فصل

۱

توابع جداساز خطی

در این فصل توابع جداساز خطی[1] را مورد بررسی قرار خواهیم داد. توابع جداساز خطی شامل دو دسته مجزای مبتنی بر فواصل و مبتنی بر توزیع‌های آماری می‌باشند. از آنجایی که توابع جداساز آماری در فصل جداگانه‌ای بررسی شده‌اند، بحث در مورد توابع جداساز خطی آماری نیز به همانجا محول شده است.

تعریف جداسازها و انواع آنها، پرسپترون، جداساز خطی فیشر، ماشین‌های بردار پشتیبان خطی و روش‌های بر پایه کمینه‌سازی خطا مطالبی هستند که در این فصل به طور مفصل مورد بحث و بررسی قرار گرفته‌اند.

یک جداساز، تابعی است که هر بردار ویژگی x را به یکی از k کلاس موجود نسبت می‌دهد. جداسازها از یک دید به دو دسته قطعی و غیرقطعی (آماری) تقسیم می‌شوند. جداسازهای قطعی بر پایه فواصل و جداسازهای آماری بر پایه توزیع‌های احتمالاتی عمل می‌کنند.

جداسازهای قطعی بر پایه فواصل بین بردارهای ویژگی عمل می‌کنند.

در دسته‌بندی بر پایه فواصل، از این واقعیت استفاده می‌شود که «بردارهای ویژگی نزدیک به هم در فضای ویژگی، بیان‌گر اشیای شبیه به هم در دنیای واقعی می‌باشند». به عنوان مثال در یک مساله دسته‌بندی دو کلاسه، اگر t_1 و t_2 نماینده‌های دو کلاس C_1 و C_2 باشند، بردار ویژگی x به کلاس C_1 تعلق دارد اگر فاصله x از t_1 کمتر از فاصله x از t_2 باشد و بردار ویژگی x به کلاس C_2 تعلق دارد اگر فاصله x از t_2 کمتر از فاصله x از t_1 باشد. یعنی،

$$\|x-t_1\| \underset{C_2}{\overset{C_1}{\lessgtr}} \|x-t_2\| \qquad (۱)$$

با توجه به اینکه $\|x-t_i\|^2=(x-t_i)^t(x-t_i)$، خواهیم داشت:

$$(x-t_1)^t(x-t_1) \underset{C_2}{\overset{C_1}{\lessgtr}} (x-t_2)^t(x-t_2) \;\Rightarrow\; t_1{}^t t_1-2x^t t_1 \underset{C_2}{\overset{C_1}{\lessgtr}} t_2{}^t t_2-2x^t t_2$$

و با در نظر گرفتن $g_i(x)=t_i{}^t t_i-2x^t t_i$، رابطه (۱) به فرم زیر در می‌آید:

$$g_1(x) \underset{C_2}{\overset{C_1}{\lessgtr}} g_2(x)$$

که می‌توان آن را به صورت زیر نیز بیان نمود:

$$g(x)=\left(g_2(x)-g_1(x)\right) \underset{C_2}{\overset{C_1}{\gtrless}} 0 \qquad (۲)$$

---

[1] Linear Discriminan Functions (LDF)

این تابع برای داده‌های کلاس اول مقدار بزرگتر از صفر و برای داده‌های کلاس دوم مقدار کوچکتر از صفر را بر می‌گرداند. از آنجایی که از تابع $g(x)$ برای تفکیک دو کلاس استفاده می‌شود، به آن تابع جداساز گفته می‌شود. به توابعی جداسازی که همانند بردار ویژگی فوق از درجه اول باشند، توابع جداساز خطی گفته می‌شود.

فرم کلی توابع جداساز خطی

برای $x=(x_1,x_2,...,x_d)$، یک بردار ویژگی در یک فضای d بعدی، تابع جداساز خطی به یکی از صورت‌های زیر می‌باشد:

$$g(x)=w^t x+w_0 \qquad (۳)$$

که در آن $w=(w_1,w_2,...,w_d)$ می‌باشد. فرم فوق را فرم معمولی توابع جداساز خطی می‌نامند. پس در دسته‌بندی دو کلاسه بوسیله توابع جداساز، باید تابعی ارائه کنیم که بوسیله آن بتوان، بطور قطعی کلاس هر بردار ویژگی جدید را تعیین نمود. برای ارائه تابع جداساز نیز باید پارامترهای (w,w_0) مشخص شوند.

فرم همگن توابع جداساز خطی

اگر بردارهای ویژگی را به صورت افزوده[۲] در نظر بگیریم، یعنی $x=(1,x_1,x_2,...,x_d)$ و همچنین برای بردار وزن‌ها نیز داشته باشیم $w=(w_0,w_1,w_2,...,w_d)$، رابطه (۳) به فرم زیر در می‌آید:

$$g(x)=w^t x \qquad (۴)$$

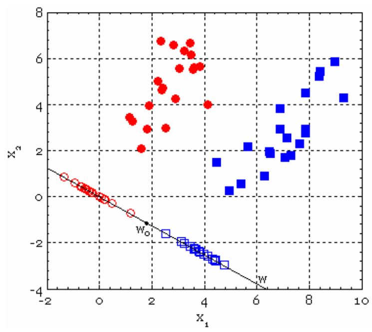
این فرم توابع جداساز را فرم همگن[۳] می‌نامند. در حل تحلیلی مسایل، کار کردن با فرم همگن توابع جداساز، راحت‌تر است. مطابق رابطه (۴)، اگر بردار ویژگی x متعلق به کلاس C_1 باشد، $w^t x>0$ و اگر بردار ویژگی x متعلق به کلاس C_2 باشد، $w^t x<0$ یا $w^t(-x)>0$ خواهد بود.

فرم نرمال توابع جداساز خطی

اگر تمامی بردارهای ویژگی کلاس C_2 را قرینه کنیم، برای تمامی بردارهای ویژگی متعلق به هر دو کلاس $g(x)=w^t x>0$ خواهد شد. با این کار فرم نرمال تابع جداساز خطی بدست می‌آید. استفاده از فرم نرمال در حل مسایل، در بسیاری از حالات، محاسبات را ساده‌تر می‌کند.

شهود هندسی دسته‌بندی با توابع جداساز خطی

در فرم معمولی یک تابع جداساز خطی، جزء $w^t x$ بیان می‌کند که داده‌ها باید بر روی ابر صفحه با بردار مشخصه w تابانده شوند. بر روی این ابرصفحه نیز w_0 مرز جداسازی را تعیین می‌کند. این شهود در شکل ۱، برای حالت دوبعدی (که ابر صفحه برابر خط خواهد بود) نمایش داده شده است.



شکل ۱: شهود هندسی دسته‌بندی با توابع جداساز خطی

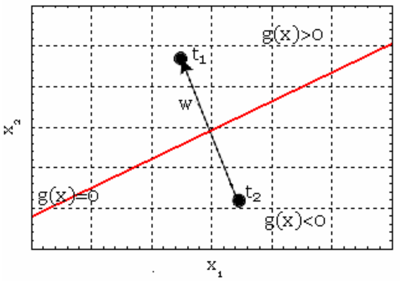
مرز جداسازی

در یک تابع جداساز، حالت تساوی با صفر را بررسی نکردیم. اگر $g(x)=0$ (در تمامی فرم‌های نمایش)، در این صورت بردار ویژگی x به کدام کلاس تعلق خواهد داشت؟ واقعیت این است که در این مورد نمی‌توان به درستی اظهار نظر کرد، چون این بردار ویژگی بر روی مرز بین دو کلاس قرار گرفته است. در حالت کلی به مجموعه تمامی نقاطی که به ازای آنها $g(x)=w^t x+w_0=0$ می‌باشد، مرز جداسازی (مرز تصمیم‌گیری) گفته می‌شود.

برای یک تابع جداساز خطی، مرز جداسازی در یک فضای دو بعدی، یک خط، در یک فضای سه بعدی یک صفحه و در حالت کلی در یک فضای d بعدی، یک ابر صفحه (d-1) بعدی خواهد بود. مفاهیم مطرح شده فوق در شکل ۲ در یک فضای ویژگی دو بعدی به تصویر کشیده شده‌اند:

[2] Augmented Feature Vector
[3] Hemogeneous

نمایش تصویری مرز جداسازی و مفاهیم مربوطه



شکل ۲: مرز جداسازی در فضای دو بعدی

در این شکل، t_1 و t_2 مراکز دو کلاس C_1 و C_2 بوده و w خط واصل t_2 و t_1 می‌باشد. خط قرمز رنگ مرز جداسازی دو کلاس (عمود بر w) می‌باشد. یک طرف خط متعلق به کلاس C_1 بوده و در آن $g(x)>0$ می‌باشد (بالای خط مرز در شکل). طرف دیگر خط نیز متعلق به کلاس C_2 بوده و در آن $g(x)<0$ می‌باشد (پایین خط مرز در شکل).

جداپذیری خطی کامل و جداپذیری خطی دوبدو

مفاهیم دسته‌بندی خطی را برای حالت دو کلاسه مورد بررسی قرار دادیم. در بسیاری از مسایل، تعداد کلاس‌ها ممکن است بیش از دو کلاس باشد. در این حالت برای دسته‌بندی می‌توان به دو صورت عمل نمود. یکی اینکه، برای جداسازی بردارهای ویژگی هر کلاس از سایر کلاس‌ها، یک تابع جداساز ارائه نمود. در این حالت برای دسته‌بندی n کلاس به n تابع جداساز نیاز است (البته دسته‌بندی با n-1 تابع جداساز نیز قابل انجام است. چگونه؟). و دیگر اینکه برای جداسازی بردارهای ویژگی یک کلاس از هر کلاس دیگر، یک تابع جداساز داشته باشیم. در این حالت برای دسته‌بندی n کلاس به $n(n-1)/2$ تابع جداساز نیاز است. در حالت اول مجموعه بردارهای ویژگی را جداپذیر خطی کامل[۴] و در حالت دوم مجموعه بردارهای ویژگی را جداپذیر خطی دوبدو[۵] گویند.

مثال ۱: یک مساله دسته‌بندی سه کلاسه را در نظر بگیرید. در این مساله برای هر کلاس یک تابع معرف به صورت زیر ارائه شده است:

$$g_1(x)=-x_1+x_2\,,\ g_2(x)=x_1+x_2-1,\ g_3(x)=-x_2$$

برای تشخیص تعلق یک داده به یک کلاس نیز از قانون زیر استفاده می‌شود:

$$x\in C_i \Leftrightarrow g_i(x)<g_j(x)\ ;\forall j\neq i$$

هر جداساز با فرم فوق را یک ماشین خطی می‌نامند. این ماشین خطی فضای ویژگی را به چه صورت افراز می‌کند؟

در این مثال ناحیه مربوط به هر کلاس، مطابق زیر، با دو نامساوی خطی مشخص می‌شود:

ناحیه ۱

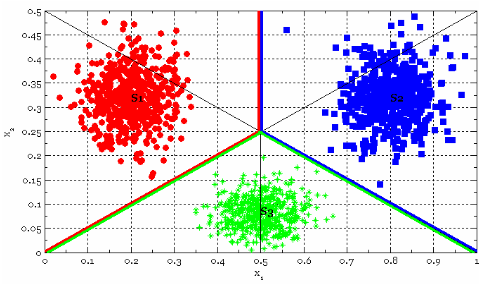
$$\underline{x}\in S_1:\begin{cases} g_1>g_2 \Rightarrow x_1<1/2 \\ g_1>g_3 \Rightarrow x_2>x_1/2 \end{cases}$$

ناحیه ۲

$$\underline{x}\in S_2:\begin{cases} g_2>g_1 \Rightarrow x_1>1/2 \\ g_2>g_3 \Rightarrow x_2>1/2(-x_1+1) \end{cases}$$

ناحیه ۳

$$\underline{x}\in S_3:\begin{cases} g_3>g_1 \Rightarrow x_2<x_1/2 \\ g_3>g_2 \Rightarrow x_2<1/2(-x_1+1) \end{cases}$$



❒

[۴] Totally Linearly Seperable
[۵] Pairwise Linearly Seperable

مثال ۲: یک مساله دسته‌بندی با سه کلاس را دز نظر بگیرید که در آن بردارهای ویژگی جداپذیر خطی کامل هستند. توابع جداساز را نیز به صورت زیر در نظر بگیرید:

$$g_1(x)=-x_1+x_2\,,\;\; g_2(x)=x_1+x_2-5,\;\; g_3(x)=-x_2+1$$

توابع جداساز به قسمی ارائه شده‌اند که $g_i(x)>0$ نمایان‌گر تعلق x به کلاس iام و $g_i(x)<0$ نمایان‌گر عدم تعلق آن به کلاس iام باشد. این جداساز فضای ویژگی را به چه صورت افراز می‌کند؟ نواحی‌ای که جداساز قادر به تصمیم‌گیری در مورد آنها نیست، را نیز مشخص نمائید.

هر ناحیه با سه نامساوی خطی روی x تعریف می شود (یعنی هر ناحیه بین سه خط محصور است).

ناحیه ۱

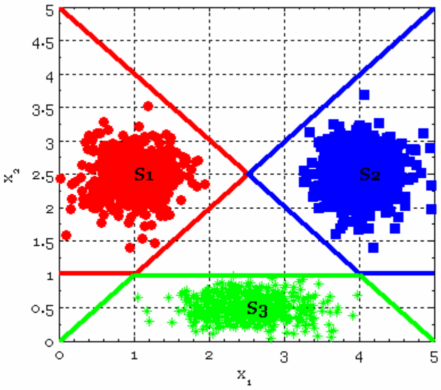
$$x\in S_1:\begin{cases}-x_1+x_2>0\\ x_1+x_2-5<0\\ -x_2+1<0\end{cases}$$

ناحیه ۲

$$x\in S_2:\begin{cases}-x_1+x_2<0\\ x_1+x_2-5>0\\ -x_2+1<0\end{cases}$$

ناحیه ۳

$$x\in S_3:\begin{cases}-x_1+x_2<0\\ x_1+x_2-5<0\\ -x_2+1>0\end{cases}$$



نواحی‌ای نیز که مقدار بیش از یک تابع برای داده‌های آنها، مثبت باشد، نواحی غیر قابل تصمیم‌گیری می‌باشند (شامل چهار ناحیه مثلثی در شکل).

❐

مثال ۳: توابع جداساز، برای یک جداساز جداپذیر خطی دوبدوی سه کلاسه، را به صورت زیر در نظر بگیرید:

$$g_{12}(x)=-x_1-x_2+5,\;\; g_{13}(x)=-x_1+3,\;\; g_{23}(x)=-x_1+x_2$$

(و $g_{ij}(x)=-g_{ji}(x)$). در این جداساز، قانون تصمیم‌گیری به صورت زیر می‌باشد:

$$x\in C_i \Leftrightarrow \forall j\neq i\; g_{ij}(x)>0$$

این جداساز فضای ویژگی را به چه صورت افراز می‌کند؟ نواحی‌ای که جداساز قادر به تصمیم‌گیری در مورد آنها نمی‌باشد را مشخص نمائید.

هر ناحیه با دو نامساوی خطی مشخص می‌شود که هر نامساوی متناظر با جداسازی کلاس مورد نظر از کلاس‌های دیگر است. در نتیجه هر ناحیه با دو خط محصور است.

ناحیه ۱

$$x\in S_1:\begin{cases}-x_1-x_2+5>0\\ -x_1+3>0\end{cases}$$

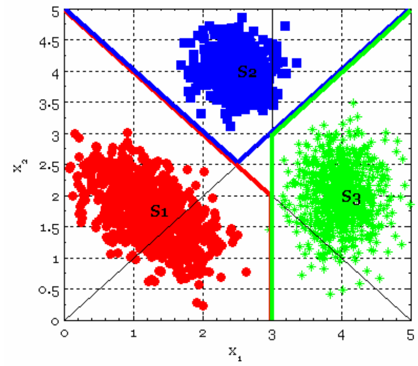
ناحیه ۲

$$x\in S_2:\begin{cases}x_1+x_2-5>0\\ -x_1+x_2>0\end{cases}$$

ناحیه ۳

$$x\in S_3:\begin{cases}x_1-3>0\\ x_1-x_2>0\end{cases}$$

همانطور که در شکل مشخص است، ناحیه مثلثی وسط، در هیچ یک از کلاس‌ها قرار نمی‌گیرد. زیرا در آن فقط $g_{13}(x)$ بزرگ‌تر از صفر می‌باشد.



❐

تمرین ۱: نشان دهید یک ماشین خطی، حالت خاصی از جداساز جداپذیر خطی دوبدو می‌باشد.

