکندگی داده ها در هر کلاس صرف نظر نموده و ضمن تعریف فاصله هر نمونه که با بردار ویژگی نمایش داده می شود تا بردار میانگین با $(\cdot, \cdot) \geq 0$ ، تابع تشخیص را به صورت زیر خلاصه نماییم:

$$\begin{aligned} (\cdot, \cdot) &= -(\cdot, \cdot) \\ \text{در نتیجه فرمول شناسایی کننده به صورت زیر نمایش داده خواهد شد:} \\ (\cdot) &= \operatorname{argmin} \cdot, \end{aligned}$$

تابع فاصله را می توان به صورت یک تابع با نرم به صورت زیر تعریف کنیم:

?

$$(x, y) := \|x - y\| = |x_1 - y_1|$$

می‌توان با در نظر گرفتن $\|x - y\| = 1$ تابع فاصله به نام فاصله شطرنجی یا بلوکی را به صورت زیر تعریف نمود:

$$(x, y) = |x_1 - y_1|$$

اگر $\|x - y\| = 2$ در نظر گرفته شود فاصله اقلیدسی به عنوان پرکاربردترین تابع فاصله به صورت زیر به دست می‌آید:

$$(x, y) = |x_1 - y_1|$$

و با در نظر گرفتن $\|x - y\| = \infty$ تابع فاصله به نام فاصله ماکزیمم تعریفی اینچنین خواهد داشت:

$$(x, y) = \max |x_i - y_i|$$

با توجه به نگرش جدیدی که به شناسایی‌کننده بیز به عنوان یک شناسایی‌کننده مبتنی بر فاصله مطرح شد، دریافتیم که از توابع فاصله دیگر نیز می‌توان طبق فرمولهای فوق در شناسایی‌کننده‌ها استفاده کنیم. در بخشی از مطلب ارائه شده، بیان شد که می‌توان به منظور طراحی و پیاده‌سازی یک سیستم شناسایی‌کننده سریع، از مقدار احتمال پیشین و پراکندگی هر کلاس صرف نظر نمود. با این حال حتی بدون صرف نظر کردن از این اطلاعات نیز در فرمول ارائه شده برای شناسایی‌کننده، توابع فاصله دیگر را می‌توان جایگزین تابع فاصله اقلیدسی نمود. استفاده از توابع فاصله هدف‌دار در برخی کاربردهای خاص می‌تواند شناسایی‌کننده را نسبت به برخی تغییرات مقاوم نموده و شاهد کاهش نرخ خطا و افزایش دقت در تشخیص درست

شناسایی کننده باشیم. به عنوان مثال فاصله تانژانت¹ و فاصله مبتنی بر اعوجاج محلی تصاویر² از جمله توابعی هستند که در سیستم‌های شناسایی تصویر بسیار پرکاربردند.

3-3-2: شناسایی کننده بدون مدل

به طور کلی در این نوع از دسته‌بندی کننده‌ها، تعدادی دسته با اعضای مشخص وجود دارد که بعضی ویژگی‌های این اعضا نیز مشخص است. در دسته‌بندی‌ها به طور معمول به دنبال یافتن الگوریتمی هستیم که بر اساس اطلاعات اولیه از دسته‌های مختلف، در صورت ورود یک عضو جدید تعلق آن به یکی از دسته‌های موجود شناسایی شود. در این پروسه تک تک نمونه‌های آموزشی معنی پیدا می‌کنند و دسته‌بندی داده‌های جدید بر اساس همین نمونه‌های آموزشی انجام می‌پذیرد و هیچ مدلی برای بررسی داده‌های ورودی وجود نخواهد داشت.

در زیر به بررسی دو نوع از پر کاربردترین الگوریتم‌های مبتنی بر این مدل پرداخته شده است.

3-3-2-1: شناسایی کننده نزدیکترین همسایه NN³

فرض می‌کنیم $D = \{x_1, \dots, x_n\}$ مجموعه‌ای از n الگوی ورودی باشد و X' متعلق به مجموعه D ، نزدیکترین الگوی ورودی به نقطه تست X باشد. قانون نزدیکترین همسایه برای دسته‌بندی X آن را در کلاسی مشابه با کلاس X' قرار می‌دهد. در واقع ورودی X به کلاسی تعلق می‌گیرد که کمترین فاصله را نسبت به X' داشته باشد. قانون نزدیکترین همسایه یک روال شبه بهینه⁴ است. یعنی نرخ خطای آن معمولاً بیشتر از حداقل نرخ خطای کمینه، (یعنی نرخ خطای الگوریتم بیز) است. ولی ثابت می‌شود که در صورت استفاده از تعداد نامحدودی از الگوهای ورودی، نرخ خطا در بدترین حالت بیشتر از 2 برابر نرخ خطای الگوریتم بیز نخواهد شد.

