فصل ۵

توابع متمایزساز خطی

۵-۱- مقدمه :

ما فرض می کنیم که شکل اصلی توزیع احتمالی شناخته شده ای را داریم و از نمونه های آموزشی برای تخمین مقادیر پارامترها استفاده می کنیم. در این فصل ، فرض می کنیم که شکل درستی برای توابع متمایزساز داریم و از نمونه های تخمین زده شده مقادیر پارامترهای دسته بندی شده استفاده می کنیم.ما فرایندهای متنوعی برای تخمین توابع متمایز ساز داریم که بعضی از آنها آماری هستند و بعضی نیستند. هیچ کدام از آنها به هر حال دانشی برای فرم های توزیع احتمالی اساسی نیاز ندارند و در یک معنی محدودشده ای ، غیر پارامتریک هستند. در این فصل با مفاهیم توابع متمایز ساز که یا با مولفه ای از X که خطی هستند و یا با بعضی از توابع از X که خطی می باشند ، آشنا می شویم. توابع متمایز ساز خطی ، ویژگی های تحلیلی متنوعی دارند.آنها می توانند بهینه باشند اگر توزیع اصلی آنها به صورت همکاری مشترک باشد مانند توزیع گوسین با همگرایی یکسان که با انتخاب هوشمندی از تشخیص گرهای ویژگی بدست می آید. حتی وقتی که بهینه نباشد ، ممکن است مقداری از کارایی را برای بدست آوردن سود ساد سازی از دست بدهیم . توابع متمایز ساز خطی برای محاسبه آسان هستند و با توجه به اطلاعات پیشنهاد شده دیگر ، دسته بندی خطی یک مورد مناسب جذاب در ابتدا برای آزمون دسته بندی است .آنها همچنین یک تعدادی از مفاهیم اصلی مهم را بدست آمده با شبکه های عصبی را استفاده خواهند کرد. مسئله پیدا کردن تابع متمایز ساز خطی برای مینیمم کردن تابع استاندارد فرموله شده است. تابع معیار استاندارد قبلی برای دسته بندی با یک احتمال خطا و یا خطای آموزشی پیشنهاد شده است که با اتلاف سود دسته بندی برای مجموعه نمونه های آموزشی است. این حقیقت مورد اهمیت است که علی رغم مناسب بودن این معیار استاندارد ، مشکلاتی برای آن وجود دارد. از آنجا که هدف دسته بندی الگوهای تست جدید می باشد ، یک خطای آموزشی کوچک ، خطای تست کوچک را تضمین نمی کند .که یک مسئله جذاب و موشکافانه ای است که در این فصل به آن پرداخته می شود. مشکل مشتق شدن متمایز ساز خطی خطای مینیمم را داریم و به این دلیل ، چندین تابع معیار استاندارد که از نظر تحلیلی قابل قبول هستند را بررسی می کنیم. در این فصل بیشتر به مطالعات درباره ویژگی های همگرایی و پیچیدگی های محاسباتی فرایند گرادیان نزولی متنوع برای مینیمم کردن توابع معیار استاندارد می پردازیم. شباهت بین بسیاری از فرایند ها ، بعضی از زمان ها ، مشکلاتی را برای نگه داشتن اختلاف بین آنها می سازد و به این دلیل یک خلاصه ای از نتایج اصلی را در جدول ۵-۱ در انتهای بخش ۵-۱۰ می بینیم.

۵-۲- توابع متمایز ساز خطی و سطوح تصمیم گیری

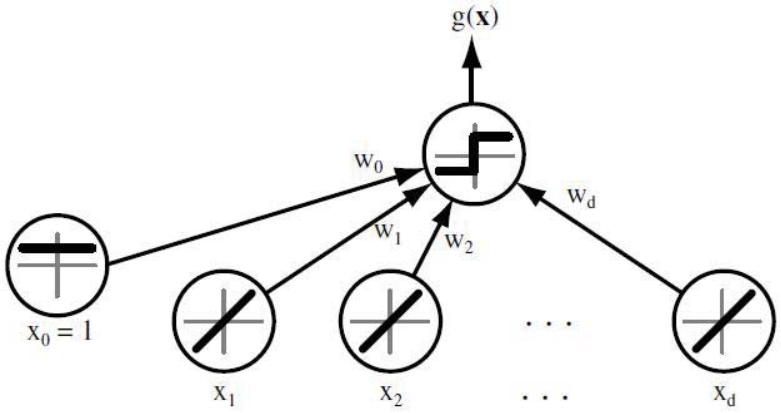
۵-۲-۱- حالت دو کلاسی

یک تابع متمایز ساز ، یک ترکیب خطی از مولفه های x است که به صورت زیر نوشته می شود :

$$g(\mathbf{x}) = \mathbf{w}^t\mathbf{x} + w_0 \qquad (۱)$$

که در آن w بردار وزنی و w_0 وزن آستانه ای و یا بایاس است. یک دسته بند دو کلاسی خطی با قوانین تصمیم گیری زیر پیاده می شود :

اگر $g(\mathbf{x}) > ۰$ باشد کلاس ω_1 اتفاق افتاده است و اگر $g(\mathbf{x}) < ۰$ باشد کلاس ω_2 اتفاق افتاده است. بنابراین ، x با کلاس ω_1 مشخص می شود اگر ضرب داخلی $\mathbf{w}^t\mathbf{x}$ از آستانه $-w_0$ بدست می آید و در غیر اینصورت از ω_2 بدست می آید. اگر $g(\mathbf{x}) = ۰$ باشد، x با کلاس دیگری تعیین می شود . ام در این فصل ما باید یک تخصیص تعریف نشده ای را بکاربریم. شکل ۵-۱- یک پیاده سازی معمولی را نشان می دهد. یک مثال واضح از ساختار معمولی ازسیستم بازشناسی الگو ها را داریم.



شکل ۵-۱: یک نمونه دسته بندی خطی با واحد ورودی d برای هر مقدار مولفه بردار ورودی را داریم. هر مقدار ویژگی ورودی x_i ، با وزن w_i ضرب می کنیم. مجموع خروجی واحد همه ضرب ها با ۱+ می باشد اگر $\mathbf{w}^t\mathbf{x} + w_0 > ۰$ باشد در غیر این صورت ۱- می شود.

معادله $g(\mathbf{x}) = ۰$ با سطح تصمیم گیری نقاط جداگانه تخصیصی برای ω_1 از نقاط تخصیصی برای ω_2 تعریف می شود. وقتی g(x) خطی است سطح تصمیم گیری ابر صفحه می شود. اگر $\mathbf{x}_1$ و $\mathbf{x}_2$ هر دو در سطح تصمیم گیری هستند سپس :

$$\mathbf{w}^t\mathbf{x}_1 + w_0 = \mathbf{w}^t\mathbf{x}_2 + w_0$$

یا

$$\mathbf{w}^t(\mathbf{x}_1 - \mathbf{x}_2) = ۰$$

این نشان می دهد که w با هر برداری در ابرصفحه نرمال می شود. به طور معمول ، ابرصفحه H به فضای ویژگی در دو نیم صفحه تقسیم می شود. ناحیه تصمیم گیری R_1 برای ω_1 و ناحیه R_2 برای ω_2 است. از آنجا که $g(\mathbf{x}) > ۰$ را داریم اگر x در R_1 باشد بردار نرمال نقاط w را در داخل R_1 داریم. بعضی از زمان ها برای هر x در R_1 یک سمت مثبت از H را داریم و

برای هر x در R_2 یک سمت منفی را داریم. تابع متمایزساز $\mathbf{g(x)}$ با یک اندازه گیری ریاضی فاصله از x برای ابرصفحه داده شده است. شاید آسانترین راه این باشد که x را به صورت زیر داشته باشیم.

