فصل چهارم

آموزش و ارزیابی

4-1: مقدمه

همانگونه که در فصل قبل با نحوه عملکرد شناسایی کننده‌ها آشنا شدید، گفتیم که آموزش و ارزیابی دو بخش مهم و اساسی یک شناسایی‌کننده آماری را تشکیل می‌دهند و در کنار آنها مرحله اعتبارسنجی که مشابه ارزیابی یا تست عمل می‌کند نیز جهت تعیین برخی پارامترها مورد استفاده قرار می‌گیرد.

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 کاملا 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 می‌بینند و در نهایت مطلوبیت کارکرد الگوریتم با داده‌های آزمون محاسبه می‌شود.

در این فصل مراحل آموزش (یادگیری) و ارزیابی (تست) را با در نظر گرفتن جزئیات و نکات کاربردی هر یک معرفی می‌نماییم. در پایان این فصل انتظار داریم نگرش خواننده نسبت به سیستم‌های شناسایی‌کننده تغییر یافته و از عمق و دقت بیشتری برخوردار گردد. این نگرش در طراحی و پیاده‌سازی سیستمهای کاربردی شناسایی الگو بسیار مفید خواهد بود.

4-2: آموزش و یادگیری

آموزش یا یادگیری در سیستمهای شناسایی الگو که بر مبنای نمونه کار می‌کنند و به عنوان شناسایی‌کننده‌های بدون مدل شناخته می‌شوند، به صورت بسیار ساده و با افزودن هر نمونه آموزشی جدید به مجموعه داده‌های موجود انجام می‌شود. اما در سیستم‌های شناسایی‌کننده بر مبنای مدل، آموزش یا یادگیری سیستم به معنی تخمین پارامترهای مدلی است که برای سیستم شناسایی‌کننده در نظر گرفته شده است. به عنوان مثال در شناسایی‌کننده بیز که به صورت زیر تعریف می‌شود:

$$\text{——} \quad (\ ) = \text{argmax} \quad (\ )\ (\ |\ )$$

منظور از آموزش سیستم، تعیین توابع احتمالاتی پیشین و درستنمایی است که به ترتیب با ( ) و (|) نمایش داده می‌شوند. اگر بردار ویژگیهای دارای عضو بوده و کلاسهای , … . , , , را خروجی سیستم شناسایی‌کننده را تشکیل دهند، تعداد پارامترهایی که برای تعیین توابع پیشین و درستنمایی باید تخمین زده شوند، شامل موارد زیر است:

الف- تابع احتمالاتی پیشین: تعداد 1 − مقدار برای ( ) , … ,( ) ,( ) . توجه داشته باشید از آنجا که مجموع مقدار توابع پیشین برای کلاس برابر با یک است، فقط تعداد 1 − تعیین مقدار شوند کافی است. مقدار توابع احتمالاتی پیشین با توجه به فرکانس نمونه‌های موجود در داده‌های آموزشی تعیین می‌شوند.

ب- تابع درستنمایی: تعداد * * 2 پارامتر برای تخمین دو پارامتر میانگین و انحراف معیار که تابع توزیع نرمال را برای عضو بردار ویژگیها به دست می‌آیند. البته توجه داشته باشید که تابع درستنمایی برای هر یک از کلاس در نظر گرفته شده برای سیستم باید جداگانه محاسبه گردد.

همانطور که اشاره شد، مقادیر مربوط به تابع احتمالاتی پیشین برای هر یک از کلاسها با شمارش در داده‌های آموزشی انجام می‌پذیرد. اما به منظور تخمین پارامترهای مدل به کار گرفته شده برای تابع درستنمایی، در اینجا دو روش یادگیری با استفاده از ممانها و روش بیشترین شباهت را مورد بررسی بیشتر قرار می‌دهیم