تمرین ۲: نشان دهید در یک مساله دسته‌بندی چند کلاسه با یک ماشین خطی، نواحی بدست آمده برای هر کلاس (ناحیه حاصل از مرزهای جداساز آن کلاس) محدب می‌باشند. (راهنمایی: برای نشان دادن محدب بودن یک ناحیه کافی است نشان دهید که $x_1 \in R_i \; and \; x_2 \in R_i \Rightarrow \lambda x_1 + (1-\lambda)x_2 \in R_i \;\; if \; 0 \le \lambda \le 1$. چرا؟)

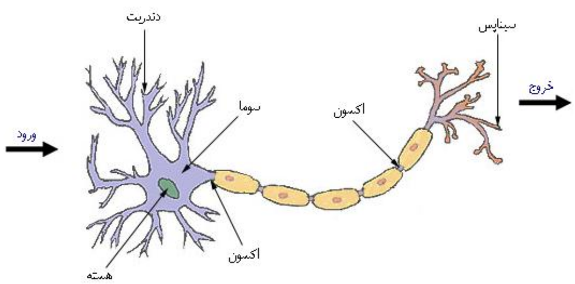
نحوه بدست آوردن توابع جداساز برای مسائل مختلف

مساله اصلی در بحث توابع جداساز خطی، نحوه بدست آوردن توابع به ازای هر مساله خاص می‌باشد. در ادامه این مبحث به روش‌های مختلف بدست آوردن توابع جداساز خطی می‌پردازیم. ابتدا نرون‌های عصبی مصنوعی را که شکل دیگری از نمایش توابع جداساز می‌باشند، معرفی نموده و الگوریتم‌های پرسپترون و Relaxation را، که بر پایه کاهش گرادیان می‌باشند، برای بدست آوردن پارامترهای آنها توضیح می‌دهیم. سپس به جداساز خطی فیشر می‌پردازیم که بر اساس تابش بردارهای ویژگی در فضای کاهش یافته جدیدی، که جداسازی بردارهای ویژگی را آسان‌تر می‌کند، عمل می‌کند. پس از آن به ماشین‌های بردار پشتیبان خطی می‌پردازیم که جداسازهایی با بیشترین حاشیه را می‌یابند. پیدا کردن توابع جداساز با روش‌های بر پایه کمینه‌سازی خطا نیز بعد از آن مورد بررسی قرار می‌گیرند که شامل روش‌های تخمین میانگین مربعات خطا، الگوریتم Widrow-Hoff، روش ماتریس شبه معکوس و متد Ho-Kahyap می‌باشند. مقایسه روش‌ها و چند بحث تکمیلی نیز پایان بخش این مبحث خواهند بود.

۱- پرسپترون

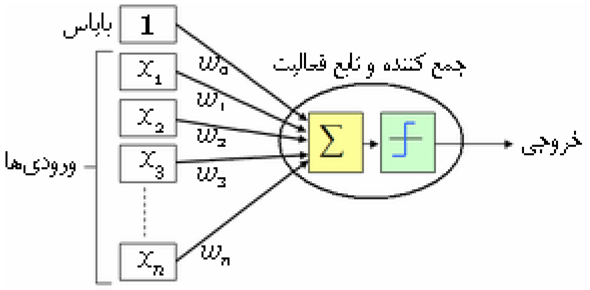
پرسپترون و نرون‌های عصبی طبیعی

پرسپترون و دیگر شبکه‌های عصبی مصنوعی الهام گرفته از ساختار عصبی بدن انسان هستند. کوچکترین جز در این ساختار نرون‌ها می‌باشند. ساختار یک نرون طبیعی در شکل ۳ نشان داده شده است. در این ساختار بدنه نرون وظیفه جذب پتاسیم توسط دندریت‌ها را دارد. این پتاسیم در بدنه ذخیره شده و هر وقت مقدار آن از حد معینی تجاوز کند، بدنه از طریق سیناپس‌ها شروع به پمپاژ پتاسیم به بیرون می‌کند. پتاسیم پمپاژ شده در فضای سیناپسی توسط دندریت‌های سایر نرون‌هایی که در فضای مجاور قرار دارند، جذب می‌شود.



شکل ۳: ساختار یک نرون طبیعی در شبکه عصبی موجودات زنده

در طراحی پرسپترون نیز از همین ایده الهام گرفته شده است. یک پرسپترون برداری از ورودی‌های با مقادیر حقیقی را گرفته و یک ترکیب خطی از این ورودی‌ها را محاسبه می‌کند. اگر حاصل از یک مقدار آستانه بیشتر بود، مقدار ۱ و در غیر این صورت مقدار ‎-۱ (یا صفر) را به خروجی خود می‌فرستد (شکل ۴).



شکل ۴: ساختار یک نرون مصنوعی (پرسپترون)

تابع پرسپترون

پرسپترون به هر کدام از ورودی‌های خود، وزن مشخصی می‌دهد. اگر دارای n ورودی $x_1,x_2,...,x_n$ باشیم و وزن‌های معادل این ورودی‌ها $w_1,w_2,...,w_n$ باشند، در این صورت ورودی پرسپترون برابر خواهد بود با:

$$w_1x_1+w_2x_2+...+w_nx_n=\sum_{i=1}^{n}w_ix_i$$

پروسپترون بر اساس بیشتر شدن یا نشدن ورودی‌اش از حد مشخصی (w_0)، خروجی‌های خود را تولید خواهد کرد. این مقایسه را می‌توان بصورت نشان داد. و اگر ورودی دیگری با مقدار ۱ و وزن w_0 برای پرسپترون قائل شویم، مقایسه بصورت $\sum_{i=0}^{n}w_ix_i\geq 0$ (فرم همگن) در خواهد آمد. در نتیجه تابع یک پرسپترون بصورت زیر مشخص می‌شود:

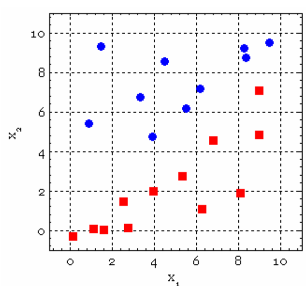
$$f(x_1,x_2,...,x_n)=\begin{cases}1 & if\ \sum_{i=0}^{n}w_ix_i\geq 0\\ -1 & otherwise\end{cases} \qquad (۵)$$

حل مسائل با پرسپترون

حال که ساختار پرسپترون را بررسی کردیم، نحوه استفاده از آن را برای حل مسایل دسته‌بندی بررسی می‌کنیم. یک مساله دسته‌بندی دو کلاسه در یک فضای d بعدی را در نظر بگیرید. و همچنین فرض کنید برای طراحی یک جداساز مناسب برای این مساله، n بردار ویژگی (داده‌های آموزشی) در اختیار داریم. این مساله چگونه با یک پرسپترون حل می‌شود؟

جداساز مورد نیاز برای حل این مساله، یک پرسپترون با d+1 ورودی (d ورودی برای دریافت ابعاد مختلف هر کدام از بردارهای ویژگی و یک ورودی نیز برای مشخص کردن حد آستانه) و یک خروجی با مقادیر ۱ یا ‎-۱ (برای نشان دادن تعلق به کلاس اول یا دوم) می‌باشد. وزن‌های نامشخص پرسپترون ($w_0,w_1,...,w_d$) باید با توجه به n داده آموزشی و ساختار تعیین شده، مشخص شوند. با مشخص شدن این وزن‌ها، تمامی قسمت‌های پرسپترون شناخته شده و تابع پرسپترون قادر خواهد بود که برای هر بردار ویژگی، کلاس آن را تعیین نماید.

مثال ٤: یک مساله دسته‌بندی دو کلاسه در یک فضای دو بعدی را با بردارهای ویژگیی آموزشی نشان داده در شکل زیر در نظر بگیرید. یک پرسپترون برای حل این مساله طراحی نمائید.



برای حل این مساله نیاز به یک پرسپترون با سه ورودی و یک خروجی داریم. اگر خروجی پرسپترون صفر باشد، نشان دهنده

یکی از کلاس‌ها (مثلا مربع‌ها) و اگر ۱ باشد، نشان دهنده کلاس دیگر (دایره‌ها) می‌باشد. برای طراحی کامل پرسپترون مربوطه، تنها مجهول‌های باقیمانده، وزن‌ها (w_2,w_1,w_0) می‌باشند. تابع پرسپترون برای این مثال نیز بصورت زیر می‌باشد:

$$f(x_1,x_2)=\begin{cases}1 & if\ w_2x_2+w_1x_1-w_0\geq 0\\ -1 & otherwise\end{cases} \qquad (۶)$$

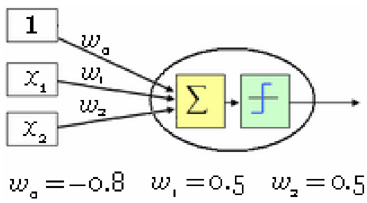
از طرفی، با توجه به شکل، می‌دانیم که معادله خط جداساز دو کلاس $x_2-x_1=0$ بوده و بالای خط ($x_2-x_1>0$)، دایره‌ها (کلاس ۱) و پایین خط ($x_2-x_1<0$)، مربع‌ها (کلاس ۱-) قرار گرفته‌اند.

با معادل‌سازی $w_2x_2+w_1x_1-w_0\geq 0$ و $x_2-x_1>0$ و همچنین معادل‌سازی $w_2x_2+w_1x_1-w_0<0$ و $x_2-x_1<0$ به این نتیجه می‌رسیم که $(w_2,w_1,w_0)=(1,-1,0)$. این نتیجه طراحی پرسپترون ما را کامل می‌کند.

❐

پرسپترون و شبیه‌سازی عملگرهای منطقی

یک پرسپترون می‌تواند بسیاری از توابع منطقی نظیر and، or، nand و nor را شبیه‌سازی. به عنوان مثال پرسپترون زیر عملکرد and را شبیه‌سازی می‌کند (خروجی پرسپترون برابر $x_1\, and\, x_2$ می‌باشد).



شکل ۵: شبیه‌سازی عملکرد and توسط پرسپترون

تمرین ۳: توابع or، nand و nor را با پرسپترون شبیه‌سازی کنید.

یک پرسپترون فقط قادر به دسته‌بندی خطی است. مسائلی که توسط یک ابر صفحه قابل جداسازی باشند. در نتیجه قادر به شبیه‌سازی توابعی نظیر xor نیز نمی‌باشد. در فصل‌های آتی، شبکه‌های عصبی چند لایه را بررسی خواهیم کرد که قادر به حل مسائل غیر خطی نیز می‌باشند.

تعیین سیستماتیک وزن‌های پرسپترون

همانطور که مشاهده کردید، در مثال‌های فوق برای تعیین وزن‌ها در طراحی پرسپترون از دانش شهودی خود استفاده کردیم. این دانش شهودی همیشه وجود ندارد یا کسب آن به راحتی امکان‌پذیر نیست. پس باید بدنبال روش مناسبی (یک الگوریتم) برای تعیین وزن‌ها، بصورت سیستماتیک باشیم.

اگر مساله را به شکل نرمال در نظر بگیریم، هدف از این الگوریتم این است که $w=(w_0,w_1,...,w_d)$ را طوری بیابد که به ازای همه بردارهای ویژگی ورودی (x ها)، داشته باشیم

$$g(x)=w^t x>0 \qquad (۷)$$

روش‌های تکراری برای حل مسایل بهینه‌سازی

یکی از روش‌های پیدا کردن w استفاده از الگوریتم‌های بهینه‌سازی تکراری می‌باشد. برای این کار، ابتدا w را بطور تصادفی انتخاب کرده و میزان خوبی آن را می‌سنجیم. اگر w بردار خوبی بود، جواب مساله پیدا شده است، ولی اگر به اندازه کافی خوب نبود، باید آن را طوری تغییر دهیم که عملکرد تابع جداساز معادل آن بهتر شود. برای بردار جدید بدست آمده نیز، باید همین عملیات آزمون و تغییر را به صورت متوالی انجام دهیم، تا جایی که بردار خوبی بدست آید.

روش پیشنهاد شده، از لحاظ تئوری روش مناسبی می‌باشد. ولی جهت اجرا، باید بر دو مشکل غلبه کنیم. یکی اینکه باید روشی برای تغییر بردار w در هر مرحله، به قسمی بیابیم که بردار جدید، بهتر از بردار قبل باشد و دیگر اینکه باید معیاری برای سنجش خوبی w ارائه دهیم.

روش کاهش گرادیان

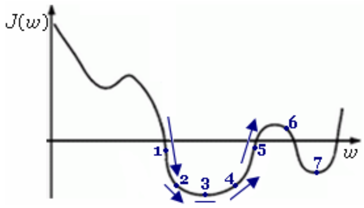
برای تولید بردار جدید از بردار قبلی از روش کاهش گرادیان[6] استفاده می‌کنیم. در این روش، هدف، پیدا کردن بردار بهینه $w=(w_0,w_1,...,w_d)$ است که تابع هزینه‌ای نظیر $J(w)$ را کمینه کند. در نتیجه برای بردار بهینه w باید داشته باشیم:

[6] Gradient Descent

$$\nabla J = \frac{\partial J}{\partial w} = [\frac{\partial J}{\partial w_0}, \frac{\partial J}{\partial w_1}, ..., \frac{\partial J}{\partial w_d}] = 0$$

شهود هندسی مفهوم گرادیان

در شکل ۶ جهت و مقدار گرادیان $\nabla J(w)$ برای یک w یک بعدی در نقاط مختلف نشان داده شده است. در نقطه کمینه مقدار گرادیان برابر صفر می‌باشد (نقطه ۳). در نقاطی که قبل از نقطه کمینه قرار دارند، جهت گرادیان (جهت شیب خط مماس بر آن نقاط) منفی بوده (نقاط ۱ و ۲) و در نقاطی که بعد از نقطه کمینه قرار دارند، جهت گرادیان مثبت می‌باشد (نقاط ۴ و ۵). مقدار گرادیان نیز در نقاطی که شیب تندتری داریم، بیشتر است (صرفنظر از علامت مقادیر، اندازه گرادیان در نقاط نمایش داده شده در شکل، به ترتیب نزولی در نقاط ۱، ۵، ۴، ۲ و ۳ می‌باشد).



شکل ۶: مقدار و جهت گرادیان در نقاط دور و بر یک نقطه کمینه

با توجه به مطالب ذکر شده، اگر در مساله اصلی، در یک مرحله، w مقدار $J(w)$ را کمینه نکند، باید مقدار آن به سمت بهبود (به سمت w بهینه) تغییر پیدا کند. با توجه به تعبیر هندسی گرادیان (و با توجه به مفاهیم نشان داده شده در شکل ۶) میزان این تغییرات باید در راستای $-\nabla J$ و ضریب مثبتی از مقدار گرادیان باشد. یعنی،

$$w_{new} = w_{old} - \eta \nabla J(w) \qquad (8)$$

η را نرخ یادگیری[7] گوئیم که می‌تواند در مراحل مختلف مقادیر متفاوتی را به خود بگیرد (هر چند که در بسیاری از موارد نرخ یادگیری، ثابت در نظر گرفته می‌شود). اگر η مقدار کوچکی داشته باشد، تعداد قدم‌ها تا رسیدن به هدف بسیار زیاد خواهد شد (زیرا در هر مرحله مقدار خیلی کمی به هدف نزدیک می‌شویم) و اگر η مقدار بزرگی داشته باشد (قدم بزرگی به سمت هدف برداشته شود)، الگوریتم ممکن است قادر به پیدا کردن کمینه عمومی نبوده و یک کمینه محلی را پیدا کند. و یا اینکه اصلا نتواند نقطه بهینه‌ای را بیابد. مثلا در شکل ۶ فرض کنید در نقطه ۲ قرار داریم. می‌دانیم که برای نزدیک شدن به هدف باید به سمت راست حرکت کنیم. اگر نرخ یادگیری مقدار بزرگی باشد، قدم بزرگی به سمت راست برداشته می‌شود و مثلا به نقطه ۶ می‌رسیم. در مراحل بعدی، نقطه کمینه عمومی ۳ توسط الگوریتم قابل کشف نیست و نقطه کمینه محلی ۷ بسته به مقدار نرخ یادگیری در آن مراحل، ممکن است یافت بشود یا نشود (الگوریتم با یک قدم بزرگ دیگر خیلی راحت می تواند از این نقطه نیز رد شود).

با توجه به مطالب گفته شده، می‌توان الگوریتم کاهش گرادیان را به صورت زیر بیان نمود:

الگوریتم کاهش گرادیان برای مسایل بهینه‌سازی

الگوریتم کاهش گرادیان برای بهینه‌سازی تابع $J(w)$

الف) یک مقدار اولیه برای w در نظر بگیر $(v_{(0)})$ و متغیر k را نیز با مقدار اولیه صفر در نظر بگیر.

ب) تا وقتی که $\nabla J(w) \neq 0$ (یا در بسیاری از مسایل تا وقتی که $\nabla J(w) > \varepsilon$) مراحل زیر را تکرار کن:

ب۱) مقداری مثبت برای $\eta_{(k)}$ در نظر بگیر.

ب۲) مقدار w را مطابق قانون $w_{(k+1)} = w_{(k)} - \eta_{(k)} \nabla J(w)$ بروز کن.

ب۳) مقدار k را یکی اضافه کن.