$$\mathrm{X} = X_p + \mathrm{r}\frac{w}{||w||}$$

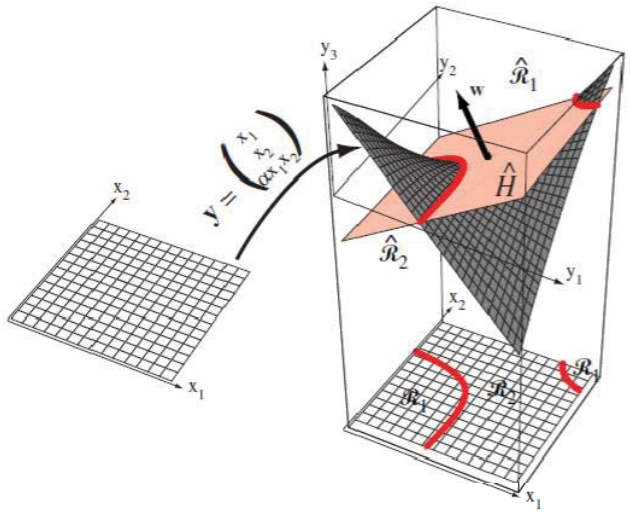
از آنجا که $\mathbf{x}_p$ تخمین نرمال x در H است و r با فاصله ریاضی توصیف می شود، r مثبت است اگر x یک سمت مثبت باشد و منفی است اگر x یک سمت منفی باشد. از آنجا که $g(\mathbf{x}_p) = ۰$ را داریم :

$$\mathrm{g(x)} = \mathrm{W^tX} + \mathrm{W}_{۰} = \mathrm{r}\|\mathrm{W}\|$$

یا

$$\mathrm{r} = \frac{g(x)}{\|w\|}$$

بویژه فاصله محل تقاطع محورها با H با $W_{۰} / \|\mathrm{W}\|$ داده می شود. اگر $w_0 > ۰$ را داشته باشیم ، محل تقاطع روی سمت مثبت H می باشد. اگر $w_0 < ۰$ را داشته باشیم ، محل تقاطع روی سمت منفی H می باشد . اگر $w_0 = ۰$ را داشته باشیم ، سپس $g(\mathbf{x})$ شکل همگن $\mathbf{w}^t\mathbf{x}$ است و ابرصفحه مجهول در محل تقاطع محور ها است. از نظر هندسی ، نتایج هندسی در شکل ۵-۲ نشان داده شده است.



شکل ۵-۲: مرز تصمیم گیری خطی H ، که در آن $g(\mathbf{x}) = \mathbf{w}^t\mathbf{x} + w_{۰} = ۰$ وجود دارد . فضای ویژگی به دو نیم صفحه R_1 (که در آن $g(\mathbf{x}) > ۰$ است) و R_2 (که در آن $g(\mathbf{x}) < ۰$ است) جدا می شود.

یک تابع متمایز ساز خطی در فضای ویژگی با سطح تصمیم گیری ابر صفحه تقسیم می شود. جهت سطح با بردار نرمال w تعیین می شود و موقعیت سطح با بایاس w_0 تعیین می شود. تابع متمایز ساز g(x) متناسب با فاصله علامت دار x تا ابرصفحه است که در آن $g(\mathbf{x}) > ۰$ وقتی که x در سمت مثبت است و در آن $g(\mathbf{x}) < ۰$ وقتی که x در سمت منفی است.

۵-۲-۲-حالت چند کلاسی

بیشتراز یک راه برای دسته بندی چندکلاسی بکار رفته در تابع متمایز ساز خطی تصور می کنیم. به طور مثال مسائل c-۱ کلاسی را ممکن است کاهش داد که در آن مسئله iام با تابع متمایز ساز خطی حل شده است که با نقاط جداشدنی به w_i تخصیص داده می شود و یا به w_i تخصیص داده نمی شود . یک روش گران تر استفاده از متمایز ساز خطی $c(c-۱)/۲$ بای هر یک از جفت کلاس ها می باشد. همان طور که در شکل ۵-۳ نشان داده شده است. هر دو این روش ها نواحی را به جایی که دسته بندی در آن تعریف نشده است هدایت می کند. تابع متمایز ساز c به صورت زیر تعریف می شود:

$$g_i(\mathbf{x}) = \mathbf{w}^t\mathbf{x}_i + w_{i0} \qquad i = ۱, ..., c, \qquad (۲)$$

X را به ω_i تخصیص می دهیم .اگر $g_i(\mathbf{x}) > g_j(\mathbf{x})$ برای همه i و j ها بجز $j \neq i$ برقرار باشد آ در این حالت سعی می کنیم کلاس بندی تعریف نشده در سمت چپ باشد. نتایج کلاس بندی یک ماشین خطی نامیده می شود. یک ماشین خطی فضای ویژگی را به c ناحیه تصمیم گیری تقسیم می کند که با $g_i(\mathbf{x})$ بزرگترین متمایزساز خطی بدست می آیداگر x در ناحیه R_i باشد. اگر R_i و R_j همسایه باشند مرز بین آنها یک بخش از ابرصفحه H_{ij} است که به صورت زیر تعریف می شود :

$$g_i(\mathbf{x}) = g_j(\mathbf{x})$$

یا

$$(\mathbf{w}_i - \mathbf{w}_j)^t\mathbf{x} + (w_{i0} - w_{j0}) = ۰$$

اگر $\mathbf{w}_i - \mathbf{w}_j$ با H_{ij} نرمال شود و فاصله علامت دارH_{ij} از x به صورت $(g_i - g_j) / \|\mathbf{wi} - \mathbf{w_j}\|$ داده شده باشد. بنابراین ، با ماشین خطی بردارهای وزنی خودشان نمی شوند اما اختلاف شان مهم است. از آنجا که $c(c-۱)/۲$ جفت ناحیه وجود دارد نیازی نیست که همه پیوسته باشند و تعدادکل ابرصفحه های تکه شده در سطح تصمیم گیری اغلب کمتر از $c(c-۱)/۲$ است که در شکل ۵-٤ نشان داده شده است. نواحی تصمیم گیری برای ماشین خطی محدب است و این محدودیت مطوئنا انعطاف پذیری و دقت دسته بندی را محدود می کند. بویژه برای کارایی خوب در نواحی تصمیم گیری تک تک باید متصل باشند و این ساخت ماشین خطی را به صورت مناسب حفظ می کند که در آن چگالی شرطی $p(\mathbf{x}|\omega_i)$ تک مدله است.

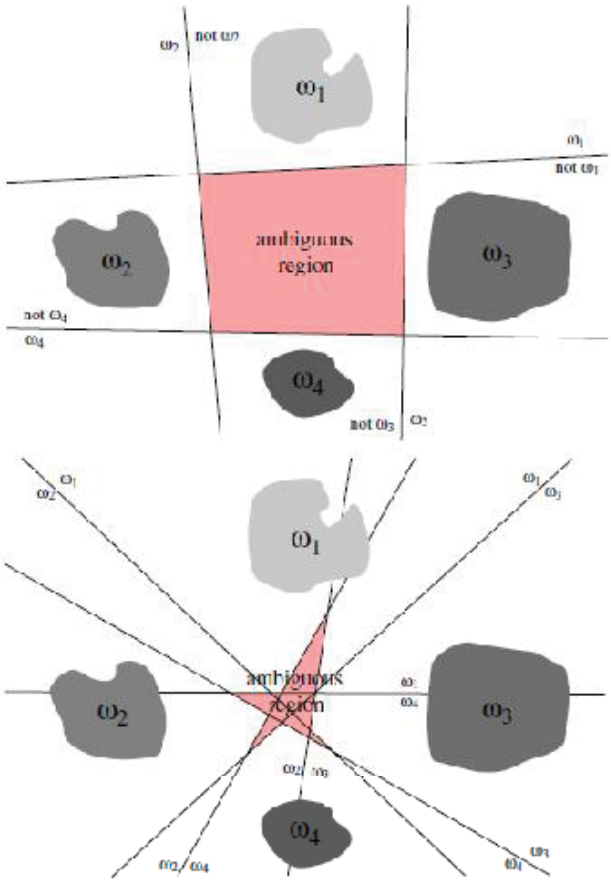
۵-۳- تعمیم توابع متمایز ساز خطی :

تابع متمایز ساز خطی g(x) به صورت زیر نوشته می شود :

$$g(x) = W_۰ + \sum_{i=۱}^{d} W_i X_i \qquad (۳)$$

ضرائب w_i مولفه های بردارهای وزنی w هستند .با اضافه کردن جملاتی شامل ضرب جفت مولفه های x، می توانیم تابع متمایز ساز درجه دو بدست بیاوریم.