¹ Tangent distance

² Image distortion model distance

³ Nearest neighbor

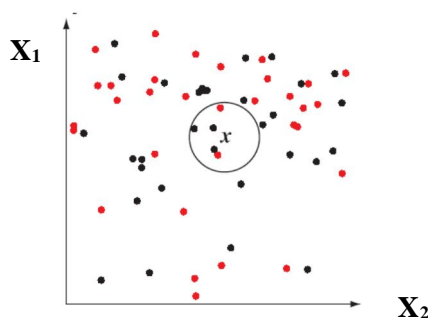
⁴ Sub-optimal

مشکلی که در رابطه با شناسایی کننده نزدیکترین همسایه پیش می‌آید این است که گاهی ممکن است نمونه تست نزدیک به یک نمونه آموزشی باشد که به صورت تصادفی در بین نمونه‌های متعلق به کلاس دیگر قرار گرفته است. در چنین حالتی به علت نزدیکی نمونه تست به یک نمونه داده‌ای پرت، شناسایی کننده علی‌رغم نزدیکی به نمونه‌های زیاد یک کلاس به کلاس دیگر تعلق می‌گیرد. روش K نزدیکترین همسایه پاسخی است برای رفع این مشکل که در ادامه به آن می‌پردازیم.

2-2-3-3: شناسایی کننده k نزدیکترین همسایه⁵KNN

الگوریتم K نزدیکترین همسایه، یک الگوریتم تعلیم با سرپرستی است. در حالت کلی از این الگوریتم به دو منظور استفاده می‌شود:

برای تخمین تابع چگالی توزیع داده‌های تعلیم و برای طبقه‌بندی داده‌های تست بر اساس الگوهای تعلیم. الگوریتم K نزدیکترین همسایه گسترشی از الگوریتم نزدیکترین همسایه است و همانطور که واضح است این الگوریتم X را در دسته‌ای طبقه‌بندی می‌کند که بیشترین تکرار را در بین K نزدیکترین همسایه X دارد. به عنوان مثال در شکل زیر X توسط این الگوریتم در کلاس سیاه قرار می‌گیرد. زیرا به ازای در نظر گرفتن 5 همسایه که در کمترین فاصله با نمونه تست قرار دارند، سه همسایه متعلق به کلاس سیاه هستند.



شکل 3-10: بررسی به وسیله الگوریتم K نزدیکترین همسایه

⁵ K nearest neighbor

KNN ساده‌ترین روش و متداول‌ترین روش بر پایه یادگیری نمونه یا همان بدون مدل است. در این روش فرض می‌شود که تمام نمونه‌ها نقاطی در فضای n بعدی حقیقی هستند و همسایه‌ها بر مبنای فواصل اقلیدسی تعیین می‌شوند. (k تعداد همسایه‌های در نظر گرفته شده است).

الگوریتم KNN برای یک تابع هدف گسسته به صورت زیر تعریف می‌شود:

$$: \rightarrow, h \quad h \quad \{ \dots, \}$$

الگوریتم یادگیری:

الگوریتم یادگیری یا آموزش استفاده شده برای KNN به صورت ساده با افزودن هر داده آموزشی به داده‌های قبلی صورت می‌گیرد. به عبارت دیگر، هر مثال آموزشی $f(x)$ را باید به لیست نمونه‌های آموزشی⁶ اضافه کنیم.

الگوریتم دسته‌بندی:

برای نمونه مورد بررسی x ، نزدیک‌ترین نمونه‌هایی از مثال‌های آموزشی به آن با x_1, \dots, x_n نشان داده می‌شوند و سپس مقدار زیر محاسبه می‌شود.

$$(x) \leftarrow \operatorname{argmax} \sum_h \delta(v, f(x)) \quad \begin{aligned} \delta(a, b) &= 1 \text{ if } a = b \\ \delta(a, b) &= 0 \text{ otherwise} \end{aligned}$$

به عنوان مثال:

اگر $1 =$ انتخاب شود، الگوریتم 1-NN مقدار نزدیک‌ترین نمونه به x را انتخاب خواهد نمود. برای مقادیر بزرگتر K متداول‌ترین مقدار بین K همسایه نزدیک انتخاب خواهد شد.

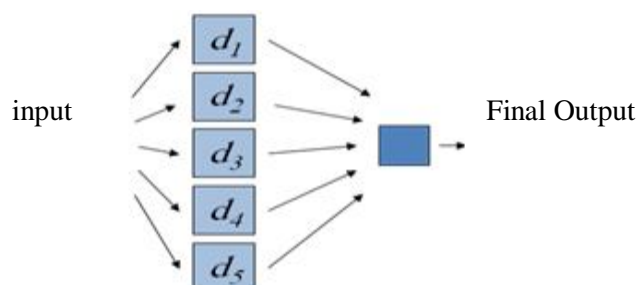
الگوریتم KNN را می‌توان به سادگی برای تابع هدف پیوسته نیز استفاده نمود. در این حالت به جای انتخاب متداول‌ترین مقدار موجود در همسایگی، مقدار میانگین K مثال همسایه محاسبه می‌شود. در نتیجه در خط آخر الگوریتم از رابطه زیر استفاده می‌شود:

¹ Training examples

$$(x) \leftarrow \frac{\sum f(x)}{k}$$

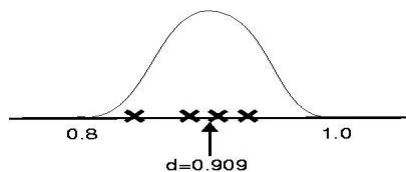
3-4: ترکیب شناسایی کننده‌ها

روش‌های مختلفی برای ترکیب نتایج دسته‌بندی کننده‌ها وجود دارد. متداول‌ترین روش‌ها میانگین‌گیری و یا استفاده از رای اکثریت هستند.