برای همگرا شدن الگوریتم (متوقف شدن الگوریتم و یافتن w بهینه)، نرخ یادگیری، علاوه بر مثبت بودن، باید دو شرط زیر را

7 Learning Rate

نیز دارا باشد:

$$\sum_{k=1}^{\infty}\eta_{(k)} \to \infty,\ \sum_{k=1}^{\infty}\left(\eta_{(k)}\right)^2 < \infty$$

تمرین ٤: نرخ یادگیری با دارا بودن شروط همگرایی، مقادیر در چه بازه‌هایی را می‌تواند بپذیرد؟

تمرین ٥: اگر در یک مساله دسته‌بندی ندانیم که بردارهای ویژگی به صورت خطی جداپذیر هستند یا نه، بهتر است از ضریب یادگیری متغیر کوچک شونده در طی پیشروی الگوریتم استفاده نمائیم تا اثرات داده‌های هم‌پوشان، در صورت وجود، کاهش یابد (نظیر الگوریتم شبیه‌سازی حرارت). چرا این عمل باعث کاهش اثرات داده‌های هم‌پوشان می‌شود؟ آیا $\eta_{(k)} = 1/k$ می‌تواند نرخ یادگیری مناسبی برای این منظور باشد (شروط همگرایی را داراست)؟

توجه کنید که الگوریتم کاهش گرادیان تضمینی نمی‌دهد که کمینه عمومی را پیدا کند. ولی به دلیل سادگی آن و اینکه برای هر تابع بهینه‌سازی‌ای قابل اعمال هست، استفاده زیادی پیدا کرده است.

به مساله تعیین سیستماتیک وزن‌های پرسپترون برگردیم. در این مساله می‌بایستی بر دو مشکل غلبه می‌کردیم. تا اینجا بر مشکل اول غلبه کرده و روشی برای بهبود بردار w در هر مرحله، تا رسیدن به یک جواب بهینه (محلی یا عمومی)، ارائه کردیم. حال باید به دنبال غلبه بر مشکل دوم، یعنی ارائه معیاری برای سنجش خوبی بردار w (یا همان ارائه تابع $J(w)$) باشیم.

تابع بهینه‌سازی مناسب برای پیدا کردن وزن‌های بهینه در پرسپترون

باید $J(w)$ را به قسمی پیدا کنیم که به ازای w بهینه کمینه شود. بدیهی‌ترین تعریفی که برای $J(w)$ می‌توان در نظر گرفت به صورت زیر می‌باشد:

$$J(w) = |X_m(w)|;\quad X_m(w) = \{samples\ x_{(i)} \mid w^t x_{(i)} < 0\}$$

$J(w)$ فوق بیان‌گر تعدادی از بردارهای ویژگی ورودی است که $g(w)$ در دسته‌بندی آنها دچار اشتباه می‌شود. این تابع، یک تابع چند تکه‌ای، با مقادیر ثابت در هر تکه بوده (چرا؟) و برای بدست آوردن ∇J مناسب نمی‌باشد (زیرا گرادیان برای مقادیرثابت برابر صفر می‌شود). بنابر این تابع دیگری به فرم زیر برای $J(w)$ پیشنهاد می‌کنیم:

$$J(w) = \sum_{x \in X_m(w)} -w^t x;\quad X_m(w) = \{samples\ x_{(i)} \mid w^t x_{(i)} < 0\}$$

می‌دانیم برای هر بردار w و هر بردار ویژگی x رابطه $w^t x/\|w\|$ نشان دهنده فاصله بردار ویژگی x از بردار w می‌باشد. در نتیجه تابع ارائه شده فوق، مجموع فواصل بردارهای ویژگی اشتباه دسته‌بندی شده، را از بردار w نشان می‌دهد. این تابع، یک تابع چند تکه‌ای، با ضابطه‌ای خطی در هر بازه بوده (چرا؟) و در هر بازه می‌توان مقدار گرادیان را برای آن بازه محاسبه نمود.

فاصله یک نقطه از یک ابر صفحه

اگر x یک نقطه در فضا و P ابر صفحه‌ای با معادله $g(x) = w^t x + w_0 = 0$ بوده و x_P تصویر نقطه بر روی ابر صفحه و r فاصله نقطه تا ابر صفحه باشد، می‌توان نوشت: $x = x_p + r.w/\|w\|$ که در آن w بردار مشخصه صفحه و $w/\|w\|$ جهت آن (فارغ از اندازه) می‌باشد. در این صورت داریم:

$$g(x) = w^t(x_p + r\frac{w}{\|w\|}) + w_0 = w^t x_p + r\frac{w^t w}{\|w\|} + w_0 = w^t x_p + w_0 + r\frac{\|w\|^2}{\|w\|} = 0 + r\|w\|$$

در نتیجه $r = g(x)/\|w\|$.

می‌توان با استفاده از این تابع و الگوریتم کاهش گرادیان به حل مساله اقدام نمود. بخش بروز رسانی مقادیر w در الگوریتم نیز به صورت $w_{(k+1)} = w_{(k)} + \eta_{(k)}\sum_{x \in X_m} x$ در می‌آید. با توجه به این قانون در هر بار بروز رسانی، مقدار w با توجه به تمامی بردارهای ویژگی اشتباه دسته‌بندی شده (یا در واقع، متناسب با مجموع فواصل بردارهای ویژگی اشتباه دسته‌بندی شده، با مرز جداسازی)، بروز می‌شود. توجه داشته باشید که بدلیل وابستگی $J(w)$ به X_m، پیدا کردن w بهینه برای $J(w)$، بصورت

تحلیلی امکان‌پذیر نمی‌باشد.

مثال ۵: یک مساله دسته‌بندی دو کلاسه، در یک فضای چهار بعدی را با بردارهای ویژگی زیر در نظر بگیرید:

$$C_1 = \{(1,1,-1,-1),(1,-1,-1,1)\},\ C_2 = \{(1,1,1,1),(-1,-1,-1,1)\}$$

می‌خواهیم دسته‌بندی این داده‌ها را با یک پرسپترون انجام دهیم. وزن‌های این پرسپترون (w_0,w_1,w_2,w_3,w_4) را بدست آورید.

ابتدا بردارهای ویژگی داده شده را به بردارهای افزوده تبدیل کرده و سپس آنها را نرمال می‌کنیم:

$$C = \{(1,1,1,-1,-1),(1,1,-1,-1,1),(-1,-1,-1,-1,-1),(-1,1,1,1,-1)\}$$

با در نظر گرفتن نرخ یادگیری به صورت $\eta_{(k)} = \eta = 1$، قانون بروز رسانی به فرم زیر در می‌آید:

$$w_{(k+1)} = w_{(k)} + \sum_{x \in X_m} x$$

با در نظر گرفتن بردار اولیه بصورت $w_{(0)} = \{0.25,0.25,0.25,0.25,0.25\}$، الگوریتم را اجرا می‌کنیم. داریم:

$$g(x_{(1)}) = 0.25 > 0,\ g(x_{(2)}) = 0.25 > 0,\ g(x_{(3)}) = -1.25 < 0,\ g(x_{(4)}) = 0.25 > 0$$

$x_{(3)}$ اشتباه دسته‌بندی شده است، پس $Z_m(w_{(0)}) = \{x_{(3)}\}$ در نتیجه $w_{(0)}$ باید بروز شود:

$$w_{(1)} = w_{(0)} + \{x_{(3)}\} = (-0.75,-0.75,-0.75,-0.75,-0.75)$$

مجددا به محاسبه X_m پرداخته و می بینیم که $X_m(w_{(1)}) = \{x_{(1)},x_{(2)},x_{(4)}\}$، در نتیجه $w_{(1)}$ باید بروز شود:

$$w_{(2)} = w_{(1)} + (x_{(1)} + x_{(2)} + x_{(4)}) = (0.25,2.25,0.25,-1.75,-1.75)$$

مجددا به محاسبه X_m پرداخته و می‌بینیم که تمامی بردارهای ویژگی درست دسته‌بندی شده‌اند. در نتیجه $w = w_{(2)}$.

❐

در حل مسایل بهینه‌سازی با الگوریتم کاهش گرادیان بی‌نهایت جواب، بسته به مقدار اولیه انتخاب شده و نرخ یادگیری، می‌تواند بدست آید.

دقت کنید که بی‌نهایت جواب دیگر نیز برای w در مثال قبل موجود می‌باشند. جواب نهایی بدست آمده کاملا بستگی به مقدار اولیه انتخاب شده و نرخ یادگیری دارد.

به جای محاسبه X_m و بروز رسانی w با تمامی داده‌های اشتباه دسته‌بندی شده (بروز رسانی دسته‌ای)، می‌توان از بروز رسانی تک نمونه‌ای استفاده نمود. در بروز رسانی تک نمونه‌ای با مشاهده هر بردار ویژگی اشتباه دسته‌بندی شده، w بروز می‌شود. در پرسپترون، قانون بروز رسانی تک نمونه‌ای معادل قانون دسته‌ای فوق، به صورت $w_{(k+1)} = w_{(k)} + \eta_{(k)} x_i$ در می‌آید.

قانون بروز رسانی تک نمونه‌ای در مقایسه با قانون بروز رسانی دسته‌ای

در بروز رسانی تک نمونه‌ای، بردارهای ویژگی را یکی یکی پشت سر هم با شبکه عصبی موجود دسته‌بندی کرده و با رسیدن به هر دسته‌بندی اشتباه، مقدار w را با توجه به همان بردار ویژگی بروز کنیم. پس از اینکه تمامی داده‌ها با شبکه، مورد آزمایش قرار گرفتند، اگر هنوز w بهینه به دست نیامده بود، برای بار دوم بردارهای ویژگی را با شبکه و با آخرین w بدست آمده، مجددا مورد آزمون قرار می‌دهیم و همین روند را تا رسیدن به جواب بهینه ادامه می‌دهیم.

تمرین ٦: یک مساله کلاسه بندی دو کلاسه، در فضای دو بعدی، را با بردارهای ویژگی زیر در نظر بگیرید:

$$C_1 = \{(2,1),(4,3),(3,5)\} \qquad C_2 = \{(1,3),(5,6)\}$$

آیا الگوریتم کاهش گرادیان متوقف می‌شود؟ چرا؟ بردارهایی را که در مراحل مختلف بدست می‌آیند رسم کنید. آیا این بردارها دارای رفتار یا ویژگی خاص مشترکی هستند؟ با چه شرطی الگوریتم را متوقف کنیم که جواب مناسبی بدست آید؟

تمرین ۷: الگوریتم کاهش گرادیان با قانون بروز رسانی دسته‌ای سریع‌تر به جواب نزدیک می‌شود یا با قانون بروز رسانی تک نمونه‌ای؟ کدام یک در مواجهه با نویزهای تکین (تک داده‌های دور افتاده) موفق‌تر عمل می‌کند؟

تمرین ۸: نشان دهید که اگر $x_i \in X_m(w_{(k)})\ ;\ \eta_{(k)} > \left|w_{(k)}^t x_i\right| / \left|x_i\right|^2$ باشد قطعا $w_{(k+1)}^t x_i$ منفی نمی‌شود.

۱.۱- معیار Relaxation برای آموزش نرون

معیار دیگری که برای آموزش یک نرون می‌تواند مورد استفاده قرار گیرد، معیار Relaxation به صورت زیر می‌باشد.

$$J(w)=\frac{1}{2}\sum_{x\in X_b(w)}\frac{(w^t x-b)^2}{|x|^t};\quad X_b(w)=\{samples\ x_{(i)}\,|\,w^t x_{(i)}\le b\}$$

در این معیار، نه تنها بردارهای ویژگی اشتباه دسته‌بندی شده در تصحیح w شرکت خواهند داشت، بلکه بردارهای ویژگی درست دسته‌بندی شده ولی نزدیک به مرز (نزدیک‌تر از $b/\|w\|$) نیز در تصحیح شرکت خواهند داشت. در این روش تلاش بر این است که حاشیه‌ای با طول $b/\|w\|$ در اطراف مرز جداساز وجود داشته باشد که قدرت تعمیم جداساز را در مواجهه با بردارهای ویژگی جدید بالا ببرد. قانون بروز رسانی دسته‌ای برای این معیار به صورت زیر در خواهد آمد:

قوانین بروز رسانی دسته‌ای و تک نمونه‌ای برای معیار Relaxation

$$w_{(k+1)}=w_{(k)}+\eta_{(k)}\sum_{x\in X_b(w)}\frac{b-w_{(k)}^t x}{|x|^2}x$$

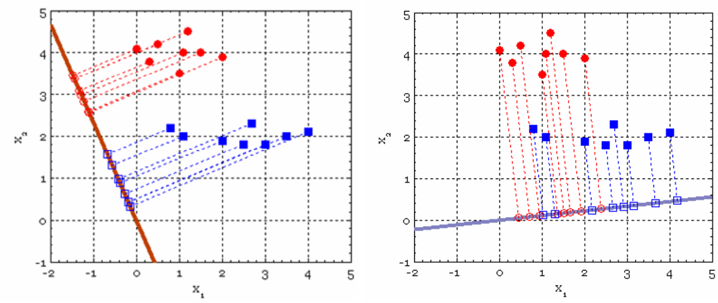
قانون بروزرسانی تک نمونه‌ای نیز برابر خواهد بود با:

$$w_{(k+1)}=w_{(k)}-\eta_{(k)}\frac{w_{(k)}^t x}{|x|^2}x$$

۲- جداساز خطی فیشر

در جداساز خطی فیشر داده‌ها در جهتی تابانده می‌شوند که بیشترین جداسازی را داشته باشند

ایده اصلی در جداساز خطی فیشر این است که داده‌ها در جهتی تابانده[8] شوند که بیشترین جداسازی را داشته باشند و براحتی بتوان مرزی برای داده‌های تابیده شده مشخص نمود (کاهش ابعاد نیز انجام می‌شود). به عنوان مثال در شکل ۷ حاصل تابش در (ب) دارای جداسازی بسیار بیشتری از حاصل تابش در (الف) می‌باشد.



شکل ۷: تابش داده‌ها در جهت‌های مختلف. جداسازی در تابش (ب) بسیار بیشتر از جداسازی در تابش (الف) می‌باشد.

برای داده‌های معرفی شده در شکل ۷، جهت‌های مختلفی وجود دارد که با تابش در آن جهت‌ها، می‌توان به جداسازی داده‌ها، بدون خطا، اقدام نمود. حال سوال پیش می‌آید که کدام یک از این جهت‌ها بهترین جهت ممکن می‌باشد و چگونه می‌توان این جهت را پیدا نمود؟

در ادامه به بررسی دو سوال فوق با یک مثال دسته‌بندی دوکلاسه در فضای d بعدی می‌پردازیم. فرض کنید در این فضا دارای n بردار ویژگی آموزشی $(x_1,x_2,\ldots,x_n)$ باشیم: n_1 بردار ویژگی از کلاس اول و n_2 بردار ویژگی از کلاس دوم.

مفهوم هندسی تابش

اگر بردار ویژگی x در جهت بردار w تابیده شود، از لحاظ هندسی حاصل تابش محل تقاطع خط عمود بر بردار w از نقطه x خواهد بود (به شکل ۷ توجه کنید). فاصله این نقطه از مرکز مختصات، $w^t x$ خواهد بود. با این کار بردار ویژگی از یک فضای دو بعدی به یک فضای یک بعدی نگاشت شده است.

[8] Project

جهت تابش با بالاترین جداسازی

حال با این مقدمه به سوال اول بر می‌گردیم. چه جهتی بهترین جهت تابش بوده و داده‌ها بعد از تابش در راستای آن، به بهترین شکل قابل جداسازی خواهند بود؟ ممکن است جواب افراد مختلف به این سوال متفاوت و حتی در برخی موارد متناقض باشد. یکی از پاسخ‌ها می‌تواند این باشد که بهترین جهت، جهتی است که در آن، داده‌های دو کلاس متمایز بیشترین فاصله را از هم داشته باشند، در حالی که داده‌های هر کلاس بیشترین نزدیکی و مقاربت را به یکدیگر داشته باشند.

پاسخ ارائه شده که توسط فیشر ارائه شده، منطقی و معقول می‌باشد، ولی این جهت به چه صورت قابل شناسایی می‌باشد؟ (سوال دوم) برای پاسخ به این سوال ناچاریم مفاهیم کیفی ارائه شده در پاسخ سوال اول را کمی کنیم. فرض کنید μ_1 و μ_2 میانگین کلاس‌های C_1 و C_2 بوده و $\hat{\mu}_1$ میانگین داده‌های تابیده شده کلاس C_1 و $\hat{\mu}_2$ میانگین داده‌های تابیده شده کلاس C_2 باشند. در این صورت می‌توان فاصله داده‌های دو کلاس را با $d=|\hat{\mu}_1-\hat{\mu}_2|$ کمی کرد. داریم:

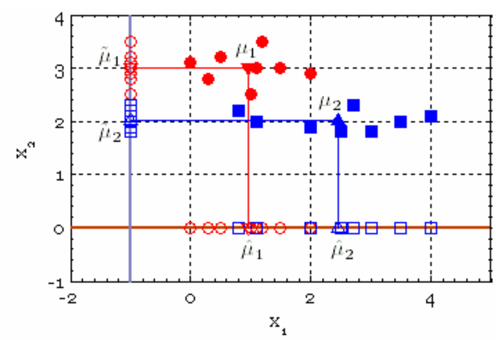
$$\mu_1=\frac{1}{n_1}\sum_{x_i\in C_1}x_i,\ \mu_2=\frac{1}{n_2}\sum_{x_i\in C_2}x_i$$

و اگر راستای تابش، w باشد، داریم:

$$\hat{\mu}_1=\frac{1}{n_1}\sum_{x_i\in C_1}w^t x_i=w^t\left(\frac{1}{n_1}\sum_{x_i\in C_1}x_i\right)=w^t\mu_1$$

و به همین ترتیب $\hat{\mu}_2=w^t\mu_2$. در نتیجه $d=|\hat{\mu}_1-\hat{\mu}_2|=|w^t\mu_1-w^t\mu_2|=w^t|\mu_1-\mu_2|$.