$$g(x) = W_۰ + \sum_{i=۱}^{d} W_i X_i + \sum_{i=۱}^{d}\sum_{j=۱}^{d} w_{ij}\, x_i x_j \qquad (۳)$$

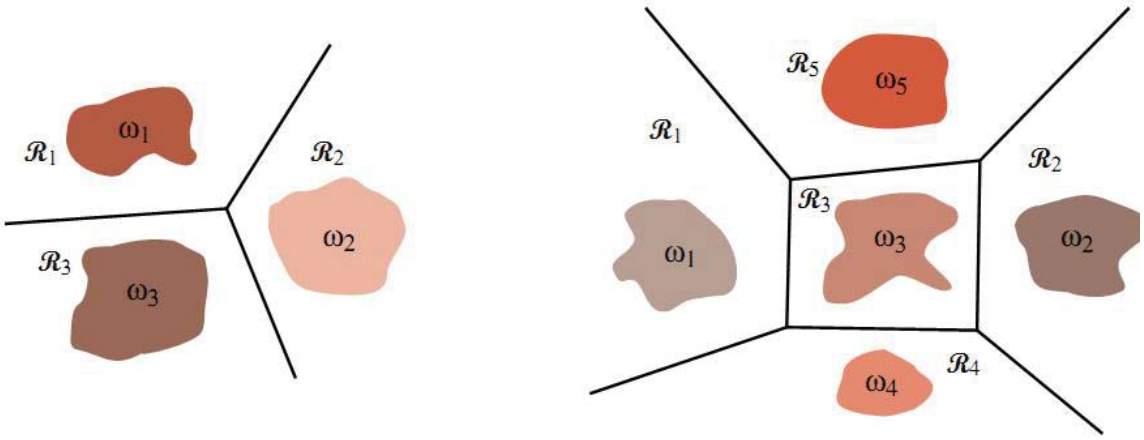


شکل ۵-۳ : مرز تصمیم گیری خطی برای مسائل ٤ کلاسی . شکل بالا دو بخشی سازی ω_i/not ω_i را نشان می دهد و قتی که شکل پایین دو بخشی سازی ω_i/ω_j را نشان می دهد. نواحی صورتی رنگ، کلاس های نامشخص تخصیص یافته اند.

از آنجا که $x_ix_j = x_jx_i$ را داریم می توان $w_{ij} = w_{ji}$را بدون توجه به اتلاف فرض کرد. بنابراین توابع متمایزساز درجه دو یک ضریب $d(d+1)/2$اضافی دارد که سطوح جداشدنی پیچیده را تولید می کند. این سطوح جدا شدنی با $g(x)= 0$ تعریف می شوند که یک سطح درجه دو یا hyperquadric است. جملات خطی در $g(x)$ با تبدیل محورها حذف می شوند. ما $\mathbf{W} = [w_{ij}]$, را تعریف می کنیم که یک ماتریس متقارن و غیر منحصر به فرد است و یک ویژگی مشخص از سطوح جدا شدنی است که با جملاتی از ماتریس مقیاس شده $\mathbf{W} = \mathbf{W}/(\mathbf{w}_t\mathbf{W}^{-1}\mathbf{w} - 4w_0)$. توصیف می شوند. اگر $\bar{\mathbf{W}}$ یک ضرب کننده مثبت از ماتریس همانی باشد ، سطح جدا شدنی یک hypersphere. می باشد. اگر $\bar{W}$مثبت تعریف شده باشد، سطح جدا شدنی یک hyperellipsoid می باشد.اگر بعضی از مقادیر ویژه $\bar{W}$مثبت باشد و مابقی منفی باشد سطح یک نوع متفاوتی از hyperhyperboloids می شود. این ها یک نوع سطوح جداشدنی هستند که حالت گوسین چند متغیره معمولی را بالا می برند. با توجه به اضافه کردن جملاتی

مانند $w_{ijk}x_i x_j x_k$ما کلاس توابع متمایز ساز چند جمله ای را بدست می آوریم. اینها به عنوان بسط سری کوتاه شده با مقادیر دلخواه $g(x)$ می باشند و تابع متمایز ساز خطی تعمیم یافته را می توان پیشنهاد کرد.

$$g(x) = \sum_{i=1}^{\hat{d}} a_i y_i(x) \quad (۵)$$



شکل ۵-٤ :» ضرب مرز تصمیم گیری با یک ماشین خطی برای یک مسئله ۳ کلاسی و ۵ کلاسی

یا

(٦)

A یک بردار وزنی $\bar{d}$بعدی است و $y_i(x)$توابع $\bar{d}$یبعضی اوقات توابع φنامیده می شوند که توابع دلخواهی از x می تواند باشد. این توابع با زیرسیستم های تشخیص ویژگی محاسبه می شود. با انتخاب صحیح این توابع و به اندازه کافی بزرگ بودن $\bar{d}$یکی را می توان برای هر تابع متمایزساز مورد نظر با یک بسط تخمین زد. تابع متمایزساز نتیجه شده برای x خطی نیست، اما برای y خطی است. $y_i(x)$توابع $\bar{d}$فقط نقاط در فضایx با d بعد را به نقاط فضایy با $a^t y$ بعد نگاشت می کند. متمایزساز متجانس $\bar{d}$نقاط در فضای تبدیل یافته را با یک ابرصفحه مجهول اولیه جدا می کند. بنابراین ، نگاشت از x به y مسئله را به پیدا کردن یک تابه متمایز ساز خطی متجانس کاهش می دهد. بعضی از مزایا و معایب این روش با توجه به نمونه های ساده توضیح داده می شوند. تابع متمایزساز درجه ۲ به صورت زیر است:

$$g(x)=a_1 + a_2 x + a_3 x^2 \quad (۷)$$

بطوریکه بردار ۳بعدی y به صورت زیر می شود:

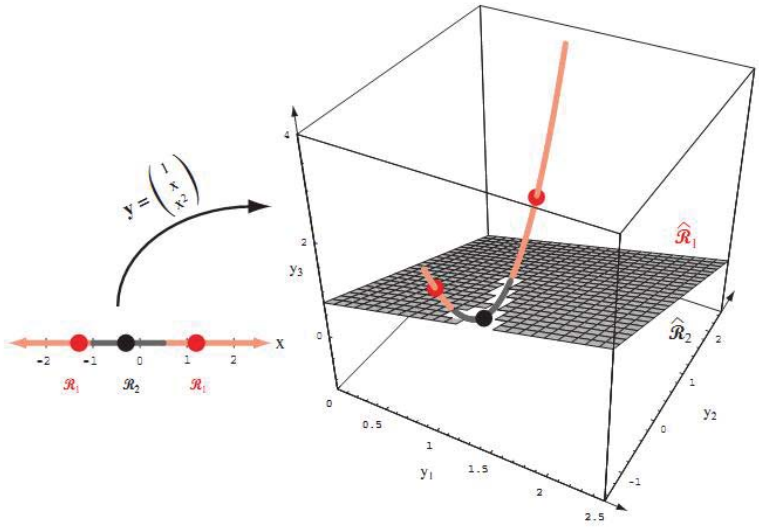
(۸)

$$y$$

نگاشت از x به y در شکل ۵-۵ نشان داده شده است. داده ها به طور دائمی ۱بعدی می مانند. از آنجا که تنوع x باعث می شود که y یک منحنی ۳ بعدی را دنبال کند. بنابراین، یک چیز با توجه به فوری بودن آن است که اگر x با احتمال پایین $p(x)$ کنترل شود ، چگالی تولید شده $\bar{p}(y)$مجددا تولید می شودو صفر می شود هرجا که به صورت منحنی باشد، که در آنجا به صورت

نامحدود است. این یک مسئله رایج است که هر وقت $\bar{d} \geq d$باشد و نگاشت نقاط از فضای d بعدی به فضایی با ابعاد بالاتر اتفاق می افتد. طرح تعریف شده $\bar{H}$ با $a^t y = ۰$ فضای y به دو ناحیه تصمیم گیری $\bar{R}_۱$و $\bar{R}_۲$تقسیم می کند.