4-2-1: روش یادگیری ممانها

اولین روشی را که برای آموزش یک سیستم شناسایی‌کننده معرفی می‌کنیم روشی است که بر اساس تعریف ممانهای هندسی انجام شده است. فرض کنید نمونه داده آموزشی برای کلاس به صورت زیر تعریف شده باشند:

$$, \dots . , \quad , \dots . , \quad \in \mathbb{R}$$

مقدار مورد انتظار یا امید ریاضی تابع $\mathbb{R} \rightarrow \mathbb{R}$: به صورت زیر تعریف خواهد شد:

$$= \quad ( ) \ (\mid)$$

$$\mathbb{R}$$

$$\begin{array}{ll} f(x) = x^{\alpha} & \text{for } x \in \mathbb{R}, \alpha \in \mathbb{N} > 0 \\ f(x) = (x - \mu)^{\alpha} & \text{where } \mu = E\{x\} \end{array}$$

$$\mu_k = \int_{\mathbb{R}^D} dx \; x \cdot p(x|k)$$

$$\hat{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N_k} x_{nk}$$

$$\hat{\Sigma}_{cd}^{(k)} = \frac{1}{N_k} \sum_{n=1}^{N_k} (x_{nk,c} - \hat{\mu}_{k,c}) \cdot (x_{nk,d} - \hat{\mu}_{k,d}).$$

$$\hat{\sigma}_{kd}^2 = \frac{1}{N_k} \sum_{n=1}^{N_k} (x_{nk,d} - \hat{\mu}_{k,d})^2.$$

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. مسلما 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.

4-2-2: روش یادگیری بیشترین شباهت

یک مدل معمولا 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 دقیقا همان داده‌های آموزشی نخواهند بود. هر تخمینی درباره کارایی بر اساس آن داده‌ها خوش‌بینانه خواهد بود

$$p(x|k, \vartheta_k) \quad .$$

$$x_{1k}, \ldots, x_{nk}, \ldots, x_{N_k k} \qquad \in \mathbb{R}^D$$

$$\vartheta_k \longrightarrow \prod_{n=1}^{N_k} p(x_{nk}|k, \vartheta_k)$$

$$\vartheta_k \longrightarrow \sum_{n=1}^{N_k} \log p(x_{nk}|k, \vartheta_k).$$

$$\hat{\vartheta}_k \;:=\; \underset{\vartheta_k}{\operatorname{argmax}} \left\{ \prod_{n=1}^{N_k} p(x_{nk}|k,\vartheta_k) \right\}.$$

$$\sum_{n=1}^{N_k} \frac{\partial}{\partial \vartheta_k} \log p(x_{nk}|k,\vartheta_k) \;\stackrel{!}{=}\; 0$$

$$\sum_{n=1}^{N_k} \frac{1}{p(x_{nk}|k,\vartheta_k)} \cdot \frac{\partial p(x_{nk}|k,\vartheta_k)}{\partial \vartheta_k} \;\stackrel{!}{=}\; 0$$

$$p(x|k,\mu_k,\sigma_k^2) \;=\; \frac{1}{\prod_{d=1}^{D} \sqrt{2\pi\sigma_{kd}^2}} \exp\left[-\frac{1}{2} \sum_{d=1}^{D} \left(\frac{x_d - \mu_{kd}}{\sigma_{kd}} \right)^2 \right]$$

Training data: $x_{1k}, \ldots, x_{nk}, \ldots, x_{N_k k}$

Log-Likelihood for class k:

$$\sum_{n=1}^{N_k} \log p(x_{nk}|k,\mu_k,\sigma_k^2) \;=\; -\frac{1}{2} \sum_{n=1}^{N_k} \left[\sum_{d=1}^{D} \left(\frac{x_{nk,d} - \mu_{kd}}{\sigma_{kd}} \right)^2 + \sum_{d=1}^{D} \log\left(2\pi\sigma_{kd}^2\right) \right]$$

$$\frac{\partial}{\partial \mu_{kd}} = \sum_{n=1}^{N_k} \left(\frac{x_{nk,d} - \mu_{kd}}{\sigma_{kd}^2} \right) \stackrel{!}{=} 0$$