قبلا توضیح ندادیم که چرا این معیار به تنهایی معیار مناسبی نیست. ولی اکنون با کمی‌سازی برخی از مفاهیم می‌توانیم این سوال را راحت‌تر جواب دهیم. برای این منظور به داده‌های دو کلاس ارائه شده در شکل ۸ توجه نمائید. داده‌های نشان داده شده، دو بار در دو جهت افقی و عمودی تابانده شده‌اند. $\hat{\mu}_1,\hat{\mu}_2$ میانگین داده‌های دو کلاس تابانده شده در جهت افقی و $\tilde{\mu}_1,\tilde{\mu}_2$ میانگین داده‌های دو کلاس تابانده شده در جهت عمودی می‌باشند.



شکل ۸: تابش داده‌ها در دو جهت متفاوت برای سنجش معیار «فاصله بین کلاس‌ها»

همانطور که مشاهده می‌کنید، $|\hat{\mu}_1-\hat{\mu}_2|$ بزرگتر از $|\tilde{\mu}_1-\tilde{\mu}_2|$ است ولی جداسازی داده‌های تابیده شده در جهت افقی راحت‌تر از جداسازی داده‌های تابیده در جهت عمودی نیست (می‌توان بین داده‌های دو کلاس تابیده شده در جهت عمودی مرزی بدون خطا رسم کرد ولی برای داده‌های تابیده شده در جهت افقی این کار امکان‌پذیر نیست). دلیل این امر، همانطور که فیشر نیز بیان می‌کند، در نظر نگرفتن فواصل بین کلاسی داده‌های تابیده شده در دو جهت می‌باشد. فواصل بین کلاسی داده‌ها، یا پراکندگی آنها، را می‌توان با مجموع فواصل داده‌های هر کلاس از میانگین آن کلاس، کمی نمود:

$$S_1=\sum_{x_i\in C_1}(x_i-\mu_1)(x_i-\mu_1)^T,\ S_2=\sum_{x_i\in C_2}(x_i-\mu_2)(x_i-\mu_2)^T$$

برای محاسبه پراکندگی داده‌های تابیده شده نیز داریم:

$$\begin{aligned}\hat{S}_1&=\sum_{\hat{x}_i\in C_1}(\hat{x}_i-\hat{\mu}_1)^2=\sum_{x_i\in C_1}(w^t x_i-w^t\mu_1)^2=\sum_{x_i\in C_1}\left(w^t(x_i-\mu_1)\right)^2=\sum_{x_i\in C_1}\left(w^t(x_i-\mu_1)\right)^t\left(w^t(x_i-\mu_1)\right)\\&=\sum_{x_i\in C_1}\left((x_i-\mu_1)^t w\right)^t\left((x_i-\mu_1)^t w\right)=\sum_{x_i\in C_1}w^t(x_i-\mu_1)(x_i-\mu_1)^t w=w^t S_1 w\end{aligned}$$

و به همین ترتیب $\hat{S}_2 = w^t S_2 w$. هر چه داده‌های یک کلاس متمرکزتر باشند، پراکندگی آن کلاس کمتر است و بالعکس.

تابع هدف فیشر

مطابق معیار فیشر و با توجه به تعاریف میانگین و پراکندگی، بهترین جهت از دید فیشر، جهتی است که میانگین داده‌های تابیده شده کلاس‌های مختلف، بیشترین فاصله را از هم داشته باشند و در حین حال پراکندگی داده‌های تابیده شده مربوط به هر کلاس کمترین مقدار ممکن باشد. و این یعنی اینکه $|\hat{\mu}_1 - \hat{\mu}_2|$ (یا معادل آن $((\hat{\mu}_1 - \hat{\mu}_2)^2$) بیشینه و $\hat{S}_1 + \hat{S}_2$ کمینه شود. پس باید بدنبال جهتی باشیم که تابع هدف زیر را بیشینه کند:

$$J(w) = \frac{(\hat{\mu}_1 - \hat{\mu}_2)^2}{\hat{S}_1 + \hat{S}_2} \qquad (۹)$$

با بسط این رابطه و جایگزینی روابط مربوط به $\hat{\mu}_1$، $\hat{\mu}_2$، $\hat{S}_1$ و $\hat{S}_2$ خواهیم داشت:

$$J(w) = \frac{(\hat{\mu}_1 - \hat{\mu}_2)^2}{\hat{S}_1 + \hat{S}_2} = \frac{w^t(\mu_1 - \mu_2)(\mu_1 - \mu_2)^t w}{w^t S_1 w + w^t S_2 w} = \frac{w^t(\mu_1 - \mu_2)(\mu_1 - \mu_2)^t w}{w^t(S_1 + S_2)w}$$

که اگر $S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^t$ و $S_w = S_1 + S_2$ در نظر گرفته شوند، خواهیم داشت:

$$J(w) = \frac{w^t S_B w}{w^t S_w w}$$

برای اینکه $J(w)$ بیشینه شود باید داشته باشیم:

$$\frac{d}{dw} J(w) = \frac{(2 S_B w) w^t S_W w - (2 S_W w) w^t S_B w}{(w^t S_W w)^2} = 0$$

برای این منظور کافی است صورت کسر فوق مساوی صفر قرار داده شود، که با تقسیم صورت بر $w^t S_W w$ خواهیم داشت:

$$S_B w - \frac{w^t S_B w}{w^t S_W w} S_W w = 0 \Rightarrow S_B w = \frac{w^t S_B w}{w^t S_W w} S_W w$$

و این یعنی،

$$S_B w = J(w).S_W w \qquad (۱۰)$$

$J(w)$ یک مقدار عددی بوده و می‌تواند با λ جایگزین شود. در نتیجه $S_W^{-1} S_B w = \lambda w$. از طرفی می‌دانیم که حاصلضرب S_B در هر برداری، راستایی در جهت $(\mu_1 - \mu_2)$ خواهد داشت. زیرا:

$$S_B y = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^t y = (\mu_1 - \mu_2)\left((\mu_1 - \mu_2)^t y\right) = \alpha.(\mu_1 - \mu_2)$$

جهت تابش فیشر

پس راستای w برابر خواهد بود با:

$$\overrightarrow{w} = S_W^{-1}(\mu_1 - \mu_2) \qquad (۱۱)$$

ما در اینجا به جای حل دقیق رابطه (۱۰) از حل شهودی مساله برای یافتن جهت بردار w استفاده کردیم. برای حل دقیق این مساله می‌توان از مساله مقادیر ویژه تعمیم یافته استفاده نمود. زیرا این رابطه، دقیقا دارای صورت یک مساله مقادیر ویژه تعمیم یافته می‌باشد.

مساله مقادیر ویژه تعمیم یافته

برای دو ماتریس داده شده A و B مساله پیدا کردن جفت مقدار (λ, x) بطوری که $Ax = \lambda Bx$، را مساله مقادیر ویژه تعمیم یافته می گویند.

تمرین ۹: با استفاده از نحوه حل مقادیر ویژه تعمیم یافته، رابطه (۱۰) را بطور دقیق حل نمائید.

مثال ٦: یک مساله دسته‌بندی در فضای دو بعدی با دو کلاس و بردارهای ویژگی زیر در نظر بگیرید:

$$c_1=\{(1,2),(2,3),(3,3),(4,5),(5,5)\}\ ,\ \ c_2=\{(1,0),(2,1),(3,1),(3,2),(5,3),(6,5)\}$$

جهت فیشر را برای این داده‌ها پیدا کنید.

ابتدا میانگین داده‌ها را بدست می‌آوریم:

$$\mu_1=\sum_{x_i\in C_1}x_i=(3,3.6),\ \ \mu_2=\sum_{x_i\in C_2}x_i=(3.3,2)$$

سپس پراکندگی هر کدام از کلاس‌ها را محاسبه می‌کنیم:

$$S_1=\sum_{x_i\in C_1}(x_i-\mu_1)(x_i-\mu_1)^T=\begin{bmatrix}10 & 8.0\\ 8.0 & 7.2\end{bmatrix},\ \ S_2=\sum_{x_i\in C_2}(x_i-\mu_2)(x_i-\mu_2)^T=\begin{bmatrix}17.3 & 16\\ 16 & 16\end{bmatrix}$$

در نتیجه:

$$S_w=S_1+S_2=\begin{bmatrix}27.3 & 24\\ 24 & 23.2\end{bmatrix},\ S_w^{-1}=\begin{bmatrix}0.39 & -0.41\\ -0.41 & 0.47\end{bmatrix}$$

و نهایتا اینکه:

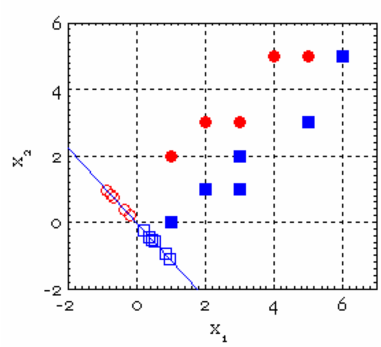
$$w=S_w^{-1}(\mu_1-\mu_2)=\begin{bmatrix}-0.79\\ 0.89\end{bmatrix}$$

مختصات داده‌های جدید (تابیده شده) بر روی محور تابش عبارتند از:

$$y_{1i}=w^t x_{1i}\ \Rightarrow\ Y=\{y_{11},y_{12},\ldots,y_{15}\}=\{0.99,1.08,0.29,1.28,0.48\}$$

$$y_{2i}=w^t x_{2i}\ \Rightarrow\ Y=\{y_{21},y_{22},\ldots,y_{26}\}=\{-0.79,-0.7,-1.49,-0.6,-1.3,-0.31\}$$

داده‌های دو کلاس، خط تابش و داده‌های تابیده شده، در شکل زیر نشان داده شده‌اند.



❐

فیشر و فضاهای چند بعدی

جداساز خطی فیشر (FLD[9]) را برای حالت دو بعدی بررسی کردیم. حال می‌خواهیم ببینیم که جداساز فیشر برای حالت چند بعدی به چه صورت خواهد بود؟ اگر به مرور محاسبات انجام شده تا بدست آوردن رابطه (١١) بپردازید، می‌بینید که تمامی معادلات و روابط نوشته شده، مستقل از تعداد ابعاد داده‌ها، برقرار بوده و صحیح می‌باشند. در نتیجه رابطه (١١) برای ابعاد بالاتر نیز برقرار بوده و برای هر فضای d بعدی، برداری d بعدی بدست می‌آید.

فیشر و مسایل چند کلاسه

جداساز خطی فیشر را برای حالت دو کلاسه بررسی کردیم. حال می‌خواهیم جداساز فیشر را برای حالت چند کلاسه بررسی کنیم. برای این منظور در ساده‌ترین حالت می‌توان از ماشین خطی کامل (با $k-1$ جهت فیشر مستقل از هم) و یا ماشین خطی دوبدو جداپذیر (با $k(k-1)/2$ جهت فیشر مستقل از هم) استفاده نمود.

[9] Fisher Linear Discriminant

می‌توان به جای تبدیل مساله چند کلاسه به مسایل دو کلاسه کوچکتر مستقل از هم و حل این مسایل کوچک‌تر، روشی را برای حل یک جای مساله نیز ارائه نمود. این روش‌ها با عنوان تحلیل جداسازهای چندتایی (MDA[10]) مشهورند.

برای حالت تعمیم یافته فیشر، یک مساله دسته‌بندی را با k کلاس، در یک فضای d بعدی در نظر بگیرید. می‌خواهیم ماتریس فیشر (w) را پیدا کنیم به طوری که هر بردار ویژگی x_i را تحت رابطه $y_i = w^t x_i$، به بردار ویژگی y_i تبدیل کرده و این تبدیل به صورتی باشد که در فضای تابش بدست آمده، پراکندگی بین‌کلاسی داده‌ها، بیشینه و مجموع پراکندگی داده‌های داخل کلاسی، کمینه باشد.

اگر تعداد بردارهای ویژگیی کلاس iام را با n_i نشان دهیم و مجموع تمامی بردارهای ویژگی برابر n باشد، پراکندگی کلی داده‌ها (S_T)، مجموع پراکندگی داده‌های داخل کلاسی (S_W) و پراکندگی بین‌کلاسی داده‌ها (S_B) در فضای تابش برابر خواهد بود با:

$$S_T = \sum_{i=1}^{n}(y_i - m)(y_i - m)^t\,,\; m = \frac{1}{n}\sum_{i=1}^{n} y_i = \frac{1}{n}\sum_{i=1}^{k} n_i m_i\,,\; m_i = \frac{1}{n_i}\sum_{y_j \in C_i} y_j$$

$$S_W = \sum_{i=1}^{k} S_i\,,\; S_i = \sum_{y_j \in C_i}(y_j - m_i)(y_j - m_i)^t$$

$$S_T = S_W + S_B \Rightarrow S_B = S_T - S_W = \sum_{i=1}^{k} n_i(m_i - m)(m_i - m)^t$$

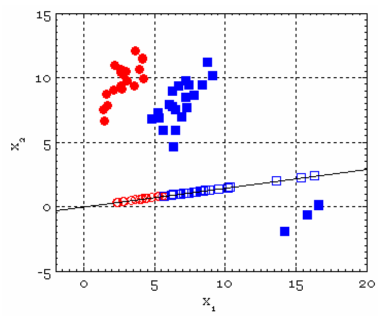
معیار بهینه‌سازی نیز به صورت برابر خواهد بود با:

$$J(w) = \frac{\det(wS_B w^t)}{\det(wS_W w^t)} \qquad (۱۲)$$

رابطه ۰ نشان می‌دهد که S_B مجموع k ماتریس از درجه یک می‌باشد و فقط k-1 از آنها مستقل می‌باشند (kامین ماتریس با توجه به این k-1 ماتریس قابل بدست آوردن است. چگونه؟). در نتیجه S_B از درجه k-1 بوده و دارای حداکثر k-1 مقدار ویژه غیر صفر می‌باشد. این نشان می‌دهد که تابش داده‌ها در فضای k-1 بعدی حاصل از بردارهای ویژه S_B تغییری در مقدار $J(w)$ نخواهد داد. و این یعنی اینکه با این روش فضایی با حداکثر k-1 بعد به عنوان فضای نگاشت شده خواهیم داشت. پس در حالت کلی جداساز خطی فیشر در یک مساله دسته‌بندی k کلاسه، فضا را به k-1 بعد کاهش می‌دهد..

استفاده از جداساز خطی فیشر همیشه مطلوب نیست.

استفاده از جداساز خطی فیشر همیشه مطلوب نیست. در جداسازی فیشر فرض می‌شود که بردارهای ویژگی تابیده شده دارای توزیع نرمال می‌باشند (چرا؟). اگر این فرض بر قرار نباشد، جداساز فیشر، جهت مناسبی را برای تابش بر نمی‌گرداند (شکل ۹).

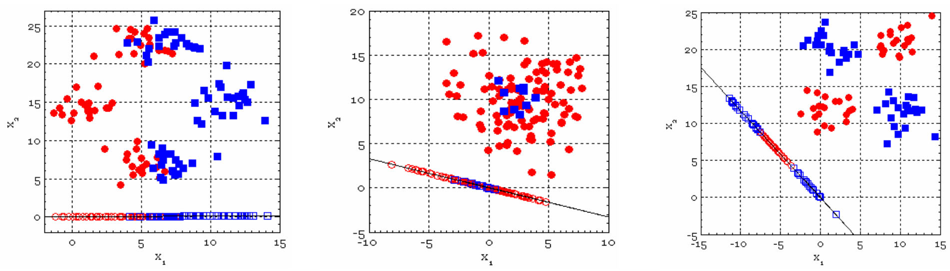


شکل ۹: جداساز فیشر وقتی که بردارهای ویژگی تابیده شده دارای توزیع نرمال نباشند، جهت مناسبی را بر نمی‌گرداند.

جداساز خطی فیشر، در حالاتی که $J(w)$ صفر بوده یا همیشه دارای مقدار بزرگی باشد، با شکست مواجه شده و قادر نیست جواب مسایل را پیدا کند. $J(w) = 0$ در حالتی رخ می‌دهد که میانگین دو کلاس مساوی باشند (حالت‌های الف و ب از شکل ۱۰). اگر داده‌ها در هر جهتی که تابانده شوند، دارای تداخل زیادی باشند، نیز حالتی است که $J(w)$ همیشه مقادیر کوچکی را بر می‌گرداند (حالت پ از شکل ۱۰). در شکل‌ها توجه کنید که جهت‌های فیشر بدست آمده نامناسب بوده و خطای بالایی را در

[10] Multiple Discriminant Analysis

جداسازی بوجود می‌آورند.