جدا کردن صفحه مطابق با $a = (-۱,۱,۲)^t$و نواحی تصمیم گیری $\bar{R}_۱$و $\bar{R}_۲$ و نواحی تصمیم گیری تطبیق یافته $R_۱$و $R_۲$در فضای x اصلی در شکل نشان داده شده است. توابع متمایز ساز درجه ۲ $g(x)= -۱ + x + ۲x^۲$مثبت هستند اگر x<-۱ یا اگر x>.۵ برقرار باشدو بنابراین $R_۱$یک ضرب کننده پیوسته است.بنابراین اگر چه نواحی تصمیم گیری در فضای y محدب هستند ، این با هیچ میانگینی در فضای x قرار می گیرد. به هر حال حتی توابع ساده مرتبط $y_i(x)$، سطوح تصمیم گیری شامل شده در فضایx می تواند تا اندازه ای پیچیده باشد. (شکل ۵-٦)



شکل ۵-۵: نگاشت $y= (-۱,x,x^۲)^t$از یک خط و تبدیل آن بدست آمده است که یک سهمی با ۳ بعد می باشد. یک صفحه با نتایج صفحه y در داخل ناحیه منطبق به دو کلاس شکسته می شود و این با یک ناحیه تصمیم گیری پیوسته بدون وضوح و سادگی در فضای x یک بعدی است.

متاسفانه ، ابعاد اشتباه کار را برای استفاده از انعطاف پذیری مشکل می کنند . یک تابع متمایزساز درجه ۲ کامل از $\bar{d} = (d+۱)(d+۲)/۲$جمله تشکیل شده است. اگر d یک مینه بزرگ باشد، مثلا d=۵۰ ، به محاسبه بیشتر این جملات نیاز هست. این تابع با درجه ۳ و بیشتر از آن دارای زمان$O(\bar{d}^۳)$هستند. بنابراین ، مولفه های $\bar{d}$از بردارهای وزنی a باید نمونه های آموزشی را تعیین کنند. اگر $\bar{d}$به عنوان نشان دادن اندازه درجه سهولت برای تابع متمایز ساز باشد نیاز به تعداد نمونه هایی بیشتر از اندازه درجه سهولت می باشد. واضح است که یک بسط سری معمولی g(x) به آسانی یک نیاز غیر واقعی برای محاسبه داده را هدایت می کند. این نقطه ضعف می تواند با توجه به الزام مرزهای بزرگ سازگار شود و یا بیت الگو های آموزشی قرار گیرد، متناوبا شبکه عصبی چند لایه دراین مسئله، کپی های مضربی از توابع غیرخطی واحد از ویژگی های ورودی را بکار می برد. این باور ممکن است سخت باشد که منفعت بسیاری از تعمیم تابع متمایز ساز خطی داریم ، می توانیم حداقل سهولت و مزیت را با نوشتن g(x) به شکل متجانس $a^t y$ داشته باشیم. به خصوص در زمینه تابع متمایزساز خطی

(۹)

g

که ۱ = x_0 است بنابراین می توان نوشت :

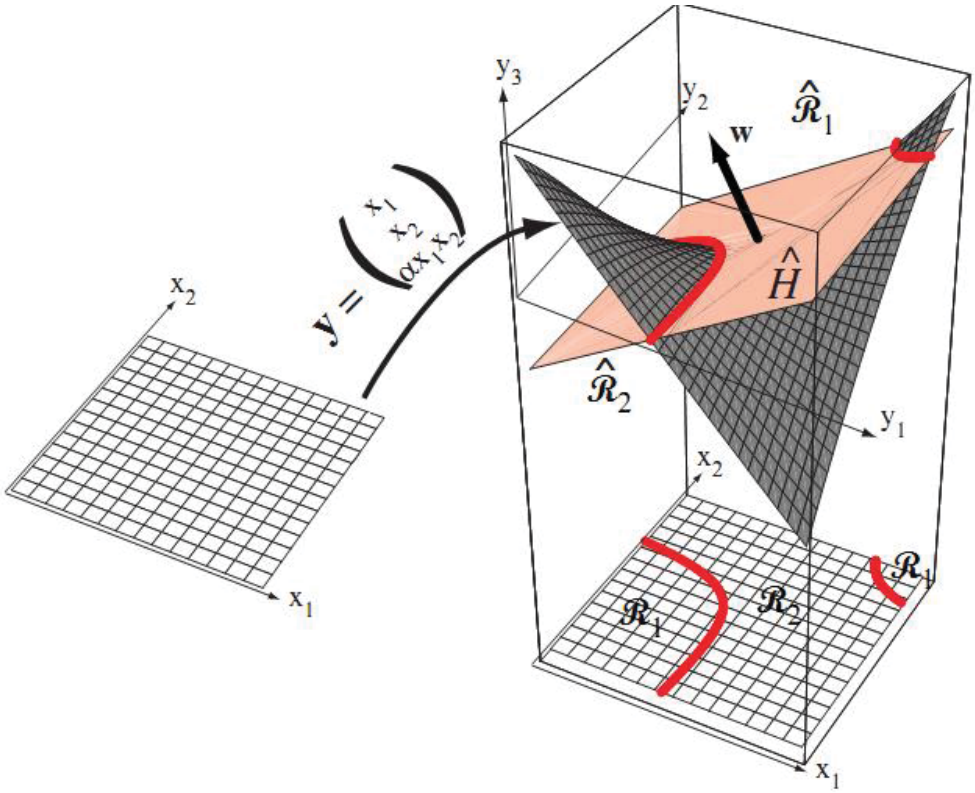
(۱۰)

y

و y بردار ویژگی افزوده شده نامیده می شود. همچنین ، بردار وزنی افزوده شده به صورت زیر نوشته می شود :

(۱۱)

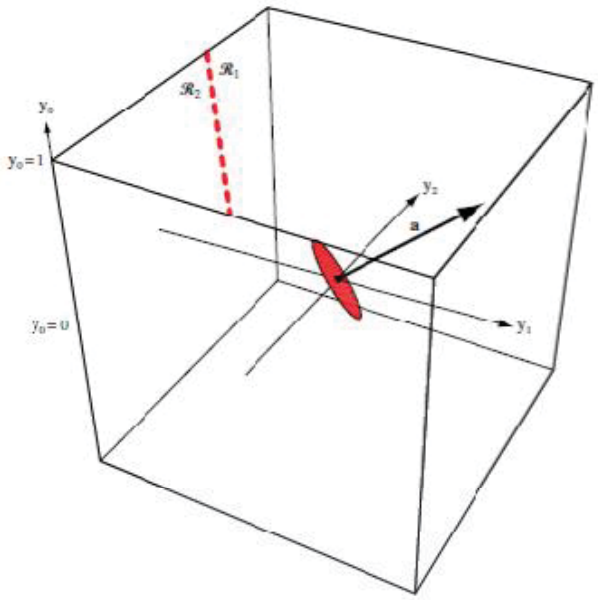
a



شکل ۵-٦ : فضای ورودی دو بعدی x که تابع چند جمله ای f را به y نگاشت می کند. تگاشت ها عبارتند از : $y_1 = x_1$ و $y_2 = x_2$ و $y_3 \propto x_1 x_2$ یک متمایز ساز خطی در فضای تبدیل یافته یک ابرصفحه است که به سمت سطوح کاهش می یابد. نقاط سمت مثبت ابرصفحه $\bar{H}$ مطابق با کلاس w_1 است و سمت دیگر آن w_2 است. در اینجا برای جملات فضای x ، R_1 یک پیوستگی پیچیده است.