$$\hat{\mu}_{kd} = \frac{1}{N_k} \sum_{n=1}^{N_k} x_{nk,d}$$

$$\frac{\partial}{\partial \sigma_{kd}^2} = \frac{1}{2\sigma_{kd}^4} \sum_{n=1}^{N_k} (x_{nk,d} - \mu_{kd})^2 - \sum_{n=1}^{N_k} \frac{1}{2\sigma_{kd}^2} \stackrel{!}{=} 0$$

$$\hat{\sigma}_{kd}^2 = \frac{1}{N_k} \sum_{n=1}^{N_k} (x_{nk,d} - \hat{\mu}_{kd})^2$$

$$\begin{aligned} \hat{\mu}_{kd} &= \frac{1}{N_k} \sum_{n=1}^{N_k} x_{nk,d} \\ \hat{\Sigma}_{kc,d} &= \frac{1}{N_k} \sum_{n=1}^{N_k} (x_{nk,d} - \hat{\mu}_{kd})(x_{nk,c} - \hat{\mu}_{kc}) \end{aligned}$$

$p(x|k, \mu_k, \Sigma)$: Gaussian distribution with μ_k and Σ

$$\Sigma \quad \rightarrow \quad \prod_{K=1}^{K} \prod_{n=1}^{N_k} p(x_{nk}|k, \mu_k, \Sigma)$$

$$\begin{aligned} \hat{\mu}_{kd} &= \frac{1}{N_k} \sum_{n=1}^{N_k} x_{nk,d} \\ \hat{\Sigma}_{cd} &= \frac{1}{N} \sum_{k=1}^{K} \left[\sum_{n=1}^{N_k} (x_{nk,d} - \hat{\mu}_{kd})(x_{nk,c} - \hat{\mu}_{kc}) \right] \\ & \text{where } N = \sum_{k=1}^{K} N_k \end{aligned}$$

- Laplace distribution:

$$p(x|\mu, v) = \frac{1}{2v_k} e^{-\frac{|x-\mu|}{v}},$$

- Binomial distribution:

$$p_N(n|\vartheta) = \left(\begin{array}{c} \mathrm{N} \\ \mathrm{n} \end{array} \right) \vartheta^n (1-\vartheta)^{N-n},$$

- Multinomial distribution:

$$\text{if} \quad \sum_{i=1}^{k} \vartheta_i = 1 \quad \text{and} \quad \sum_{i=1}^{k} n_i = N$$

$$p_N(n_1, \ldots, n_k | \vartheta_1, \ldots, \vartheta_k) = \frac{N!}{n_1! \cdot \ldots \cdot n_k!} \cdot \vartheta_1^{n_1} \cdot \ldots \cdot \vartheta_k^{n_k},$$

- Poisson distribution:

$$p(n|\lambda) = \frac{\lambda^n}{n!} e^{-\lambda}.$$

و ممکن است مایوس‌کننده باشد. به همین دلیل مجموعه‌های آموزش و تست، باید از دو مجموعه داده مستقل انتخاب شوند. مجموعه تست[1] مجموعه‌ای از داده هاست 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.

3-2-4: نکات کاربردی

poolingهمانگونه که تاکنون مشاهده کردید، تاکیدی 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

[1] Test set or evaluation set

$$N_k \geq (10, \ldots, 100) \cdot D$$

- Smoothing $\hat{\Sigma}_k$ using $\hat{\Sigma}$:

$$\tilde{\Sigma}_k := (1 - \lambda_k)\hat{\Sigma}_k + \lambda_k \hat{\Sigma} ,$$

انجام می‌دهیم نتیجه بسیار ضعیفی را شاهد هستیم که این می‌تواند به این علت باشد که یک یا چند پارامتر تعیین کننده 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 اعتبارسنجی پارامترها را تنظیم می‌کنیم. به این مجموعه از داده‌ها است که برای سازگارکردن پارامترهای طبقه‌بندی‌کننده استفاده می‌شوند مجموعه اعتبارسنجی یا توسعه[2] می‌گوییم.

به عبارت دیگر، اگر مدل برای داده‌های آموزشی خیلی مناسب‌تر از داده‌های تست ارزیابی شود، احتمالا علت آن این است که روی هم افتادگی[3] 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. مثلا 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.