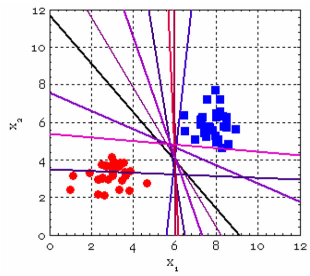


(پ) (ب) (الف)

شکل ۱۰: حالات نامناسب برای استفاده از جداساز خطی فیشر، با جهت‌های نامناسب بدست آمده.

۳- جداسازهای با بیشترین حاشیه

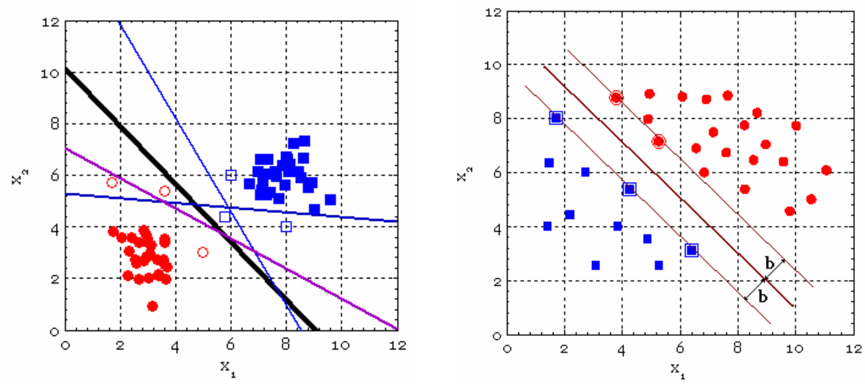
برای هر مساله دسته‌بندی بصورت خطی جداپذیر می‌توان بی‌نهایت ابر صفحه ارائه نمود که به عنوان جداساز عمل نمایند (شکل ۱۱). سوالی که پیش می‌آید این است که کدام یک از این جداسازها بهترین می‌باشد؟



شکل ۱۱: بهترین جداساز کدام است؟

جداساز با بیشترین حاشیه، دارای کمترین خطا در تعمیم خواهد بود.

ایده اصلی در جداسازهای با بیشترین حاشیه[11] همانطور که از نامشان پیداست این است که ابرصفحه‌هائی را با بیشترین حاشیه بدست آورند که کلاس‌ها را از هم جدا کند. حاشیه به فاصله‌ای گفته می‌شود که دو خط موازی جداساز بطور مساوی از دو طرف طی می‌کنند، تا یکی از آن دو به یکی از داده‌ها برخورد نماید (شکل ۱۲). از بین جداسازهای خطی، جداسازی که بیشترین حاشیه را داشته باشد (بیشترین فاصله را از داده‌ها داشته باشد)، خطای تعمیم را حداقل خواهد کرد (شکل ۱۳).



(شکل ۱۳) (شکل ۱۲)

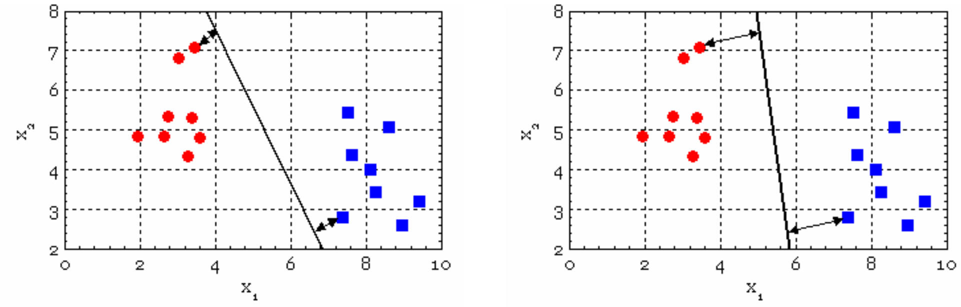
شکل ۱۲: یک مجموعه داده با مرز جداسازی. حاشیه جداساز با ضخامت b نیز نشان داده شده است..

شکل ۱۳: جداسازهای با حاشیه بیشتر، در هنگام تعمیم خطای کمتری خواهند داشت. اشکال توپر داده‌های آموزشی و اشکال توخالی داده‌های جدید (آزمایشی) هستند. به رابطه عملکرد صحیح در مواجهه با داده‌های جدید و اندازه حاشیه توجه کنید.

[11] Maximum Margin

پیدا کردن جداساز با بیشترین حاشیه، چگونه؟

حال که مفهوم حاشیه و جداساز با بیشترین حاشیه مشخص شد باید به دنبال راهی برای پیدا کردن جداساز با بیشترین حاشیه باشیم. ایده اصلی برای انجام این کار این است که سعی کنیم جداسازی را بیابیم که فاصله آن از نزدیک‌ترین داده بیشینه باشد. از آنجایی که در دو طرف یک خط (یا در حالت کلی یک ابر صفحه) داده‌های دو کلاس قرار دارند، فاصله گرفتن خط از داده‌های یکی از کلاس‌ها مترادف خواهد بود با نزدیک شدن خط به داده‌های کلاس دیگر. در نتیجه، خط (ابر صفحه) بهینه در حالتی بدست خواهد آمد که فاصله خط جداساز از نزدیک‌ترین داده‌های دو طرف مساوی و بیشینه باشد (شکل ۱۴).



شکل ۱۴: جداسازی بیشترین حاشیه را خواهد داشت که فاصله آن از نزدیک‌ترین داده‌های دو طرف مساوی و بیشینه باشد (الف).

پیدا کردن بردارهای پشتیبان مستلزم شناخت جداساز با بیشرین حاشیه می‌باشد و پیدا کردن جداساز با بیشترین حاشیه نیز مستلزم شناخت بردارهای پشتیبان است!

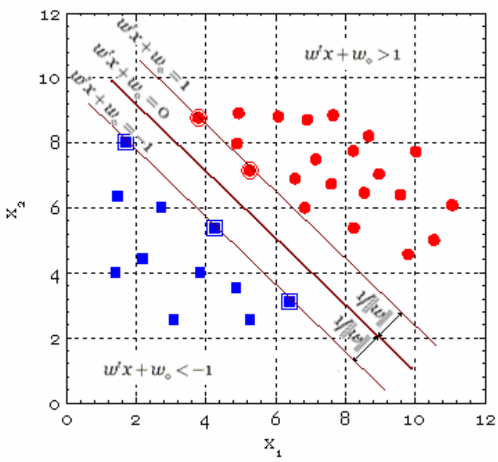
نزدیک داده‌های دو طرف، به جداساز با بیشترین حاشیه را بردارهای پشتیبان می‌نامند. پیدا کردن بردارهای پشتیبان کار مشکلی است، چون پیدا کردن آنها مستلزم شناخت جداساز با بیشرین حاشیه می‌باشد و پیدا کردن جداساز با بیشترین حاشیه نیز مستلزم شناخت بردارهای پشتیبان است! در ادامه سعی می‌کنیم با شناخت بیشتر از مساله و فرموله کردن و ساده‌سازی آن بر این حلقه غلبه کنیم.

جداساز با بیشترین حاشیه یک جداساز خطی بوده و همانطور که قبلا مشاهده کردیم دارای رابطه $g(x)=w^t x+w_0=0$ می‌باشد. فاصله هر داده x از این ابر صفحه برابر است با:

$$d=\frac{|g(x)|}{\|w\|}=\frac{|w^t x+w_0|}{\|w\|}$$

فرض کنید x_i بردار پشتیبان بوده و تابع جداساز $g(x)$ نیز، با تغییر اندازه w و w_0، طوری تنظیم شده باشد که:

$$|g(x_i)|=|w^t x_i+w_0|=1$$



شکل ۱۵: رابطه جداساز و حاشیه‌ها

در این صورت فاصله x_i تا جداساز، $1/\|w\|$، اندازه حاشیه، $2/\|w\|$ و معادلات خطوط حاشیه، $w^t x+w_0=\pm 1$ خواهند بود (شکل ۱۵). صورت مساله نیز به این صورت درمی‌آید: برای $g(x)=w^t x+w_0$ جفت مقدار (w,w_0) را طوری بیابید که

$2/\|w\|$ بیشینه بوده و برای داده‌های آموزشی داده شده، داشته باشیم:

$$\begin{cases} g(x) \geq 1 & if\ x \in C_1 \\ g(x) \leq -1 & if\ x \in C_2 \end{cases}$$

با تعریف متغیر z بصورت

$$\begin{cases} z = 1 & if\ x \in C_1 \\ z = -1 & if\ x \in C_2 \end{cases}$$

مساله بهینه‌سازی معادل پیدا کردن جدا ساز با بیشترین حاشیه

رابطه فوق به صورت $z.g(x) \geq 1$ در می‌آید. همچنین بیشینه کردن $2/\|w\|$ معادل کمینه‌سازی $\frac{1}{2}\|w^2\|$ می‌باشد. با این تغییرات مساله به یک مساله استاندارد بهینه‌سازی با قیدهای نامساوی، تبدیل خواهد شد:

$$\begin{aligned} &\text{minimize } J(w) = \frac{1}{2}\|w^2\| \\ &\text{subject to } z_i.(w^t x_i + w_0) - 1 \geq 0, \quad \forall i \end{aligned} \qquad (۱۳)$$

مسایل بهینه‌سازی، ضرایب لاگرانژ و شرایط KKT

یک مساله بهینه‌سازی با قیدهای نامساوی دارای حالت کلی زیر می‌باشد:

$$\begin{aligned} &\text{minimize } J(\theta) \\ &\text{subject to } f_i(\theta) \geq 0, \quad i = 1,2,\ldots,m \end{aligned}$$

لاگرانژ نشان داد که می‌توان قیدها را در تابع اصلی بهینه شونده ادغام نموده و مساله را به صورت زیر در آورد:

$$\text{minimize } L(\theta,\lambda), \quad L(\theta,\lambda) = J(\theta) - \sum_{i=1}^{m} \lambda_i f_i(\theta)$$

که در آن λ را ضرایب لاگرانژ گویند که مجهول بوده و طوری باید بدست آیند که تابع $L(\theta,\lambda)$ را بیشینه نمایند. پس خواهیم داشت:

$$\min_\theta J(\theta) = \min_\theta \max_\lambda L(\theta,\lambda) = \max_\lambda \min_\theta L(\theta,\lambda)$$

همچنین نشان داده شده است برای اینکه θ^* یک کمینه‌ساز محلی برای مساله فوق باشد، باید شرایط زیر را دارا باشد (شرایط Karush-kuhn-Tucker):

$$\begin{aligned} &(1)\ \frac{\partial}{\partial\theta} L(\theta^*,\lambda) = 0 \\ &(2)\ \lambda_i \geq 0, \quad i = 1,2,\ldots,m \\ &(3)\ \lambda_i f_i(\theta^*) = 0, \quad i = 1,2,\ldots,m \end{aligned}$$

صورت لاگرانژی مساله و بیان شرایط KKT برای آن

صورت لاگرانژی این مساله به فرم زیر می‌باشد:

$$L(w,w_0,\lambda) = \frac{1}{2} w^t w - \sum_{\forall i} \lambda_i \left[z_i.(w^t x_i + w_0) - 1 \right] \qquad (۱۴)$$

که برای (w, w_0) بهینه، باید شرایط زیر (شرایط KKT) برقرار باشد:

$$\begin{aligned} &(1)\ \frac{\partial}{\partial w} L(w,w_0,\lambda) = 0, \quad \frac{\partial}{\partial w_0} L(w,w_0,\lambda) = 0 \\ &(2)\ \lambda_i \geq 0, \quad \forall_i \\ &(3)\ \lambda_i [z_i(w^t x_i + w_0) - 1] = 0, \quad \forall i \end{aligned} \qquad (۱۵)$$

با ترکیب روابط (۱۴) و (۱۵) خواهیم داشت:

$$\frac{\partial}{\partial w}L(w,w_0,\lambda)=w-\sum_{\forall i}\lambda_i z_i x_i=0 \;\Rightarrow\; w=\sum_{\forall i}\lambda_i z_i x_i$$
$$\frac{\partial}{\partial w_0}L(w,w_0,\lambda)=-\sum_{\forall i}\lambda_i z_i=0 \;\Rightarrow\; \sum_{\forall i}\lambda_i z_i=0$$

پس صورت نهایی مساله با استفاده از ضرایب لاگرانژ بصورت زیر در می‌آید:

$$\begin{aligned}&\max_{\lambda}\; L(w,w_0,\lambda)\\&\text{subject to:}\\&\qquad (1)\; w=\sum_{\forall i}\lambda_i z_i x_i \qquad (۱۶)\\&\qquad (2)\; \sum_{\forall i}\lambda_i z_i=0\\&\qquad (3)\; \lambda\ge 0\end{aligned}$$

با ترکیب قید (۱) از رابطه (۱۶) در تابع $L(w,w_0,\lambda)$ خواهیم داشت:

$$\begin{aligned}L(w,w_0,\lambda)&=\frac{1}{2}w^t w-\sum_{\forall i}\lambda_i\left[z_i.(w^t x_i+w_0)-1\right]\\&=\frac{1}{2}\sum_{\forall i}\lambda_i z_i x_i^t\sum_{\forall j}\lambda_j z_j x_j-\sum_{\forall i}\lambda_i z_i\sum_{\forall j}\lambda_j z_j x_j^t x_i-w_0\sum_{\forall i}\lambda_i z_i+\sum_{\forall i}\lambda_i \qquad (۱۷)\\&=-\frac{1}{2}\sum_{\forall i}\lambda_i\lambda_j z_i z_j x_i^t x_j-w_0\sum_{\forall i}\lambda_i z_i+\sum_{\forall i}\lambda_i\end{aligned}$$

و با توجه به قید (۲) از رابطه (۱۶) و همچنین توجه به این نکته که در رابطه (۱۷) (w,w_0) ظاهر نشده‌اند، رابطه (۱۷) به صورت زیر در می‌آید:

$$L(\lambda)=-\frac{1}{2}\sum_{\forall i}\sum_{\forall j}\lambda_i\lambda_j z_i z_j x_i^t x_j+\sum_{\forall i}\lambda_i \qquad (۱۸)$$

صورت نهایی مساله بهینه‌سازی معادل با پیدا کردن جداساز با بیشترین حاشیه

در نتیجه صورت نهایی مساله با استفاده از ضرایب لاگرانژ بصورت زیر ساده می‌شود:

$$\begin{aligned}&\max_{\lambda}\; L(\lambda)=\sum_{\forall i}\lambda_i-\frac{1}{2}\sum_{\forall i}\sum_{\forall j}\lambda_i\lambda_j z_i z_j x_i^t x_j\\&\text{subject to:} \qquad\qquad (۱۹)\\&\qquad (1)\; \sum_{\forall i}\lambda_i z_i=0\\&\qquad (2)\; \lambda\ge 0\end{aligned}$$

$L(\lambda)$ را می‌توان بصورت ماتریسی نیز بیان نمود:

$$L(\lambda)=\sum_{\forall i}\lambda_i-\frac{1}{2}\begin{bmatrix}\lambda_1\\ \vdots\\ \lambda_n\end{bmatrix}^t H\begin{bmatrix}\lambda_1\\ \vdots\\ \lambda_n\end{bmatrix}=\sum_{\forall i}\lambda_i-\frac{1}{2}\lambda^t H\lambda \qquad (۲۰)$$

که در آن $\lambda=\left[\lambda_1\cdots\lambda_n\right]^t$ و $H_{ij}=z_i z_j x_i^t x_j$.

بیشینه‌سازی $L(\lambda)$ می‌تواند با روش‌های تکراری محاسبات عددی (نظیر برنامه‌ریزی درجه دوم) حل شود.

مشخصات بردارهای پشتیبان، مرز جداسازی و تابع جداساز با بیشترین حاشیه

با حل این مساله، مقادیر λ_i ها بدست می‌آیند. تمامی بردارهای ویژگی‌ای که به ازای آنها λ_i غیر صفر باشد، بردارهای پشتیبان می‌باشند (چرا؟). w از رابطه $w=\sum_{\forall i}\lambda_i z_i x_i$ بدست می‌آید. w_0 نیز از هر کدام از (x_i,z_i) هایی که λ_i معادل آنها صفر نباشد، از رابطه $w_0=1/z_i-w^t x_i$ قابل حصول است.

تابع جداساز نیز بصورت

$$g(x)=\left(\sum_{x_i\in S}\lambda_i z_i x_i\right)^t x+w_0 \qquad (۲۱)$$

می‌باشد که در آن S مجموعه بردارهای پشتیبان می‌باشد، یعنی $S=\{x_i \mid \lambda_i \neq 0\}$.