نگاشت d بعدی فضای x به فضای u با d+۱ بعد از نظر ریاضی کم اهمیت است اما به آسانی کامل است . علاوه بر مولفه های ثابت به x ، همه ارتباطات بین نمونه ها را حفظ می کنیم. نتایج بردار های فضای y در زیر فضای d بعدی اشتباه است چون خود

آن فضای xمی باشد. سطح ابرصفحه تصمیم گیری تعریف شده $\bar{H}$با $a^t y = 0$ به فضای y اولیه می رود . حتی ابر صفحه تطبیقی H می تواند هر موقعیت در فضای xباشد. فاصله از y به $\bar{H}$با $\frac{|a^t y|}{\|a\|}$داده شده است و یا$\frac{|g(x)|}{\|a\|}$. از انجا که $\|a\| > \|w\|$داریم این فاصله کمتر و یا بیشتر مساوی با فاصله x تا H است. با استفاده از این نگاشت ، مسائل را برای پیدا کردن یک بردار وزنی a و یک وزن آستانه ای w.به پیدا کردن مسائل بردار وزنی واحد a کاهش می دهیم (شکل ۵-۷)



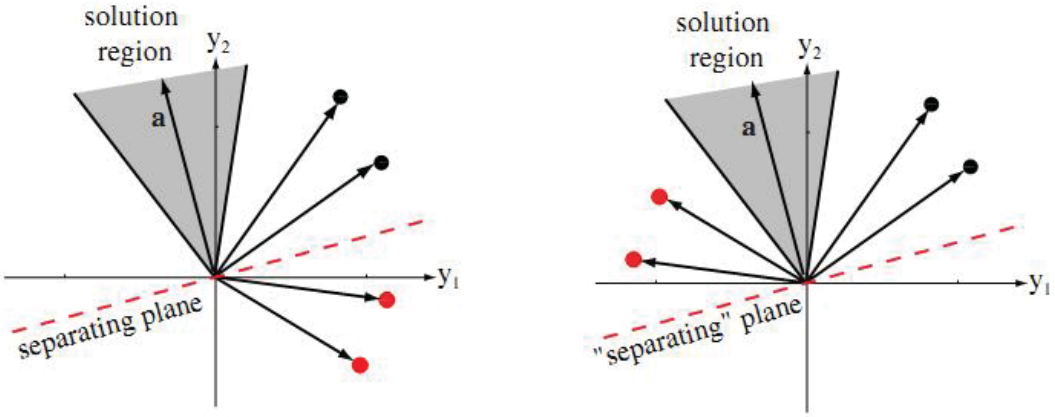
شکل ۵-۷ : فضای ویژگب تخصیصی ۳ بعدی y و بردارهای وزنی تخصیصی . مجموعه نقاط $a^t y = 0$یک صفحه است که یک صفحه عمود به a و عبور از محل تقاطع فضای y که با یک سطح سبز رنگ نشان داده شده است. مانند صفحه ای که محل تقاطع فضای x دو بعدی در بالا است. البته با یک خط نقطه چین نشان داده شده است. بنابراین بردار وزنی تخصیصی a برای هر خط تصمیم گیری در فضای x هدایت می شود.

۵-٤-روش جداسازی خطی دو کلاسی

۵-٤-۱-هندسه و اصطلاحات

فرض کنید که یک مجموعه از n نمونه $y_1, ..., y_n$ را داریم و با ω_1 و یا ω_2 برچسب زده ایم. از این نمونه ها برای تعیین وزن های a در یک تابع تفکیک خطی $g(x) = a^t y$ می خواهیم استفاده کنیم. فرض کنید که برای جواب موجود احتمال خطا کم است. سپس یک روش قابل قبول داشتن یک بردار وزنی برای دسته بندی همه نمونه های درست است. اگر یک بردار وزنی موجود باشد، می توان گفت که نمونه ها به صورت خطی جداشدنی هستند. یک نمونه به صورت درست دسته بندی می شود،

اگر $\mathbf{a}^t\mathbf{y}_i > 0$ و $\mathbf{y}_i$ با ω_1 برچسب بخورند و یا اگر $\mathbf{a}^t\mathbf{y}_i < 0$ و $\mathbf{y}_i$ با ω_2 برچسب بخورند. پیشنهاد نرمال کردن ساده ترین راه برای حالت دو کلاسی است در مقایسه با جایگزین کردن همه نمونه هایی که با ω_2 برای منفی ها برچسب خورده اند . با این نرمالیزه کردن ، برچسب گذاری را کنار می گذاریم و بردار وزنی $\mathbf{a}$ را بطوریکه $\mathbf{a}^t\mathbf{y}_i > 0$ برای همه نمونه ها باشد را نگه می داریم. بردارهای وزنی را بردارهای جدا شدنی و یا به طور معمولی تر یک بردار جواب می نامیم. بردار وزنی $\mathbf{a}$ می تواند به صورت یک نقطه مشخص با فضای وزنی باشد. هر نمونه $\mathbf{y}_i$ به عنوان یک الزام روی موقعیت های ممکن تز یک بردار جواب قرار میگیرد. معادله $\mathbf{a}^t\mathbf{y}_i = 0$ به صورت یک ابرصفحه تعریف می شود بطوریکه مبدا فضای وزنی با داشتن $\mathbf{y}_i$ به عنوان یک بردار نرمال است. بردار جواب اگر موجود باشد باید در سمت مثبت از هر طرف ابرصفحه باشد.بنابراین یک بردار جواب باید در فصل مشترک n نیم صفحه باشد. هر بردار از این ناحیه ، یک بردار جواب است. ناحیه منطبق تاحیه جواب نامیده می شود و نباید با ناحیه تصمیم گیری در فضای ویژگی منطبق به هر کلاس ویژه تداخل داشته باشد. یک مثال یک بعدی از ناحیه جواب برای هر دو حالت نرمال شده ونرمال نشده در شکل ۵–۸ نشان داده شده است.



شکل ۵–۸ : ٤ نمونه آموزشی (سیاه برای ω_1 و قرمز برای ω_2)و ناحیه جواب برای فضای ویژگی را داریم. شکل سمت چپ داده های سطری را نشان می دهد . بردارهای جواب یک طرح از الگو های جدا شدنی را برای دو کلاسی نشان می دهد در شکل سمت راست ، نقاط قرمز نرمال شده اند. به طور مثال با تغییر در علامت. حالا بردار جواب به یک صفحه که همه نقاط نرمال شده در یک طرف هستند هدایت می شوند.

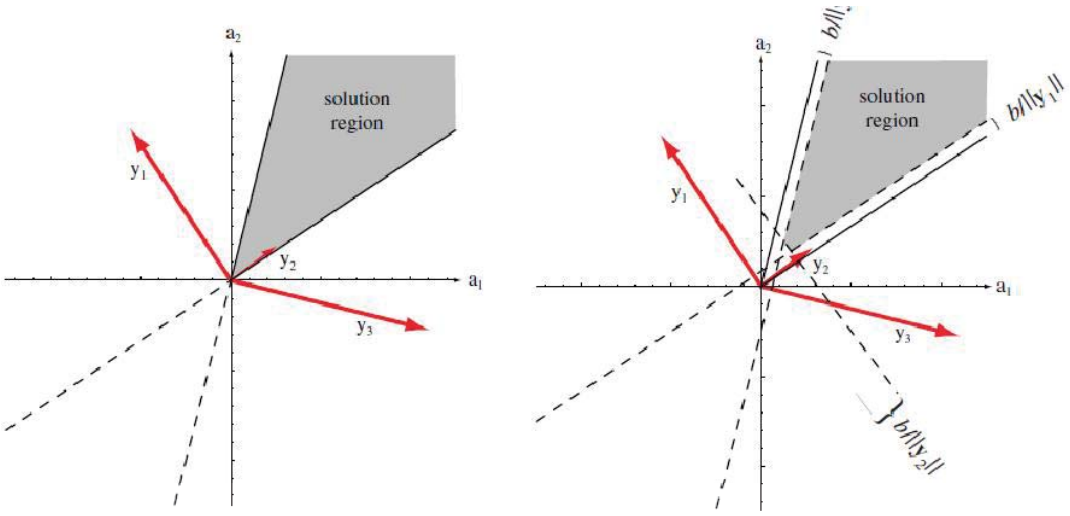
از این بحث کاملا مشخص است که بردار های جواب ف مجددا ، اگر موجود باشند یکتا نیستند. چندین روش برای موارد اضافی تحمیل شده وجود دارند تا یک بردار جواب بدست بیاید. یک احتمال ، جستجوی بردار وزنی با طول واحد است که فاصله مینیمم نمونه های صفحه جدا شدنی را ماکزیمم می کند. احتمال دیگری برای جستجوی بردار وزنی با طول مینیمم با پذیرفتم $\mathbf{a}^t\mathbf{y}_i \geq b$ برای هم $\mathbf{i}$ ها است که در آن $\mathbf{b}$ یک ثابت مثبت است که مرز نامیده می شود. همانطور که در شکل ۵–۹ نشان داده شده است نتایج نواحی جواب فصل مشترک نیم صفحه هایی با شرط $\mathbf{a}^t\mathbf{y}_i \geq b > 0$ را شکل می دهد که ناحیه جواب قبلی برای آن اشتباه بوده است، که با توجه به مرز های قدیمی با فاصله $b/\|\mathbf{y}i\|$ اتفاق افتاده است.

$$\mathbf{a}(k+1) = \mathbf{a}(k) - \eta(k)\,\nabla J(\mathbf{a}(k)) \qquad (16)$$

با این تلاش برای پیدا کردن بردارهای نزدیک به میانه ناحیه جواب یک جواب نتیجه شده برای طبقه بندی نمونه های بهسازی شده تست جدید می باشد. در بیشتر زمینه هایی که بحث شده است به هر حال ما باید هر جوابی را در میان ناحیه جواب قبول

کنیم. مفهوم اصلی این است که یک روند تکراری استفاد هشده که همگرا نیست با یک نقطه محدود در مرز است. مسئله این است که از معرفی هر مرز جلوگیری کنیم با توجه به $\mathbf{a}^t\mathbf{y}_i \geq b > 0$ برای همه i ها.