---

[2] Development
[3] Over-fitting

3-4: ارزیابی یا تست

در این فصل به بررسی دو نوع از شناسایی‌کننده‌های بر مبنای مدل[4] و بدون مدل[5] می‌پردازیم. شناسایی‌کننده‌های مبتنی بر مدل با فرض قرار دادن یک مدل برای سیستم شناسایی کننده 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.

1-3-4: نحوه محاسبه نرخ خطا

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 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.

2-3-4: وابستگی به اندازه بردار ویژگیها

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. مثلا میزان احتمال این‌که یک نمونه داده شده متعلق به یک کلاس

---

[2] Model based
[3] Model free

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 در اختیار باشد و هدف ما انتساب آن به یکی از دو دسته و باشد. از دیدگاه آماری ما به‌دنبال معیاری برای دسته‌بندی به محتمل‌ترین دسته هستیم. با این تعبیر معیار معقول برای انجام این دسته‌بندی، استفاده از (|) و (|) یا احتمالات پسین دو دسته k و k است. (|) احتمال تعلق به دسته k است وقتی که بردار مشاهده شده باشد. بنابراین برای انتخاب محتمل‌ترین دسته کافی است این احتمال‌ها را با یکدیگر مقایسه کنیم:

$$\begin{array}{l} (\ |\ ) > (\ |\ ) \rightarrow \in \\ (\ |\ ) < (\ |\ ) \rightarrow \in \end{array} \qquad (1)$$

بنا بر قضیه بیز احتمال‌های پسین (k |x) و(x| k) را می‌توان به احتمال‌های پیشین[1] و تابع درست‌نمایی[2] دسته‌ها مرتبط ساخت:

$$P(\ |\ ) = \frac{(\ )\ (\ |\ )}{(\ )} \qquad = 1,2 \qquad (2)$$

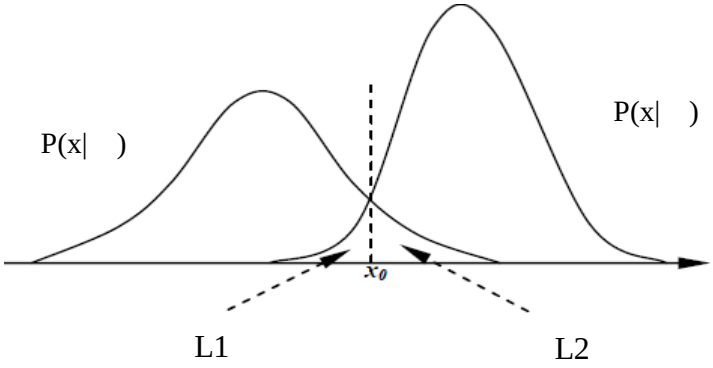
که در این عبارت (k) احتمال وقوع دسته k یا احتمال پیشین دسته iام بدون در نظر گرفتن بردار X است. ( ) احتمال وقوع بردار x و (|) احتمال مشاهده بردار x در میان نمونه‌های مربوط به دسته k است. به عبارت دیگر، وقتی که می‌دانیم بردار مذکور به دسته k تعلق دارد احتمال (|) که به آن اصطلاحا تابع درست‌نمایی دسته iام نسبت به x گفته

[1] Priori probability
[2] Likelihood function

می‌شود نشان دهنده احتمال رویت بردار x است. حال اگر روابط (1) و (2) را با هم ترکیب کنیم، به معیار مقایسه زیر می‌رسیم:

$$L(x) \equiv \frac{\quad}{(\ |\ )} \quad , \quad \begin{matrix} (\ ) > \frac{(\ )}{(\ )} \rightarrow \\ (\ ) < \frac{(\ )}{(\ )} \rightarrow \end{matrix} \qquad (3)$$

که ( ) را نسبت درست‌نمایی[1] و نسبت $\frac{(\ )}{(\ )}$ را مرز مقایسه[2] می‌نامند. در شکل3-1 نمونه یک بعدی از این معیار مشاهده می‌شود.