مثال ۷: برای یک مساله دسته‌بندی دو کلاسه در فضای دو بعدی با بردارهای ویژگی زیر، جداساز با بیشترین حاشیه را بیابید.

$$C_1=\{(1,6),(1,10),(4,11)\},\ C_2=\{(5,2),(7,6),(10,4)\}$$

ماتریس‌های X و z برای این مثال بصورت زیر خواهند بود (۱ برای کلاس اول و ۱- برای کلاس دوم):

$$X^t=\begin{bmatrix} 1 & 1 & 4 & 5 & 7 & 10 \\ 6 & 10 & 11 & 2 & 6 & 4 \end{bmatrix},\ Z^t=\begin{bmatrix} 1 & 1 & 1 & -1 & -1 & -1 \end{bmatrix}$$

برای محاسبه ماتریس H از رابطه $H=XX^t.zz^t$ استفاده می‌کنیم (زیرا درایه این ماتریس $H_{ij}=z_i z_j x_i^t x_j$ می‌باشند):

$$H=\begin{bmatrix} 37 & 61 & 70 & -17 & -43 & -34 \\ 61 & 101 & 114 & -25 & -67 & -50 \\ 70 & 114 & 137 & -42 & -94 & -84 \\ -17 & -25 & -42 & 29 & 47 & 58 \\ -43 & -67 & -94 & 37 & 85 & 94 \\ -34 & -50 & -84 & 58 & 84 & 116 \end{bmatrix}$$

با داشتن ماتریس H باید بردار λ با اعداد مثبت را طوری بیابیم که $L(\lambda)=\sum_{i=1}^{6}\lambda_i-\frac{1}{2}\lambda^t H\lambda$ بیشینه شده و $Z^t\lambda$ برابر صفر شود. برای حل این مساله بهینه‌سازی از ابزارهای مربوطه می‌توان بهره برد. به عنوان مثال مطلب، قادر به حل مسائل بهینه‌سازی استاندارد (به فرم زیر) می‌باشد:

$$\text{minimize } L_D(\lambda)=\frac{1}{2}\lambda^t H\lambda+f^t\lambda \quad \text{constraint to } A\lambda\leq a \text{ and } B\lambda=b$$

که در آن A، B و H ماتریس‌های n×n بوده و f، a و b بردارهای nتایی هستند.

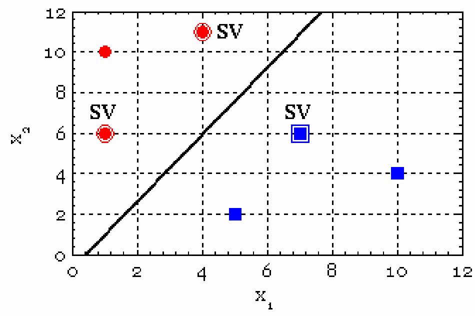
در مثال ما با در نظر گرفتن بردار f بصورت برداری از ۱-ها، بردارهای a و b بردارهایی از صفرها، ماتریس A بصورت ماتریس قطری با درایه‌های قطری ۱- و ماتریس B با سطر اول برابر z و صفر برای سایر سطرها، مساله به فرم استاندارد تبدیل شده و با دستور زیر در مطلب قابل حل می‌باشد:

$$\lambda=quadprog(H,f,A,a,B,b)$$

با حل مساله بهینه‌سازی فوق، λ به صورت $\lambda=[0.036 \quad 0 \quad 0.039 \quad 0 \quad 0.076 \quad 0]^t$ بدست می‌آید. در نتیجه (w,w_0) نیز برابر خواهند بود با:

$$w=\sum_{i=1}^{n}\lambda_i z_i x_i=(\lambda.z)^t x=[-0.33 \quad 0.20]^t$$

$$w_0=\frac{1}{z_1}-w^t x_1=0.13$$



❐

۴- روش‌های بر پایه کمینه‌سازی خطا

روش‌های بر پایه کمینه‌سازی خطا، در حالتی که داده‌ها به صورت خطی جداپذیر نباشند، خطای موجود را کمینه می‌کنند.

یکی از بزرگترین مزایای استفاده از جداسازهای خطی سادگی استفاده از آنهاست. تا جایی که در برخی از موارد با وجود اینکه می‌دانیم که بردارهای ویژگی داده شده بصورت خطی از هم جداپذیر نیستند ولی باز هم علاقمند به استفاده از توابع جداساز خطی هستیم. در این موارد تلاش بر این است که جداسازی یافت شود که خطای موجود را کمینه نماید.

در این بخش به بررسی روش‌هایی می‌پردازیم که بر اساس کمینه‌سازی خطا اقدام به یافتن بهترین جداساز می‌نمایند. در این قسمت ابتدا به عنوان یک مثال روش‌های بر پایه کمینه‌سازی خطا را برای حالتی بررسی می‌کنیم که داده‌های خطی شده دارای توزیع نرمال باشند. سپس در بخش‌های بعدی به بررسی روش‌های کلی‌تر بر پایه مربعات خطا خواهیم پرداخت.

۴.۱- روش‌های برپایه کمینه‌سازی خطا و توزیع‌های نرمال

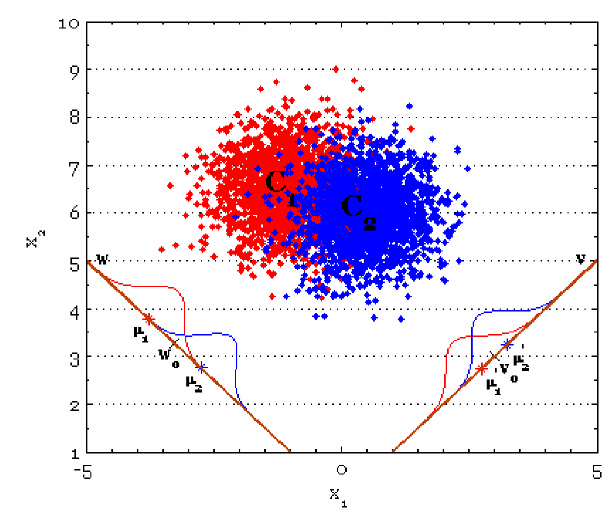
[Fukunaga] همانطور که پیش‌تر ذکر شد، در رابطه $g(x)=w^t x+w_0$ جزء $w^t x$ بیان می‌کند که داده‌ها باید بر روی بردار w تابانده شوند. و بر روی این ابرصفحه (خط در حالت دو بعدی) که داده‌ها قرار دارند، w_0 مرز جداسازی را تعیین می‌کند (شکل ۱۶).

اگر بردارهای ویژگی بصورت نرمال توزیع شده باشند یا تعداد آنها زیاد باشد، نگاشت آنها نیز نرمال خواهد بود.

اگر بردارهای ویژگی داده شده برای دسته‌بندی، بصورت نرمال توزیع شده باشند، نگاشت آنها نیز نرمال خواهد بود (شکل ۱۶). اگر بردارهای ویژگی بصورت نرمال توزیع نشده باشند ولی تعداد آنها زیاد باشد، نیز، حاصل نگاشت خطی بردارهای ویژگی، نزدیک به نرمال خواهد بود. زیرا در نگاشت خطی، هر بردار ویژگی نگاشت شده، مجموع وزن‌دار ابعاد بردار ویژگی معادلش در فضای چند بعدی بوده و چون حاصلجمع چند ترم می‌باشد، قضیه حد مرکزی[12] در مورد آن صادق است.

تمرین ۱۰: توابع جداساز خطی برای حالتی که داده‌ها با مقایسه میانگین‌های داده‌های دو کلاس جداپذیر هستند بهتر عمل می‌کند یا برای حالتی که داده‌ها با مقایسه پراکندگی (واریانس) داده‌های دو کلاس جداپذیر هستند یا هر دو؟

در شکل ۱۶، مساحت فضاهای برهم‌افتادگی نمودارهای گاوسی نمایانگر میزان خطاها هستند. خطای جداسازی، پس از تابش داده‌ها در جهت w، کمتر از خطای جداسازی، پس از تابش داده‌ها در جهت v، می‌باشد.



شکل ۱۶: نگاشت خطی داده‌ها، جداسازی و خطا

جداساز خوب، جداسازی است که جهت (w,w_0) را به قسمی پیدا کند که کمترین میزان خطا را برای جداسازی، پس از تابش داده‌ها در آن جهت (در فضای g)، داشته باشیم.

اگر داده‌ها از یک جمعیت نرمال انتخاب شده باشند، حاصل تابش آنها نیز دارای توزیعی نرمال خواهد بود. میزان خطا نیز بوسیله

[12] Central Limit Theorem

پارامترهای این توزیع نرمال $\left(\eta_i = E[g(x)|C_i], \sigma_i^2 = Var[g(x)|C_i]\right)$ تعیین خواهد شد که $\left(\eta_i, \sigma_i^2\right)$ توابعی از (w, w_0) هستند. در این حالت داریم:

$$\begin{aligned} \eta_i &= E[g(x)|C_i] = W^t E[x|C_i] + w_0 = w^t \mu_i + w_0 \\ \sigma_i^2 &= Var[g(x)|C_i] = w^t E\left[(x-\mu_i)(x-\mu_i)^t \,|\, C_i\right] w = w^t \Sigma_i w \end{aligned} \qquad (۲۲)$$

فرض کنید $J(\eta_1, \eta_2, \sigma_1^2, \sigma_2^2)$ معیاری باشد که برای تعیین بهترین (w, w_0) باید بهینه شود. در این صورت می‌بایستی

$$\begin{cases} \dfrac{\partial J}{\partial w} = \dfrac{\partial J}{\partial \sigma_1^2}\dfrac{\partial \sigma_1^2}{\partial w} + \dfrac{\partial J}{\partial \sigma_2^2}\dfrac{\partial \sigma_2^2}{\partial w} + \dfrac{\partial J}{\partial \eta_1}\dfrac{\partial \eta_1}{\partial w} + \dfrac{\partial J}{\partial \eta_2}\dfrac{\partial \eta_2}{\partial w} = 0 \\ \& \\ \dfrac{\partial J}{\partial w_0} = \dfrac{\partial J}{\partial \sigma_1^2}\dfrac{\partial \sigma_1^2}{\partial w_0} + \dfrac{\partial J}{\partial \sigma_2^2}\dfrac{\partial \sigma_2^2}{\partial w_0} + \dfrac{\partial J}{\partial \eta_1}\dfrac{\partial \eta_1}{\partial w_0} + \dfrac{\partial J}{\partial \eta_2}\dfrac{\partial \eta_2}{\partial w_0} = 0 \end{cases} \qquad (۲۳)$$

با ترکیب روابط (۲۲) و (۲۳) خواهیم داشت:

$$2\left[\frac{\partial J}{\partial \sigma_1^2}\Sigma_1 + \frac{\partial J}{\partial \sigma_2^2}\Sigma_2\right] w + \left[\frac{\partial J}{\partial \eta_1}\mu_1 + \frac{\partial J}{\partial \eta_2}\mu_2\right] = 0 \qquad (۲۴)$$

$$\frac{\partial J}{\partial \eta_1} + \frac{\partial J}{\partial \eta_2}\frac{\partial \eta_2}{\partial w} = 0 \qquad (۲۵)$$

جداساز خطی برای حالتی که بردارهای ویژگی دارای توزیع نرمال بوده و معیار ما کمینه‌سازی خطا می‌باشد

با جایگذاری رابطه (۲۵) در رابطه (۲۴) و توجه به این نکته که خطا فقط به جهت w بستگی دارد و نه به سایز آن و همچنین با ساده‌سازی ترم‌های ثابت، خواهیم داشت:

$$\begin{aligned} & w = \left[s\Sigma_1 + (1-s)\Sigma_2\right]^{-1}(\mu_2 - \mu_1) \qquad ; s = \frac{\partial J / \partial \sigma_1^2}{\partial J / \partial \sigma_1^2 + \partial J / \partial \sigma_2^2} \\ & w_0 = the\, Solution\, of \left\langle \frac{\partial J}{\partial \eta_1} + \frac{\partial J}{\partial \eta_2} = 0 \right\rangle \end{aligned} \qquad (۲۶)$$

مثال ۸: با کمینه‌سازی خطا، (w, w_0) را برای حالتی بدست آورید که معیار کمینه‌سازی خطا برابر $J = \dfrac{(\eta_1 - \eta_2)^2}{\sigma_1^2 + \sigma_2^2}$ (معیار فیشر) باشد.

با توجه به رابطه (۲۶) ابتدا مقدار s را محاسبه می‌کنیم. داریم:

$$\frac{\partial J}{\partial \sigma_1^2} = \frac{\partial J}{\partial \sigma_2^2} = \frac{-(\eta_1 - \eta_2)^2}{(\sigma_1^2 + \sigma_2^2)^2} \Rightarrow s = \frac{1}{2}$$

در نتیجه $w = \left[\frac{1}{2}\Sigma_1 + \frac{1}{2}\Sigma_2\right]^{-1}(\mu_2 - \mu_1)$ و w_0 نیز باید با توجه به داده‌ها بدست آید (جواب بدست آمده را با جوابی که قبلا برای فیشر بدست آورده بودیم مقایسه نمائید).

❒

تمرین ۱۱: نشان دهید که با کمینه‌سازی خطا، جداساز بهینه برای حالتی که معیار کمینه‌سازی به صورت $J = \dfrac{p_1\eta_1^2 - p_2\eta_2^2}{p_1\sigma_1^2 + p_2\sigma_2^2}$ باشد، دارای پارامترهای $\left\langle w = [p_1\Sigma_1 + p_2\Sigma_2]^{-1}(\mu_2 - \mu_1), w_0 = -w^t(p_1\mu_1 + p_2\mu_2)\right\rangle$ خواهد بود.

تمرین ۱۲: با کمینه‌سازی خطا، (w, w_0) را برای حالتی بدست آورید که معیار کمینه‌سازی خطا برابر تابع خطای بیز در فضای تابش (فضای g) باشد. خطای بیز برابر است با:

$$.\varepsilon = p_1 \int_{-\eta_1/\sigma_1}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{\zeta^2}{2}} d\zeta + p_2 \int_{-\infty}^{-\eta_2/\sigma_2} \frac{1}{\sqrt{2\pi}} e^{-\frac{\zeta^2}{2}} d\zeta$$

۴.۲- روش میانگین مربعات خطا

تابع میانگین مربعات خطا در حالتی که داده‌ها به صورت خطی جداپذیر نیستند

[theodiridis] این بخش را با تمرکز بر حالت دو کلاسه شروع می‌کنیم. برای یک مساله دسته‌بندی دو کلاسه، تابع جداساز خطی همگن $g(x)$ را در نظر بگیرید، که برای کلاس C_1 مقدار $+1$ و برای کلاس C_2 مقدار -1 را بر می‌گرداند. از آنجایی که این تابع جداساز دارای خطا می‌باشد، جواب صحیح را به ازای هر x بر نمی‌گرداند. y را بقسمی در نظر بگیرید که تعلق واقعی داده x را به کلاس‌های C_1 و C_2 به ترتیب با مقادیر $+1$ و -1 نشان دهد (خروجی‌های مطلوب). در این صورت تابع میانگین مربع خطا (MSE[13]) برابر خواهد بود با:

$$J(w)=E\left[\left|y-g(x)\right|^2\right]=E\left[\left|y-x^t w\right|^2\right] \qquad (۲۷)$$

w باید طوری تعیین شود که $J(w)$ کمینه شود یا $\hat{w}=\arg\min_{w} J(w)$. برای کمینه کردن $J(w)$ نیز باید داشته باشیم:

$$\frac{\partial J(w)}{\partial w}=2E\left[x(y-x^t w)\right]=2E\left[xy-xx^t w\right]=2\left(E[xy]-E\left[xx^t\right]w\right)=0$$

در نتیجه:

$$\hat{w}=\frac{E[xy]}{E\left[xx^t\right]}=\frac{E[xy]}{R_x}=R_x^{-1}E[xy] \qquad (۲۸)$$

که در آن R_x ماتریس همبستگی[14] بردارهای ویژگی (معادل ماتریس کوواریانس) بوده و $E[xy]$ همبستگی متقابل[15] بردارهای ویژگی و خروجی مطلوب می‌باشد:

$$R_x=E[xx^t]=\begin{bmatrix} E[x_1x_1] & \ldots & E[x_1x_n] \\ E[x_2x_1] & \cdots & E[x_2x_n] \\ \vdots & \vdots & \vdots \\ E[x_nx\ ] & \cdots & E[x_nx_n] \end{bmatrix}, \qquad E[xy]=E\begin{bmatrix}\begin{bmatrix} x_1y \\ x_2y \\ \vdots \\ x_ny \end{bmatrix}\end{bmatrix}$$

روش میانگین مربعات خطا برای حالت چند کلاسه

بحث فوق را می‌توان به حالت k کلاسه نیز بسط داد. در این حالت باید k تابع جداساز خطی $g_i(x)$ بر اساس معیار میانگین مربعات خطا طراحی شوند. برای خروجی‌های مطوب نیز اگر $x\in C_i$، باید $y_i=1$ باشد، در غیر اینصورت $y_i=0$. در نتیجه برای هر بردار ویژگی x بردار $y^t=[y_1,...,y_k]$ را به عنوان خروجی مطلوب خواهیم داشت. در این مساله، هدف پیدا کردن k تابع جداساز یا در اصل یافتن $W=[w_1,...,w_k]$ می‌باشد. در این صورت خواهیم داشت:

$$\hat{W}=\arg\min_{W} E\left[\left\|y-W^t x\right\|^2\right]=\arg\min_{W} E\left[\sum_{i=1}^{k}(y_i-w_i^t x)^2\right]$$

که معادل k مساله مستقل کمینه‌سازی میانگین مربعات خطا می‌باشد. و این یعنی اینکه برای طراحی مجموعه توابع جداساز برای حالت k کلاسه بر اساس میانگین مربعات خطا، کافی است که مساله بصورت k زیر مساله کوچکتر دو کلاسه در نظر گرفته شود که برای هر زیر مساله iام، می‌بایستی داده‌های کلاس C_i به عنوان یک کلاس و سایر داده‌ها، به عنوان کلاس دیگر در نظر گرفته شوند (ماشین جداپذیر خطی کامل).