۵-٤-۲-فرایند نزولی شیب :



شکل ۵-۹ : تاثیر حاشیه ها روی ناحیه جواب. در سمت چپ حالت هیچ مرزی (b=۰) معادل با حالتی مشابه نشان داده شده در در شکل ۵-۸ می باشد. در سمت راست با حالت $b > 0$ ناحیه جواب با مرز $b/\|\boldsymbol{y}_i\|$ کاهش یافته است.

یک راه پیدا کردن جواب ، مجموعه ای از نابرابری های خطی ++++ تعریف شده با تابع معیار استاندارد J(a) مینیمم شده است که a یک بردار جواب است. مسئله کاهشی مینیمم کردن تابع مقیاس شده است به عنوان یک مسئله برای حل کردن فرایند نزولی شیب است. شیب نزولی اصلی خیلی ساده است. با یک انتخاب دالخواه از a(۱) و محاسبه بردار گرادیان $\nabla J(\mathbf{a}(۱))$شروع می کنیم. مقدار بعدی a(۲) با انتقال بعضی از فاصله ها از a(۱) در سراشیبی نزولی مستقیم می باشد به طور مثال در میان گرادیان نزولی. به طور معمول ، a(k+۱) با a(k) طبق معادله زیر بدست می آید.

$$\mathbf{a}(k+۱) = \mathbf{a}(k) - \eta(k)\,\nabla J(\mathbf{a}(k)) \qquad (۱۲)$$

که در آن η فاکتور مقیاس مثبت شده است و یا نرخ یادگیری که با اندازه مراحل تنظیم شده است. ما امید واریم که دنباله بردارهای وزنی با مینیم کردن جواب J(a) همگرا خواهد شد. شکل الگوریتم به صورت زیر است :

Algorithm ۱ (Basic gradient descent)
۱.begin initialize a, criterion θ, η(.), K =۰
۲. do K ← K+۱
۳. a ← a –η(K)∇J(a)
٤. until η(k)J(a)<
٥.return a
٦.end

مسائل بسیاری با فرآیند نزولی گرادیان های به هم پیوسته شناخته شده اند. خوشبختانه ما باید تابع ای را که می خواهیم مینیمم کنیم را بسازیمو باید از مشکلات جدی پرهیز کنیم. یک راه مواجه شدن با آن تکرار کردن است. به هر حال با انتخاب نرخ یادگیری $\eta(k)$ است. اگر $\eta(k)$ خیلی کوچک باشد همگرایی آرام است . ولی اگر $\eta(k)$ خیلی بزرگ باشد، فرآیند بهسازی مختل می شود و حتی واگرا می شود. ما یک روش برای تنظیم نرخ یادگیری را بررسی می کنیم . فرض کنید که تابع معیار استاندارد با بسط مرتبه دوم و مقدار a(k) تخمین زده خواهد شد.

$$J(a) \simeq J(a(k)) + \nabla J^t \left(a - a(k)\right) + \frac{۱}{۲} (a - a(k))^t H(a - a(k)) \qquad (۱۳)$$

که در آن H ماتریس Hessian از مشتق جزئی مرتبه دوم $\partial^۲ J/\partial_{ai}\partial_{aj}$ است که با a(k) ارزیابی می شود. سپس با جانشینی a(k+۱) در معادله ۱۲ به معادله ۱۳ می رسیم :

$$J(a(k+۱)) \simeq J(a(k)) + \eta(k)\|\nabla J\|^t + \frac{۱}{۲} \eta^۲ (k)\nabla J^t H \nabla J$$

با دنبال کردن این مسئله که در آن J(a(k+۱)) با انتخاب زیر می نیمم می شود :

$$\eta K = \frac{\|\nabla J\|^2}{\nabla J^t H \nabla J} \qquad (۱٤)$$

که در آن H به a بستگی دارد و بطور غیر مستقیم به k بستگی دارد. سپس با انتخاب $\eta(k)$ فرضیه مورد توجه استفاده می شود. توجه کنید که اگر تابع معیار استاندارد J(a) به صورت مربعی در سرتاسر ناحیه مورد نظر است ، سپس H ثابت است و η یک ثابت وابسته به k است. یک روش تکراری بدست آمده بدون توجه به معادله ۱۲ و انتخاب a(k+۱) برای مینیمم کردن بسط مرتبه دوم است. که در الگوریتم نیوتن در خط ۳ از الگوریتم ۱ با عبارت زیر جایگزین می شود :

$$\mathbf{a}(k + ۱) = \mathbf{a}(k) - \mathbf{H}^{-1} \nabla J \qquad (۱٥)$$

الگوریتم ۲ با شیب نزولی نیوتن به صورت زیر انجام می شود :

Algorithm ۲ (Newton descent)

۱. **begin initialize** a, criterion**θ**

۲. **do**

۳. a ← a $-H^{-1}\nabla J(a)$

٤. **until** $H^{-1}\nabla J(a)$

٥. **return** a

٦. **end**

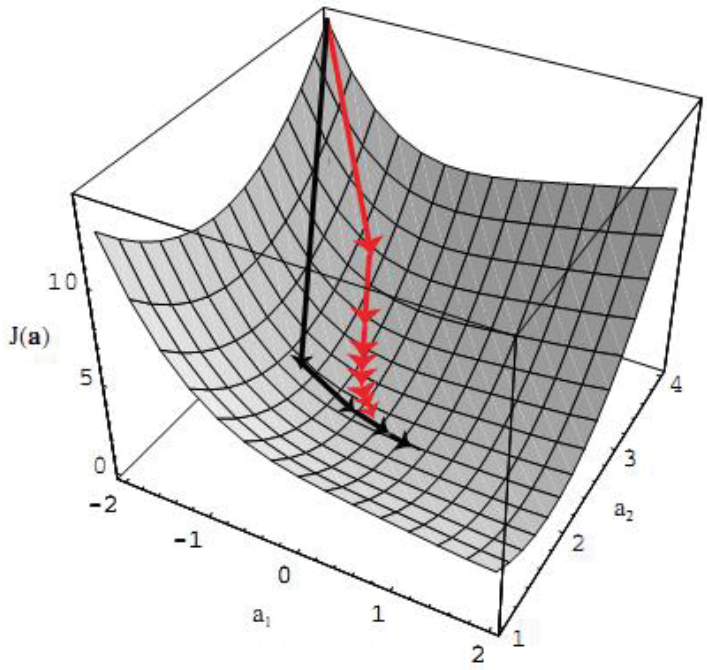
شیب نزولی نمونه و الگوریتم نیوتن در شکل ٥-١٠ مقایسه شده است. به طور معمول ، الگوریتم نیوتن با یک بهبود بهتر در هر مرحله نسبت به الگوریتم گرادیان نزولی نمونه می دهد. به هر حال الگوریتم نیوتن اگر ماتریس Hessian به نام H منحصر به فرد باشد ، کاربردی نیست. علاوه بر این ، وقتی H غیر منحصر به فرد باشد ، زمان $O(d^3)$ برای معکوس ماتریس روی هر تکرار به آسانی با سود نزولی جبران خواهد شد.در حقیقت ، اغلب زمان کمتری نسبت به تنظیم $\eta(k)$ برای ثابت η که کمتر مورد نیاز است ، می باشد و ساختن یک بهسازی کمتر برای محاسبه بهینه $\eta(k)$ در هر مرحله می باشد. (تمرین کامپیوتری ١)

٥-٥- مینیمم کردن تابع استاندارد پرسپترون

٥-٥-١- تابع استاندارد پرسپترون

حال مسئله ساختن تابع معیار استاندارد برای حل کردن نابرابری خطی $\mathbf{a}^t\mathbf{y}_i > 0$ را بررسی می کنیم. بیشترین انتخاب واضح است که $J(\mathbf{a}; \mathbf{y}_1, ..., \mathbf{y}_n)$ ، تعداد نمونه های اشتباه دسته بندی شده با a باشد. به هر حال این تابع یک ثابت تکه ای است ، که یک شرایط ضعیفی برای جستجوی گرادیان می باشد. یک انتخاب بهتر تابع معیار استاندارد پرسپترون است.