شکل3-1: نمونه‌ای از 2 دسته توابع درست نمایی و مرز بین آنها

مطابق این شکل x مرز تساوی توابع درست‌نمایی (|) و (|) می‌باشد. بنا به معیار (3) اگر x > ، جزء دسته k محسوب می‌شود و اگر x < ، x جزء دسته k به حساب می‌آید (درحالت کلی برداری از ویژگی‌هاست که در این مثال یک بعدی، استثنائا از نماد بردار استفاده نشده است). در عمل راحت‌تر است که با منفی لگاریتم ( ) کار کنیم که تابع تفکیک[3] خوانده می‌شود:

$$h\ (\ ) \equiv -\ \ (\ ) = \quad (\ |\ ) - \quad (\ |\ ) \qquad (4)$$

---

[3] Likelihood ratio
[4] Threshold
[1] Discrimination function

در ادامه خواهیم دید که این فرم از رابطه در صورت گوسی بودن توابع توزیع، ما را به یک تابع تفکیک خطی می‌رساند. طبیعتا در مواردی که دو دستهٔ مورد نظر هم احتمال باشند، (k) = (k) می‌گردد و مرز مقایسه ( ) h عدد صفر خواهد بود. بدین ترتیب ( )h 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 بیز[1] گویند.

اگر مجددا 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) 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 درست[2] انجام شده است. اما وقتی از معیارهای آماری همچون معیار بیز برای دسته‌بندی استفاده می‌کنیم بسته به شکل (|) و (|) و میزان همپوشانی آن‌ها، بروز حالات (2) و (3) اجتناب ناپذیر هستند.

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

$$= \quad (\mid) \quad + \quad (\mid) \qquad (5)$$

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

[2] Bayes classifier
[1] Correct classification (CC)

محاسبه توابع درست‌نمایی (|) و (|) ، تشکیل ( ) و ( )h_{12} میسر نمی‌باشد و تنها در حالاتی که (|) ها دارای توزیع گوسی با متوسطهای M و ماتریس‌های کوواریانس Σ باشند، ( ) h به فرم زیر خواهد بود:

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

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

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:

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

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.

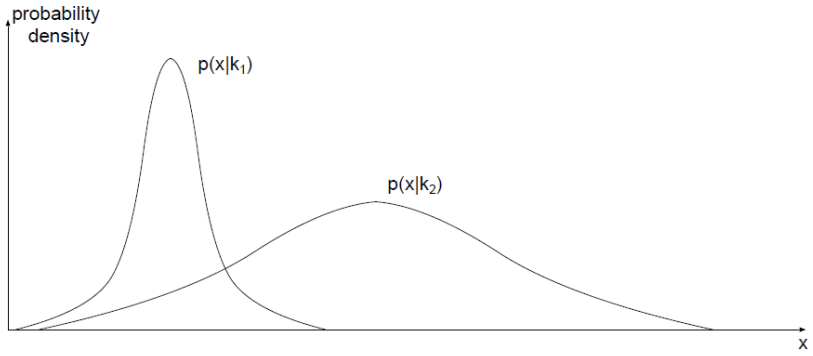
برای تعمیم روش دسته‌بندی تفکیک‌کننده بیز به بیش از دو دسته، از ایده تفکیک‌کننده‌های تکه‌ای خطی استفاده می‌کنیم. برای این منظور فرض کنید که بخواهیم بردار ویژگی را در یکی از N دسته (i = 1,2, … , N) قرار دهیم. در این‌صورت مطابق رابطه (6) و یا هر معیار فاصله دیگر $h_{ij}(x)$ را به ازای کلیه مقادیر i و j تشکیل می‌دهیم و در صورت وجود

یک که به ازای آن شرط $h_{\neg}(x) < 0$ $(i \neq k, i = 1,2,\ldots,N)$ برقرار باشد، را در دسته ام جای می‌دهیم. به عبارت دیگر:

$$h_{\neg}(x) < 0, h_{\neg}(x) < 0, \ldots, h_{\neg}(x) < 0 \Rightarrow \in k \qquad (9)$$