۴.۳- روش کمترین میانگین مربعات (الگوریتم Widrow-Hoff)

دیدیم که برای بدست آوردن تحلیلی w از طریق رابطه (۲۸) نیاز به ماتریس همبستگی بردارهای ویژگی و بردار همبستگی متقابل ورودی‌ها و خروجی‌های مطلوب است. بدست آوردن این اطلاعات همیشه راحت نیست، بخصوص اگر تعداد بردارهای ویژگی ورودی بسیار زیاد باشد. در نتیجه مایلیم روشی داشته باشیم که بتوان بدون استفاده از این اطلاعات آماری به کمینه‌سازی رابطه (۲۷) اقدام نمود. برای این منظور می‌توانیم از روش کاهش گرادیان بهره بگیریم.

قانون بروز رسانی متغیرها (روش تک نمونه‌ای) برای این مساله بصورت زیر در می‌آید:

[13] Mean Square Error
[14] Correlation Matrix
[15] Cross-Correlation

$$w_{(k+1)} = w_{(k)} + \eta_{(k)} x_{(i)} \left(y_{(i)} - w_{(k)}^t x_{(i)} \right) \qquad (۲۹)$$

کمترین میانگین مربعات به میانگین مربعات خطا همگرا می شود.

که در آن (x_k, y_k)ها جفت‌های بردارهای ویژگی آموزشی و خروجی‌های مطلوب معادل آنها می‌باشند که بصورت متوالی به این رابطه معرفی می‌شوند. این الگوریتم به عنوان کمترین میانگین مربعات (LMS[16]) مشهور است که به میانگین مربعات خطا همگرا می‌شود. الگوریتم‌های LMS دیگری نیز با شرایط مختلف پیشنهاد شده‌اند.

توجه کنید که رابطه بازگشتی (۲۹) می‌تواند به عنوان الگوریتم تعیین سیستماتیک وزن‌های یک نرون خطی نیز بکار گرفته شود. این نوع از الگوریتم‌های آموزشی که خروجی مطلوب را به مجموع وزن‌دار ورودی‌های آموزشی نرون اعمال می‌کنند، اولین بار توسط Widrow و Hoff بکار گرفته شدند (الگوریتم Widrow-Hoff) و این ساختار که با نام adaline[17] شناخته می‌شود، در واقع نرونی است که در آموزش آن به جای پرسپترن از LMS استفاده می‌شود.

۴.۴- روش مجموع مربعات خطا (روش ماتریس شبه معکوس)

یک مساله دسته‌بندی به فرم نرمال را در نظر بگیرید. در این مساله برای تمامی بردارهای ویژگی باید $w^t x_i > 0$ باشد. یا اگر به ازای هر بردار ویژگی x_i عدد مثبت b_i (با هر مقدار دلخواهی) را داشته باشیم، باید برای تمامی بردارهای ویژگی $w^t x_i = b_i$ باشد. از آنجایی که $w^t x_i / \|w\|$ فاصله بردار ویژگی x_i از بردار w می‌باشد، می‌توان گفت که b_iها متناظر فواصل بردارهای ویژگی معادلشان از بردار w (با ضریب ثابت $1/\|w\|$) می‌باشند. پس حل معادله $Xw = b$ و بدست آوردن w، معادل بدست آوردن $g(x) = w^t x$ می‌باشد.

روش‌های تحلیلی قادر به حل مسایل جداسازی از طریق تخمین بر پایه مجموع مربعات خطا نیستند.

اگر بخواهیم مساله را به طریق تحلیلی حل کنیم به نتیجه $w = X^{-1} b$ می‌رسیم. برای محاسبه X^{-1}، ماتریس X باید یک ماتریس مربعی بوده و غیر تکین[18] باشد (دترمینان آن مخالف صفر باشد). از آنجایی که در هر سطر یکی از بردارهای ویژگی قرار می‌گیرد، تعداد سطرهای X برابر تعداد بردارهای ویژگی بوده و تعداد ستون‌ها نیز برابر تعداد ابعاد بردارهای ویژگی می‌باشد. مربعی بودن ماتریس X یعنی اینکه تعداد بردارهای ویژگی با تعداد ابعاد برابر باشد. اتفاقی که در عمل به ندرت رخ می‌دهد. اگر مربعی نبودن X بدلیل بیشتر بودن تعداد ابعاد بردارهای ویژگی از تعداد بردارهای ویژگی می‌بود، می‌توانستیم با اعمال یکی از الگوریتم‌های کاهش ابعاد، ماتریس را مربعی کنیم، ولی در این حالت، نمی‌توانیم اقدامی برای رفع مشکل انجام دهیم. در نتیجه علیرغم اینکه روش تحلیلی تضمین می‌کند که ابر صفحه جداساز را بیابد، در عمل قابل استفاده نیست.

بنابراین باید به دنبال یک حل تقریبی برای این معادله باشیم. هر چند که امکان دارد که این حل تقریبی لزوما ابر صفحه کاملا جداسازی را بدست ندهد، ولی می‌توان امیدوار بود که جوابی که بدست می‌آید، جواب تقریبا مناسبی باشد. برای این منظور می‌توان به جای پیدا کردن جواب صفر $Xw - b$ سعی نمود که آن را تا حد امکان کمینه نمود. در این صورت مساله به یک مساله بهینه‌سازی با معیار کمینه‌سازی زیر تبدیل می‌شود:

$$J(w) = \|Xw - b\|^2 = \sum_{i=1}^{n} (w^t x_i - b_i)^2 = \sum_{i=1}^{n} e_i^2$$

از آنجایی که $Xw - b$ نمایانگر خطا می‌باشد (چرا؟)، معیار فوق را معیار کمینه‌سازی مربعات خطا[19] می‌نامند. و همچنین بدلیل اینکه مجموع مربعات خطا را نیز کمینه می‌کند به روش تخمین مجموع مربعات خطا نیز مشهور است. در اصل در روش تخمین مجموع مربعات خطا، همانند روش میانگین مربعات خطا عمل می‌کنیم با این تفاوت که تابع هدف به صورت فوق در نظر گرفته می‌شود.

برای بدست آوردن w بهینه داریم:

$$\frac{\partial J(w)}{\partial w} = \sum_{i=1}^{n} 2 x_i (w^t x_i - b_i) = 2X^t (Xw - b) = 0$$

در نتیجه خواهیم داشت:

[16] Least Mean Squares
[17] Adaptive linear element
[18] Singular
[19] Minimum Squared Error Criterion

روش ماتریس شبه معکوس برای بدست آوردن تخمین مجموع مربعات خطا

$$X^tXw - X^tb = 0 \Rightarrow X^tXw = X^tb \Rightarrow w = \frac{X^tb}{X^tX} = (X^tX)^{-1}X^tb$$

X^tX ماتریس همبستگی بردارهای ویژگی (معادل ماتریس کوواریانس) می‌باشد که یک ماتریس مربعی می‌باشد که دارای d+1 سطر و ستون بوده و اغلب غیر تکین می‌باشد. $(X^tX)^{-1}X^t$ را نیز ماتریس شبه وارون[20] X می‌گویند (چون داریم، $\left((X^tX)^{-1}X^t\right)X = (X^tX)^{-1}(X^tX) = I$).

در تعیین b_iها در یک مساله، توجه داشته باشید که اگر برای همه iها $b_i = 1$ باشد، به ازای همه بردارهای ویژگی $g(x) = w^tx_i = 1$ خواهد شد. w بدست آمده نیز معادل ابر صفحه‌ای خواهد بود که قادر به جداسازی تمامی بردارهای ویژگی می‌باشد. پس در یک مساله برای بدست آوردن تابع جداساز، اگر اطلاعاتی از مساله نداشته باشیم، برای تعیین مقادیر b_iها، می‌توان در ساده‌ترین حالات تمامی آنها را برابر ۱ در نظر گرفت.

تخمین مجموع مربعات خطا، در nهای بزرگ، به سمت MSE میل می‌کند

توضیحاتی که برای حالت چند کلاسه در MSE داده شد، در اینجا نیز صادق است. همچنین لازم به ذکر است که تخمین مجموع مربعات خطا، در nهای بزرگ، به سمت MSE میل می‌کند (چرا؟).

مثال ۹: کلاس C_1 حاوی بردارهای ویژگی $\left\{[0.2,0.7]^t,[0.3,0.3]^t,[0.4,0.5]^t,[0.6,0.5]^t,[0.1,0.4]^t\right\}$ و کلاس C_2 حاوی بردارهای ویژگی $\left\{[0.4,0.6]^t,[0.6,0.2]^t,[0.7,0.4]^t,[0.8,0.6]^t,[0.7,0.5]^t\right\}$ را در نظر بگیرید. مرزهای جداساز بهینه را با روش مجموع مربعات خطا طراحی نمائید.

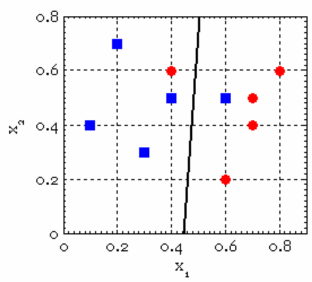
مرز جداساز بهینه به فرم $w_2x_2 + w_1x_1 + w_0 = 0$ خواهد بود. در نتیجه هدف پیدا کردن $w = [w_2, w_1, w_0]^t$ بهینه خواهد بود. برای ساختن ماتریس X ابتدا ۱۰ بردار ویژگی داده شده را با در نظر گرفتن ۱ به عنوان بعد سوم‌شان به یک ماتریس ۱۰×۳ تبدیل می‌کنیم. b_iها را نیز برای بردارهای ویژگی متعلق به کلاس اول برابر ۱ و برای بردارهای ویژگی متعلق به کلاس دوم برابر ۱− در نظر می‌گیریم. در نتیجه بردار b از ۵ عدد ۱ و ۵ عدد ۱− ساخته می‌شود. سپس ماتریس همبستگی بردارهای ویژگی و را به صورت زیر محاسبه می‌کنیم:

$$X^tX = \begin{bmatrix} 2.8 & 2.24 & 4.8 \\ 2.24 & 2.41 & 4.7 \\ 4.8 & 4.7 & 10 \end{bmatrix} \quad \& \quad X^tb = \begin{bmatrix} -1.6 \\ 0.1 \\ 0.0 \end{bmatrix}$$

پس،

$$w = (X^tX)^{-1}X^tb = [-3.218, 0.241, 1.431]^t$$

در شکل زیر، داده‌های دو کلاس و مرز جداساز بهینه بدست آمده نمایش داده شده‌اند.



❒

روش ماتریس شبه معکوس در صورتی که ابعاد X^tX بالا باشد یا بردارهای ویژگی دارای همبستگی بالایی باشند، دچار مشکل می‌شود.

اگر تعداد ابعاد X^tX بالا باشد، محاسبه وارون آن هزینه‌بر است. همچنین اگر بردارهای ویژگی دارای همبستگی بالایی باشند (اگر سطرهای X ترکیبی خطی از سایر سطرها باشند)، دترمینان X^tX نزدیک به صفر خواهد شد و محاسبه وارون آن مشکل‌ساز خواهد شد. در این حالت برای حل مساله می‌توان از روش‌های بهینه‌سازی تکراری استفاده نمود. مثلا در استفاده از الگوریتم کاهش گرادیان، نظر به اینکه $\nabla J(w) = 2X^t(Xw - b)$ می‌باشد، قانون بروز رسانی تک نمونه‌ای به صورت زیر در

[20] Pseudoinverse

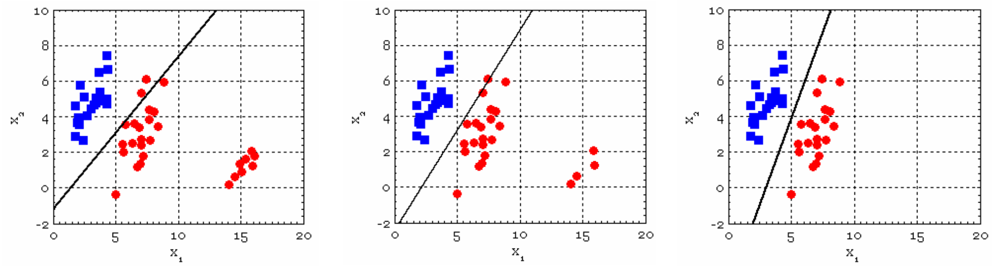
خواهد آمد:

$$w_{(k+1)} = w_{(k)} - \eta_{(k)} x_{(i)} \left(w_{(k)}^t x_{(i)} - b_{(i)} \right)$$

که معادل Widrow-Hoff در تخمین میانگین مربعات خطا است.

روش مجموع مربعات خطا پایدار نبوده و نسبت به تغییرات داده‌ها بسیار حساس است.

[Bishop] روش مجموع مربعات خطا پایدار نبوده و نسبت به تغییرات داده‌ها بسیار حساس است. در شکل ۱۷ اثرات اضافه نمودن داده‌ها، بر مرز بدست آمده بوسیله روش مربعات خطا نشان داده شده است دقت کنید که داده‌های دور از مرز، که بیشترین حاشیه امنیت را دارند، منجر به بیشترین تاثیرات بر روی درستی جداساز می‌شوند.



(پ) (ب) (الف)

شکل ۱۷: روش مجموع مربعات خطا بسیار حساس به داده می‌باشد. شکل‌های فوق، به ترتیب از الف تا پ، اثرات اضافه نمودن داده‌ها بر مرز جداپذیری را نشان می‌دهند.

متد Ho-Kashyap

در روش تخمین مجموع مربعات خطا، راهکاری برای تعیین مقادیر b_iهای بهینه ارائه ندادیم. سوالی که ممکن است پیش بیاید این است که چه مقادیری برای خروجی‌های مطلوب دو کلاس در نظر گرفته شوند تا جواب بهتری بدست آید؟

مجموع مربعات خطا ممکن است همیشه جواب بهینه‌ای بدست ندهد، هر چند که اکثر اوقات جواب قابل قبولی را بدست می‌دهد.

در روش تخمین مجموع مربعات خطا قادر به حل $Xw = b$ نبودیم و با کمینه‌سازی $Xw - b$ فقط تقریبی از آن رابه دست آوردیم. یعنی $Xw \approx b$ یا $w^t x_i = b_i + \varepsilon_i$. مقادیر ε_i ها ممکن است منفی نیز باشد. در این حالت اگر ε_iها اعداد منفی کوچکی باشند، باز هم $w^t x_i$ بزرگتر از صفر بوده و دسته‌بندی به درستی انجام می‌شود. ولی چنانچه تقریب بدست آمده مناسب نبوده و ε_iها اعداد منفی بزرگی باشند، $w^t x_i$ منفی شده و دسته‌بندی اشتباه انجام می‌شود. نتیجه اینکه تقریب مجموع مربعات خطا ممکن است همیشه جواب بهینه‌ای بدست ندهد، هر چند که اکثر اوقات جواب قابل قبولی را بدست می‌دهد.

مثال ۱۰: آیا اگر b_iها اعداد بزرگی در نظر گرفته شوند، می‌توان بر مشکل عدم بهینگی روش تخمین مجموع مربعات خطا غلبه کرد؟

خیر. فرض کنید به جای b_iها اعداد βb_i در نظر گرفته شوند که در آن β عدد مثبت دلخواهی باشد. در این صورت w برابر خواهد بود با:

$$w = \arg\min_w \|Xw - \beta b\|^2 = \arg\min_w \beta^2 \|X(w/\beta) - b\|^2 = \arg\min_w \|X(w/\beta) - b\|^2 = \beta w_{old}$$

در نتیجه اگر برای یکی از داده‌ها و تابع جداساز قبلی داشته بوده باشیم $w_{old}^t x_i < 0$. در تابع جداساز فعلی نیز خواهیم داشت:

$$. w^t x_i = \beta w_{old}^t x_i < 0$$

و این یعنی اینکه اندازه b_iها به تنهایی مهم نیست. بلکه اختلاف آنها نسبت به هم تاثیرگذار است.