شکل 3-11: ترکیب شناسایی کننده‌ها

انگیزه اصلی ترکیب دسته‌بندی‌ها این است که ما هنگام طراحی یک سیستم یادگیر انتخاب‌های فراوانی داریم، مثل نحوه نمایش، پارامترهای یادگیر، داده‌های آموزشی و غیره. این تنوع باعث می‌شود که نوعی از واریانس در عملکرد سیستم وجود داشته باشد. در نتیجه اگر سیستم‌های مختلفی داشته و از نتایج آن‌ها استفاده شود، این امکان وجود دارد که توزیع خطا حول آن به حداقل برسد.



شکل 3-12: کاهش توزیع خطا با ترکیب دسته‌بندی‌ها

اگر از نتیجه چند دسته‌بندی‌کننده به صورت $f_{com} = vote(f_i, f_j, f_k, f_l, f_m)$ استفاده شود به شرط مستقل بودن توابع با استفاده از روابط توزیع دو جمله‌ای داریم:

$$P(\text{error}) = \sum_{k=0}^n p^k (1-p)^{n-k}$$

برای این‌که بتوان نتیجه مناسبی از ترکیب دسته‌بندی‌کننده‌ها گرفت، این دسته‌بندی‌کننده‌ها باید هر یک به تنهایی در حد قابل قبولی دقیق باشند. البته نیازی به بسیار دقیق بودن آنها نیست زیرا هر کدام می‌توانند به عنوان مکمل دیگری عمل کنند. به عبارت دیگر همگی نباید مشابه هم بوده و نتیجه یکسانی تولید کنند زیرا در این صورت دلیلی برای ترکیب آنها باقی نمی‌ماند. دو روش برای ترکیب دسته‌بندی‌کننده‌ها وجود دارد:

• ساختارهای آماری⁷

در این روش پاسخ چندین سیستم خبره بدون در نظر گرفتن سیگنال ورودی با هم ترکیب می‌شوند.

• ساختارهای پویا⁸

در این روش سیگنال ورودی در انتخاب مکانیسم ترکیب سیستم‌های خبره تاثیر می‌گذارد. خروجی خبره‌ها توسط یک شبکه یا چندین شبکه به صورت غیرخطی با هم ترکیب می‌شوند. ترکیب دسته‌بندی‌کننده‌ها باعث می‌شود که خطای حاصل از ترکیب، از خطای میانگین کمتر شود.

$$(f - t) = \frac{1}{m} (f - t) - \frac{1}{m} (f - f)$$

$$f = \frac{1}{m} f$$

⁷ Statistic structures

⁸ Daynamic structures

که f خروجی سیستم شناسایی کننده نام و f خروجی سیستم شناسایی ترکیبی در مقابل t به عنوان خروجی مورد نظر سنجیده می‌شوند. این فصل را که با معرفی شناسایی‌کننده‌ها آغاز شد با آشنایی با روشهای استفاده همزمان و ترکیبی شناسایی‌کننده‌ها پایان می‌بریم. در ادامه فصلی با عنوان پایگاه داده خواهیم داشت که با نحوه عملکرد شناسایی‌کننده‌ها ارتباط مستقیمی دارد. از آنجا که شناسایی‌کننده‌های آماری در هر سه مرحله آموزش، توسعه و آزمون وابسته به اطلاعات و نمونه‌های آماری هستند، معرفی شرایط و مشخصات پایگاه داده استاندارد از اهمیت بالایی برخوردار است.

مراجع

1. K. Fukunaga, "Introduction to statistical pattern recognition", 2nd edition, Academic Press, 1990
2. . S. Theodoridis, K. Koutroumbas, "Pattern recognition", Academic Press, 1999.
3. T. Y. Young, T. K. Calvert, "Classification, estimation and pattern recognition", Elsevier, 1974.
4. Y. Yang and X. Liu. A re-examination of text categorization methods. In proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval
5. J. He, A. Tan and C. Tan. Comparative study on chinese text categorization methods. On the PRICAI 2000 Workshop on text and web mining, Melbourne, Australia, 2000.
6. Y. Yang, "An evaluation of statistical approaches to text categorization", Journal of information retrieval, 1999.
7. H. Ney, "Pattern recognition and neural networks", Lecture notes, Informatik 6, RWTH-Aachen, Germany, 2013.