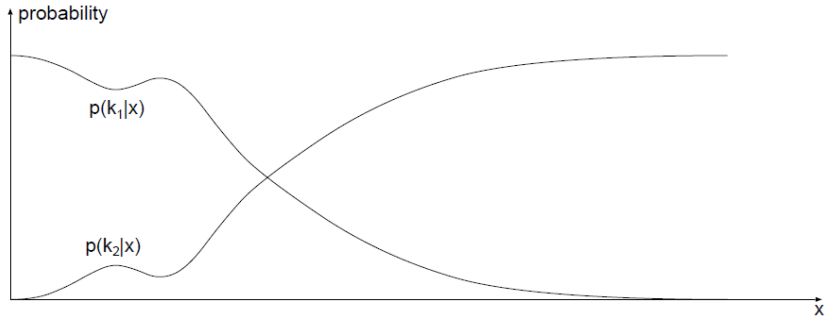
طبیعتا ممکن است برای برخی از ها نتوانیم آن را مطابق این رابطه در هیچ یک از دسته‌ها جای دهیم. بدین ترتیب مطابق شکل 3-2 نواحی ایجاد می‌شوند که با استفاده از معیار فوق به هیچ دسته‌ای تعلق ندارند. در مبحث تفکیک‌کننده‌های تکه‌ای خطی به این نواحی، نواحی طرد[1] گفته می‌شود.

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 درست نمایی مربوط به این دو دسته مطابق شکل عع نمایش داده شود، تابع احتمالاتی پسین آن به صورت شکل عع خواهد بود.