❒

فرض کنید که در یک مساله که با تخمین مجموع مربعات می‌خواهیم ابر صفحه جداساز را بدست آوریم، بردارهای ویژگی بصورت خطی قابل جداسازی باشند. در این صورت w^* و b^* وجود دارند به طوری که $Xw^* = b^* > 0$. می‌دانیم که با داشتن b^* قادر به یافتن w^* خواهیم بود. اما خود b^* نیز مجهول می‌باشد. پس باید سعی کنیم همزمان هر دو مجهول را بیابیم. یعنی تابع زیر را باید کمینه کنیم:

$$J(w,b)=\|Xw-b\|^2\ ;\ b>0$$

بهینه‌سازی تابع هدف نسبت به دو متغیر به کمک روش‌های بهینه‌سازی تکراری

حل این معادله با روش‌های تحلیلی ممکن نیست. به همین دلیل از الگوریتم‌های دیگری برای حل این مساله بهره می‌گیریم. برای این منظور باید دو گام زیر تا رسیدن به همگرایی برداشته شوند:

۱. ثابت نگه داشتن b و کمینه کردن $J(w,b)$ نسبت به w.

۲. ثابت نگه داشتن w و کمینه کردن $J(w,b)$ نسبت به b.

گام اول با روش ماتریس شبه وارون قابل انجام است. برای یک b ثابت، مقدار w برابر خواهد بود با:

$$\nabla J_w(w,b)=2X^t(Xw-b)=0 \Rightarrow w=(X^tX)^{-1}X^tb$$

انجام گام دوم کمی مشکل‌تر است. زیرا $\nabla J_b(w,b)=-2(Xw-b)=0$ نتیجه $b=Xw$ را در بر خواهد داشت که چون محدودیت مثبت بودن اجزای b در آن لحاظ نشده است قابل استفاده نیست. سعی می‌کنیم الگوریتم کاهش گرادیان را با اعمال تغییراتی برای حل این مساله به کار ببریم. قانون بروز رسانی تک نمونه‌ای در حالت معمولی برای این مساله بصورت زیر خواهد بود:

$$b_{(k+1)}=b_{(k)}+\eta_{(k)}2(Xw_{(k)}-b_{(k)})$$

هر کدام از درایه‌های Xw_k-b_k در صورت منفی شدن می‌تواند منجر به منفی شدن درایه‌ای از b شود. برای رفع این مشکل تمامی درایه‌های منفی Xw_k-b_k را صفر می‌کنیم. یعنی،

$$b_{(k+1)}=b_{(k)}+\eta_{(k)}\left[(Xw_{(k)}-b_{(k)})+\left|(Xw_{(k)}-b_{(k)})\right|\right]$$

با انجام این عمل کماکان با کاهش گرادیان به سمت مقدار بهینه نزدیک می‌شویم ولی سرعت کاهش گرادیان و رسیدن به جواب بهینه کم شده است. حالت نهایی روش فوق که به روال Ho-Kashyap معروف است، به صورت زیر می‌باشد:

روال Ho-Kashyap

۱. مقادیر اولیه‌ای برای $w_{(0)}$ و $b_{(0)}>0$ در نظر بگیر و مقدار اولیه k را نیز برابر صفر قرار بده

۲. مقدار $b_{(k+1)}$ را از قانون $b_{(k+1)}=b_{(k)}+\eta_{(k)}\left[(Xw_{(k)}-b_{(k)})+\left|(Xw_{(k)}-b_{(k)})\right|\right]$ بدست بیاور.

۳. مقدار $w_{(k+1)}$ را از رابطه $w_{(k+1)}=(X^tX)^{-1}X^tb_{(k+1)}$ بدست بیاور.

۴. k را برابر k+1 قرار بده.

قدم‌های فوق را تا وقتی $Xw_{(k)}-b_{(k)}\geq 0$ یا $k>k_{\max}$ یا $b_{(k+1)}=b_{(k)}$ تکرار کن.

توجه کنید که برای همگرایی باید نرخ یادگیری دارای مقادیری با شرط $0<\eta<1$ باشد. چرا؟

تمرین ۱۳: شرایط پایانی روال Ho-Kashyap را بررسی کرده و برای هر مورد بیان کنید که چرا این شرط قرار داده شده است و در چه وضعیتی این شرایط برقرار خواهند شد.

تمرین ۱٤: روال Ho-Kashyap در حالتی که داده‌ها به صورت خطی جداپذیر نباشند، چه رفتاری از خود نشان خواهد داد؟

تمرین ۱۵: آیا شرایطی وجود دارد که در آن روال Ho-Kashyap همگرا نشود؟

تمرین ١٦: نشان دهید که روال Ho-Kashyap برای حالت k کلاسه، متناظر k مساله دو کلاسه خواهد بود.

۵- جمع‌بندی و مباحث تکمیلی

در این قسمت به مقایسه روش‌های مطرح شده می‌پردازیم. هم‌چنین برخی از موضوعات را که به مباحث فوق مرتبط بوده ولی فرصت اشاره به آنها در طی مباحث مطرح شده بوجود نیامد را در اینجا مطرح می‌کنیم.

۵.۱- مقایسه روش‌ها

در جدول زیر روش‌هایی را که بررسی کردیم، با معیارهایی که بر اساس آنها عمل می‌کنند و همچنین جواب نهایی‌ای که به دست می‌آوردند را آورده‌ایم:

روش	معیار	الگوریتم یافتن پاسخ
پرسپترون	$J(w)=\sum_{x\in X_m(w)} -w^t x$	$w_{(k+1)}=w_{(k)}+\eta_{(k)}x_i$
Relaxation	$J(w)=\frac{1}{2}\sum_{x\in X_b(w)}\frac{(w^t x-b)^2}{\lvert x\rvert^t}$	$w_{(k+1)}=w_{(k)}-\eta_{(k)}\frac{w_{(k)}^t x}{\lvert x\rvert^2}x$
فیشر	$J(w)=\frac{w^t S_B w}{w^t S_w w}$	$\vec{w}=S_W^{-1}(\mu_1-\mu_2)$
ماشین‌های بردار پشتیبان	$\max_{\lambda}\ L(\lambda)=\sum_{\forall i}\lambda_i-\frac{1}{2}\sum_{\forall i}\sum_{\forall j}\lambda_i\lambda_j z_i z_j x_i^t x_j$ $\text{subject to: (1) }\sum_{\forall i}\lambda_i z_i=0\ \&\ (2)\ \lambda\geq 0$	بدست آوردن λ_i های نا صفر: برنامه‌ریزی درجه دوم $w=\sum_{\forall i}\lambda_i z_i x_i\ \&\ w_0=1/z_i-w^t x_i$
کمینه‌سازی خطا برای داده‌های با توزیع نرمال	$J(\eta_1,\eta_2,\sigma_1^2,\sigma_2^2)$	$s=\frac{\partial J/\partial\sigma_1^2}{\partial J/\partial\sigma_1^2+\partial J/\partial\sigma_2^2}$ $w=\left[s\Sigma_1+(1-s)\Sigma_2\right]^{-1}(\mu_2-\mu_1)$ $w_0=\textit{the Solution of}\left\langle\frac{\partial J}{\partial\eta_1}+\frac{\partial J}{\partial\eta_2}=0\right\rangle$
میانگین مربعات خطا	$J(w)=E\left[\lvert y-x^t w\rvert^2\right]$	$w=R_x^{-1}E[xy]$
کمترین میانگین مربعات (Widrow-Hoff)	$J(w)=E\left[\lvert y-x^t w\rvert^2\right]$	$w_{(k+1)}=w_{(k)}+\eta_{(k)}x_{(i)}\left(y_{(i)}-w_{(k)}^t x_{(i)}\right)$
مجموع مربعات خطا (ماتریس شبه معکوس)	$J(w)=\sum_{i=1}^{n}(w^t x_i-b_i)^2=\sum_{i=1}^{n}e_i^2$	$w=(X^t X)^{-1}X^t b$
مجموع مربعات خطا (Widrow-Hoff)	$J(w)=\sum_{i=1}^{n}(w^t x_i-b_i)^2=\sum_{i=1}^{n}e_i^2$	$w_{(k+1)}=w_{(k)}-\eta_{(k)}x_{(i)}\left(w_{(k)}^t x_{(i)}-b_i\right)$
Ho-Kashyap	$J(w,b)=\lVert Xw-b\rVert^2\ ;\ b>0$	$b_{(k+1)}=b_{(k)}+\eta_{(k)}\left[(Xw_{(k)}-b_k)+\left\lvert(Xw_{(k)}-b_{(k)})\right\rvert\right]$ $w_{(k+1)}=(X^t X)^{-1}X^t b_{(k+1)}$

پرسپترون در در حالتی که بردارهای ویژگی به صورت خطی جداپذیر باشند، همگرا شده و ابرصفحه جداساز را می‌یابد. ولی وقتی داده‌ها بصورت خطی جدا پذیر نباشند، همگرا نمی‌شود. با ترکیب چند لایه‌ای از نرون‌های متصل به هم می‌توان مرزهای جداسازی پیچیده‌تری را نیز بدست آورد و به حل مسایل جداپذیر غیر خطی نیز پرداخت.

روش‌های بر پایه کمینه‌سازی خطا در حالتی که داده‌ها به صورت خطی جداپذیر نباشند نیز همگرا می‌شوند، ولی تضمین نمی‌کنند که جواب بهینه را بیابند. ولی متد Ho-Kashyap که فاصله داده‌ها تا مرز جداپذیری را نیز خود تعیین می‌کند، اگر چه از لحاظ زمان اجرایی هزینه‌برتر است، ولی همیشه به یک جواب بهینه همگرا می‌شود. این روش‌ها در مقابل نویزهای تکین

بسیار حساس می‌باشند.

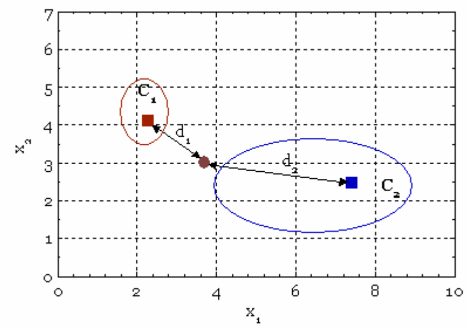
روش ماشین بردار پشتیبان نیز بهترین جواب ممکن را برای تعمیم به دست می‌دهد. این روش می‌تواند برای جداسازی داده‌های غیر خطی نیز به کار گرفته شود که در فصل‌های بعدی به آن خواهیم پرداخت.

۵.۲- مباحث متفرقه

یک کلاس و چند نماینده

روش‌های دسته‌بندی بر اساس کمترین فاصله در صورتی معتبر می‌باشند که کلاس‌ها دارای توزیعی نرمال با کوواریانس یکسان باشند.

[Bow&Bow] در روش دسته‌بندی بر اساس کمترین فاصله، یک بردار ویژگی از هر کلاس (یا یک نقطه در آن فضا) به عنوان نماینده آن کلاس در نظر گرفته شده و داده‌های آزمایشی بر اساس مقایسه فاصله با این بردار ویژگی (نقطه) دسته‌بندی می‌شوند. این کار در صورتی معتبر می‌باشد که کلاس‌ها دارای توزیعی نرمال با کوواریانس یکسان باشند. اگر این شرط برقرار نباشد، دسته‌بندی دچار اشتباه خواهد شد (شکل ۱۸).



شکل ۱۸: در صورت عدم برقراری شرایطی خاص، دسته‌بندی بر اساس کمترین فاصله، نتایج اشتباهی ارائه خواهد داد.

برای رفع این مشکل می‌توان از هر کلاس چندین بردار ویژگی (نقطه) را به عنوان نماینده کلاس در نظر گرفت. در این حالت فاصله یک بردار ویژگی از یک کلاس، برابر فاصله آن بردار ویژگی از نزدیک‌ترین نماینده کلاس به خودش خواهد بود. یعنی:

$$d(x,w_k) = \min_{m=1,\dots,N_k} \{d(x,z_k^m)\} \qquad (۳۰)$$

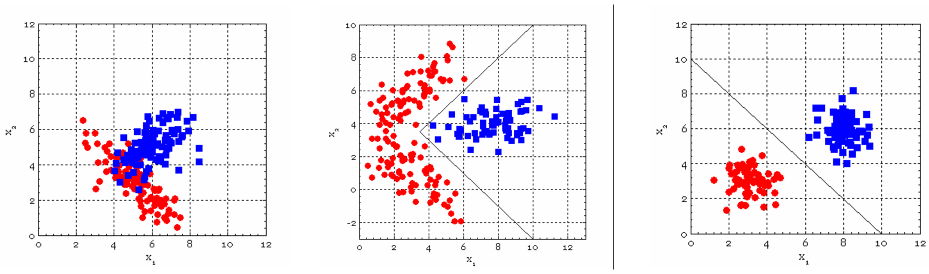
که در آن k نشان دهنده کلاس kام، m نشان دهنده نماینده mام و N_k تعداد نماینده‌های کلاس kام می‌باشد.

تمرین ۱۷: یک مساله دسته‌بندی با k کلاس و دسته‌بندی بر اساس کمترین فاصله را در نظر بگیرید. برای جلوگیری از خطاهای احتمالی از هر کلاس چندین نماینده انتخاب می‌شود. رابطه مرزهای جداساز را برای این مساله بیان نمائید.

جداسازی خطی تکه‌ای

یک تابع جداپذیر خطی تکه‌ای، تابعی است که در زیر نواحی‌های مختلف از فضای ویژگی، خطی می‌باشد، اما در کل فضا خطی نیست

[Bow&Bow] در برخی حالات بردارهای ویژگی داده شده به بصورت خطی جداپذیر نیستند ولی بصورت خطی تکه‌ای جداپذیرند. یک تابع جداپذیر خطی تکه‌ای، تابعی است که در زیر نواحی‌های مختلف از فضای ویژگی، خطی می‌باشد، اما در کل فضا خطی نیست. این تابع مرزهای خطی تکه‌ای را در زیرنواحی‌های مختلف بدست می‌دهد (شکل ۱۹).

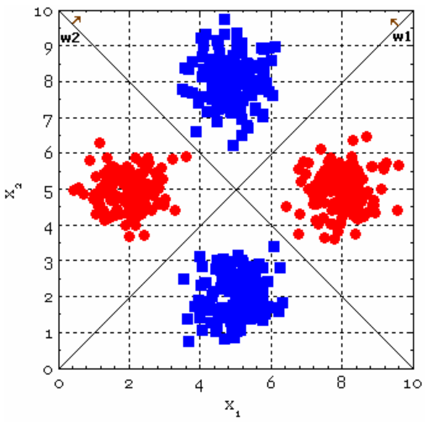


(پ) (ب) (الف)

شکل ۱۹: جداپذیری خطی (الف)، جداپذیری خطی تکه‌ای (ب) و جداناپذیری

جداسازی و ماشین‌های چند لایه‌ای

[Bow&Bow] دو خط متقاطع در یک صفحه، آن را به ۴ ناحیه مختلف تقسیم می‌کنند. ممکن است این ۴ ناحیه دو بدو طوری به دو کلاس مختلف تعلق داشته که بصورت خطی جداپذیر نباشند (شکل ۲۰).



شکل ۲۰: تقسیم فضا با دو خط به دو کلاس خطی جدا ناپذیر

در یک ماشین جداساز دولایه، لایه اول ارائه کننده خطوط جداساز بوده و لایه دوم در مورد نحوه تقسیم‌بندی نواحی بدست آمده تصمیم‌گیری می‌کند.

این داده‌ها را می‌توان با یک جداساز دو لایه (ماشین دو لایه) از هم تفکیک نمود. لایه اول در این ماشین، ارائه کننده خطوط جداساز بوده و لایه دوم در مورد نحوه تقسیم‌بندی نواحی بدست آمده تصمیم‌گیری می‌کند.

با توجه به توضیحات ارائه شده، لایه اول ماشین دولایه حاوی تعدادی ماشین جداساز خطی معمولی می‌باشد. فرض کنید $g_1(x)$ و $g_2(x)$ توابع جداساز در این لایه باشند، که w_1 و w_2 مرزهای جداساز متناظر آنها می‌باشند که در شکل فوق نشان داده شده‌اند. همچنین فرض کنید برای هر مرز جداساز، ناحیه‌ای که با پیکان علامت خورده است، ناحیه‌ای باشد که تابع جداساز متناظر آن، در آن ناحیه مقدار مثبتی را برمی‌گرداند. در این صورت برای چهار ناحیه تشکیل شده خواهیم داشت:

	A	B	C	D
$g_1(x)$	+	-	-	+
$g_2(x)$	+	+	-	-
$x \in$	C_1	C_2	C_1	C_2

در این مثال، برای لایه دوم نیز که وظیفه تصمیم‌گیری را بر عهده دارد، به راحتی می‌توان یک ضرب کننده را قرار داد. این ضرب کننده خروجی‌های دو تابع جداساز را در هم ضرب نموده و بر اساس نتیجه بدست آمده تصمیم‌گیری می‌کند. مقدار برگشتی مثبت توسط ضرب کننده نمایانگر تعلق داده به کلاس C_1 و مقدار منفی نشان دهنده تعلق داده به کلاس C_2 می‌باشد.