J_1 (١٦)



شکل ۵-۱۰ : دنباله بردارهای وزنی با روش گرادیان نزولییک نمونه قرمز رنگ و الگوریتم نیوتن مرتبه دوم سیاه رنگ. روش نیوتن در هر مرحله بهبود را همراه دارد. حتی زمانیکه نرخ یادگیری بهینه برای هر دو روش استفاده می شود. به هر حال اضافه کردن محاسبات وزنی معکوس ماتریس Hessian درروش نیوتن معمولا مورد توجه نیست و نزولی نمونه ها کافی است.

از آنجا که $Y(a)$ ، مجموعه ای از نمونه های دسته بندی شده اشتباه با a است. (اگر هیچ نمونه ای اشتباه دسته بندی نشود، Y خالی است و J_p صفر می شود.) از آنجا که $a^t y \leq 0$ را داریم اگر y اشتباه دسته بندی شود، $J_p(a)$ هرگز منفی نمی شود. ، صفر می شود اگر a یک بردار جواب باشد، یا اگر a یک مرز تصمیم گیری باشد. از نظر هندسی ، $J_p(a)$ ، متناسب با جمع فاصله های نمونه های اشتباه دسته بندی شده برای مرز تصمیم گیری است. شکل ۵-۱۱ J_p برای یک نمونه مثال دو بعدی نشان می دهد.

$$\nabla J_p = \sum_{y \in Y}(-y) \qquad (۱۷)$$

و از اینجا به بعد با قانون به روز کردن داریم :

$$a(k+1) = a(k) + \eta(k)\sum_{y \in Y_k} y \qquad (۱۸)$$

که در آن Y_k یک مجموعه نمونه های اشتباه دسته بندی شده با $a(k)$ است. بنابراین الگوریتم پرسپترون به صورت زیر می شود :

Algorithm ۳ (Batch Pereceptron)

۱. **begin initialize** a, $\eta(.), criterion\ \theta, k = 0$

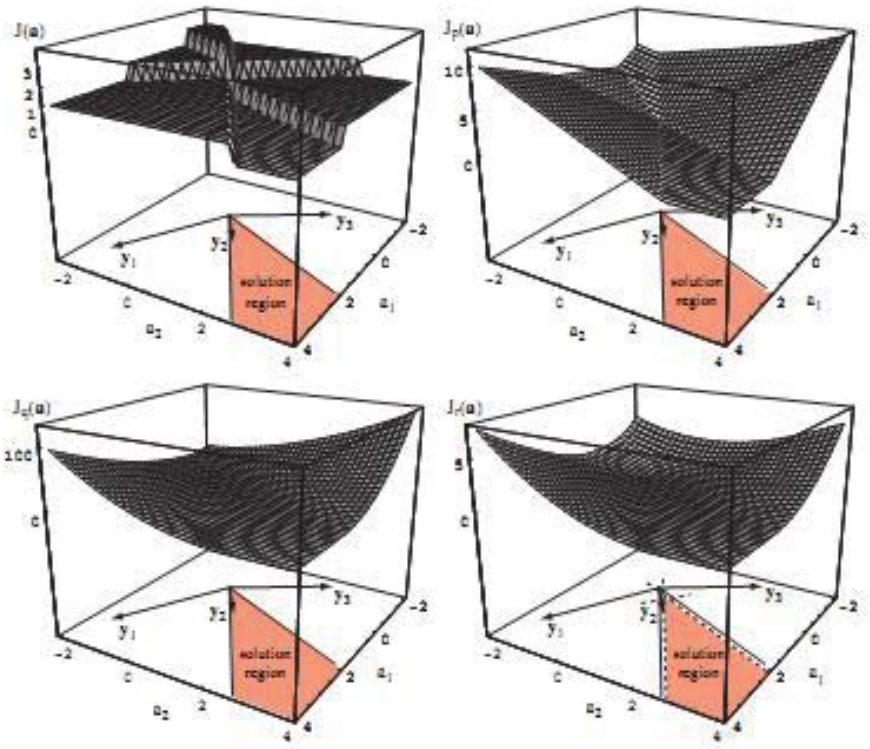
۲. **do** k ← k+1

۳. a ← a+η(k) $\sum_{y \in y_k} y$

٤. **until** η(k) $\sum_{y \in y_k} y < \theta$

٥. **return** a

٦.**end**



شکل ۵-۱۲: معیار استاندارد پرسپترون J_p به عنوان یک تابع از بردارهای وزنی a_1 و a_2 برای مسئله ۳ الگو رسم شده اند. بردارهای

وزنی صفر می شوند و الگوریتم به ترتیب با بردارهای مساوی برای نرمال کردن الگوهای اشتباه دسته بندی شده اضافه می شوند. در مثال نشان داده شده $\mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_1, \mathbf{y}_3$ هر بار که بردارها در ناحیه جواب اشتباه هستند و تکرار خاتمه می یابد. با بروز رسانی مرحله دوم بردارهای دورتر از ناحیه جواب را بهد از اولین بروز رسانی نامزد کرده و بدست می آوریم. طبق قضیه ۵-۱. (در یک تکرار با روش ئسته ای همه نقاط اشتباه دسته بندی شده برای هر مرحله تکرار با مسیر آرام تر به فضای وزنی هدایت می شوند.)

۵-۵-۲- اثبات همگرایی برای تصحیح نمونه های ساده :

ما باید یک تست ویژگی های همگرایی از الگوریتم پرسپترون با یک تنوع ساده تر برای انالیز بیاوریم. با تست کردن بیشتر $a(k)$ روی همه نمونه ها و بر اساس بهسازی مجموعه Y_kاز نمونه های آموزشی اشتباه دسته بندی شده ، باید نمونه های یک دنباله را بررسی کنیم و بردار های وزنی را حتی یک نمونه ساده اشتباه دسته بندی می شود را تغییر دهیم. هدف از اثبات این همگرایی جزئیات دنباله هایی است که به اندازه هر نمونه دیده شده در دنباله نامحدود مهم نیستند. ساده ترین راه تضمین می کند که تکرار نمونه ها به صورت چرخشی و دایره وار است. از یک نقطه نظر انتخاب تصادفی اغلب ترجیح داده شده است. (بخش ۵-۸-۲). واضح است که نه نمونه های تک ونه نمونه های دسته ای از الگوریتم پرسپترون ، متصل نیستنداز آنجا که باید همه الگو های آموزشی را ذخیره کنیم و به طور بالقوه مجددا بررسی کنیم.دو ساده سازی که در روشن ساختن این توضیح کمک می کند عبارتند از : اول ، ما باید موقتا توجه خود را به روشی که در آن $\eta(k)$ ثابت است جلب کنیم که به آنروش افزایشی- ثابت نامیده می شود. بنظ می رسد از معادله (۱۸) که اگر $\eta(t)$ ثابت باشد ، خوب بودن فقط برای مقیاس پذیری نمونه ها است بنابراین روش افزایشی – ثابت با $\eta(t) = ۱$ می تواند بدست آید بدون اینکه هیچ تلفاتی داشته باشد. در ساده سازی دوم فقط شامل علامت ها است. وقتی که نمونه ها به ترتیب بررسی می شوند، بعضی از دسته بندی ها اشتباه انجام خواهد شد. از آنجا که ما باید فقط بردارهای وزنی را تغییر می دهیم وقتی که خطا وجود دارد باید به نمونه های اشتباه دسته بندی شده توجه کنیم. بنابراین باید دنباله ای از نمونه ها را با استفاده از برجسته بودن مشخص کنیم به طور مثال : $\mathbf{y}^1, \mathbf{y}^2, ..., \mathbf{y}^k, ...,$ از آنجا که هر $\mathbf{y}^k$ یکی از n نمونه $\mathbf{y}_1, \mathbf{y}_n ...,$ می باشد و هر $\mathbf{y}^k$ اشتباه دسته بندی می شود. برای مثال اگر نمونه ها ی $\mathbf{y}^1, \mathbf{y}^2$ و $\mathbf{y}^3$ به صورت چرخشی بررسی شوند و اگر نمونه های مارک شده اشتباه دسته بندی شده باشند،