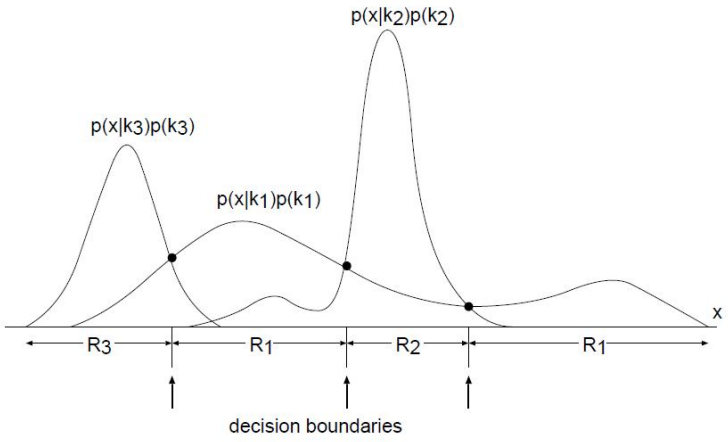
شکل3-2: تابع راست نمایی برای دو کلاس

[1] Reject region



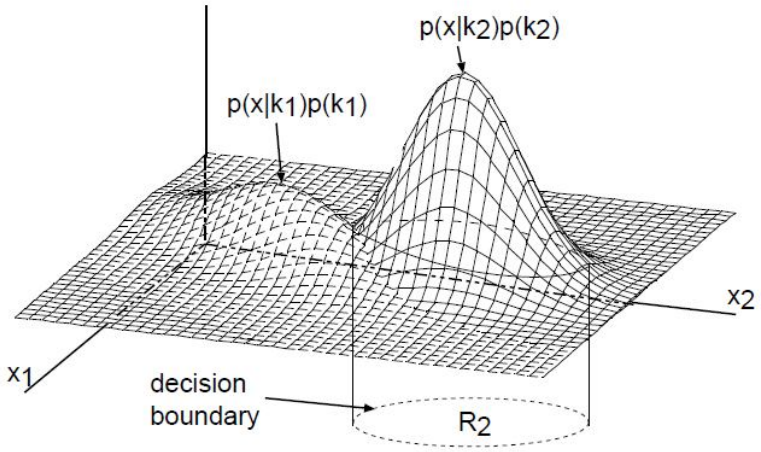
شکل3-2: تابع احتمالاتی پسین

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



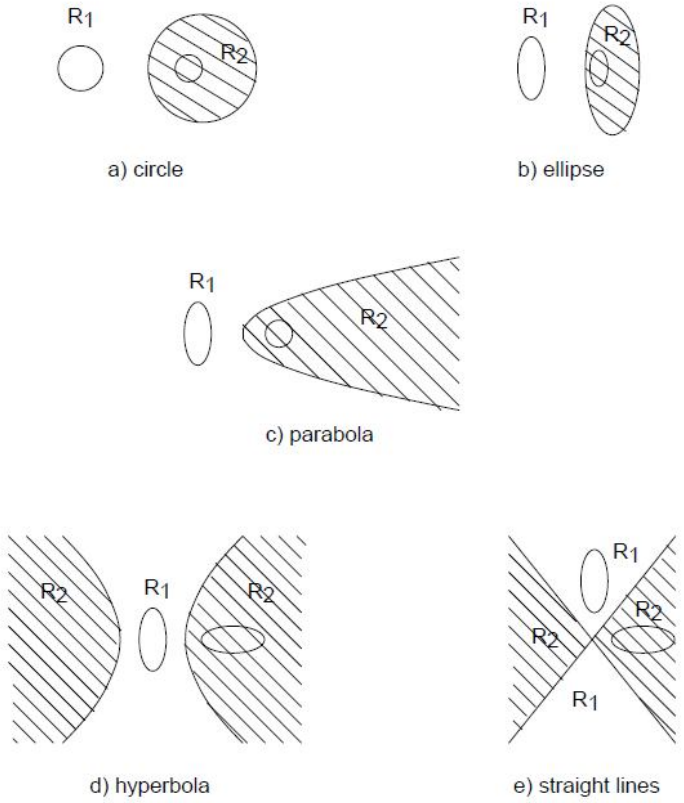
شکل3-2: حدود مرزی بین سه کلاس در فضای یک بعدی

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



شکل3-2: 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.



شکل3-2: 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 می‌شود زیرا کلاس اول در مقایسه با کلاس دوم پراکندگی و فاصله بیشتری را تا مرکز تحمل می‌کند.

4-3-3: ارزیابی متقابل

به طور کلی در این نوع از دسته‌بندی‌کننده‌ها، 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.

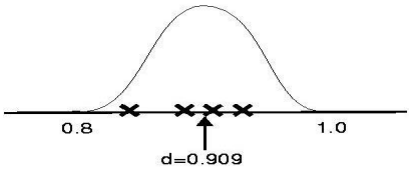
در زیر به بررسی دو نوع از پر کاربردترین الگوریتم‌های مبتنی بر این مدل پرداخته شده است.

4-4 :یادگیری بیز

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

$$\begin{aligned} p(\vartheta_k; x_{1k}, \ldots, x_{N_k k}) &= p(\vartheta_k) \cdot p(x_{1k}, \ldots, x_{N_k k} | \vartheta_k) \\ &= p(\vartheta_k) \cdot \prod_{n=1}^{N_k} p(x_{nk} | \vartheta_k) \\ &\quad \text{with the model} \quad p(x_{nk} | \vartheta_k) \end{aligned}$$

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.



شکل3-11: 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:

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

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 آماری**[2]

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

- **ساختارهای پویا**[3]

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.

---

[2] Statistic structures
[3] Daynamic structures

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

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

که f خروجی سیستم شناسایی کننده iام 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 Korkshop on Text and Keb Mining, Melbourne,
6. Y. Yang. An evaluation of statistical approaches to text categorization. Journal of Information Retrieval. 1999
7. BURGES, Christopher J. C., A Tutorial on Support Vector Machines for Pattern Recognition, 1998.
8. HEARST, Marti A., Trends & Controversies: Support vector machines, 1998.
9. CRISTIANINI, N. and J. SHAKE-TAYLOR, An Introduction to Support Vector Machines and other kernel-based learning methods
10. BERKICK, An Idiot's guide to Support vector machines (SVMs)
11. ASANO, A., Support vector machine and kernel method, Pattern information processing (2004 Autumn Semester) Session 12 (05. 1. 21)
12. RIFKIN, Ryan, Current Topics of Research III: Theory And Implmentation Of Support Vector Machines Rifkin