(۱۹)

سپس دنباله $\mathbf{y}^1, \mathbf{y}^2, \mathbf{y}^3, \mathbf{y}^4, \mathbf{y}^5, ...$ دنباله $\mathbf{y}_1, \mathbf{y}_3, \mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_2, ...$ را مشخص می کند. با این مفهوم ، قانون افزایشی-ثابت برای تولید یک دنباله بردار وزنی به صورت زیر نوشته می شود :

(۲۰)

$$\left.\begin{array}{ll} X(1) & \text{arbitrary} \\ A(k+1) = a(k) + y^k & K \geq 1 \end{array}\right\}$$

که در آن $\mathbf{a}^t(k)\mathbf{y}^k \leq ۰$ برای همه k ها برقرار است. اگر تعداد کل الگو ها را مشخص کند الگوریتم به صورت زیر می شود:

Algorithm٤ (Fixed – increment single – sample Perceptron)

١.begin initialize a, k=٠

٢. do k ← (k+١) modn

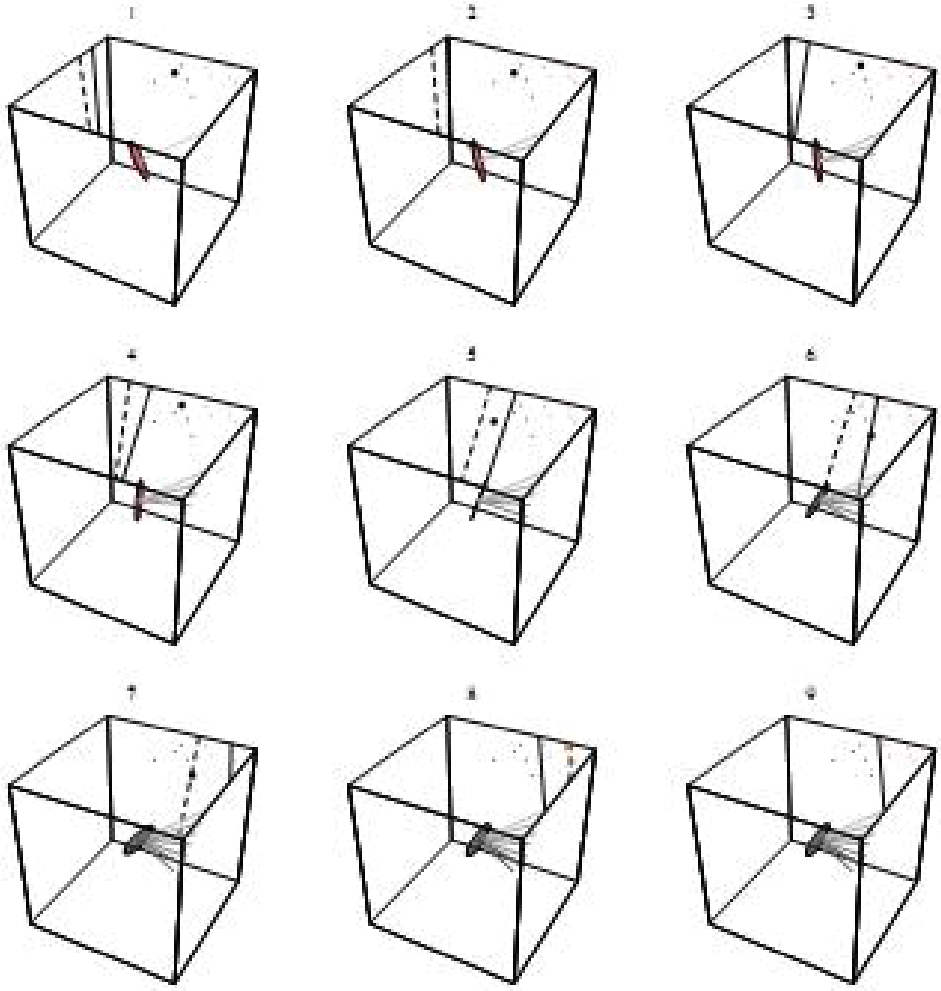
٣. if y_k is misclassified by a then a a ← a- y_k

٤. until all patterns properly classified

٥.return a

٦. end

قانون پرسپترون افزایشی–ثابت ، از ساده ترین الگوریتم هایی است که برای حل سیستم های خطی نابرابر پیشنهاد شده است. از نظر هندسی ، تفسیر فضای وزنی کاملا آشکار است. از آنجا که a(k) دسته بندی اشتباه $\mathbf{y}^k$ می باشد ، a(k) یک سمت مثبت از صفحه $\mathbf{y}^k$ با شرط $\mathbf{a}_t\mathbf{y}_k = ٠$ نیست. با اضافه شدن $\mathbf{y}^k$ به a(k) بردارهای وزنی به طور مستقیم به سمت ابر صفحه حرکت می کنند. آیا اینکه این صفحه متقازع است یا خیر ، ضرب داخلی جدید $\mathbf{a}_t(k+١)\mathbf{y}_k$ بزرگتر از ضرب داخلی قدیمی $\mathbf{a}_t(k)\mathbf{y}_k$ با مقدار $\|y^k\|^٢$می باشدو بهسازی به طور آشکار با بردارهای وزنی پیشرفت می کند.



شکل ۵-۱۳ : نمونه های دو کلاسی ω_1 (مشکی) و ω_2 (قرمز) در فضای ویژگی افزایشی با بردارهای وزنی a افزایشی نشان داده شده اند.در هر مرحله از قانون افزایشی ثابت یکی از الگو های اشتباه دسته بندی شده $\mathbf{y}^k$ با نقطه بزرگی نشان داده شده است. یک بهسازی $\Delta\mathbf{a}$ با بردارهای وزنی اضافه شده است. برای یک نقطه ω_1 یا یک نقطه دورتر برای ω_2. بنابراین تغییرات مرز تصمیم گیری با موقعیت خط تیره توپر است . دنباله نتایج بردارهای a نشان داده شده استو مقادیر بعدی تیره تر نشان داده شده است. در این مثال با ۹ مرحله یک بردار جواب بدست می آیدو گروه ها به با موفقیت با نشان دادن مرزهای تصمیم گیری جدا می شوند.

واضح است که این الگوریتم اگر نمونه ها به صورت خطی جدایی پذیر باشند خاتمه می یابد . حال درستی خاتمه چیدا کردن نونه ها را وقتی به صورت خطی جدایی پذیرند اثبات می کنیم.

قضیه ۵-۱ (همگرایی پرسپترون.) اگر نمونه های آموزشی به صورت خطی جدایی پذیر باشند ، سپس دنباله ای از بردارهای وزنی با الگوریتم ٤ و با بردار جواب a خاتمه خواهند یافت.

اثبات :

در جستجوی اثبات : سعی می کنی مکه نشان دهیم که هر بهسازی با بردارهای وزنی نزدیک به ناحیه جواب بدست می آید.بنابراین ممکن است سعی کنیم به نشان دادن اینکه اگر $\hat{a}$ هر بردار جوابی باشد ، سپس $\|a(k+1)-\hat{a}\|$ کمتر از $\|a(k)-\hat{a}\|$ است . از زمانیکه این با توجه به درست بودن نتیجه مطلوبی را ندهد باید بررسی کنیم که بردارهای جواب بعد از یک مدت طولانی قابل قبول هستند.$\hat{a}$ برای هر بردار جوابی داشته باشیم ، بطوریکه$\hat{a}^t y_i$ برای همه i ها مثبت باشد و α یک فاکتور مقیاس مثبت باشد. از معادله ۲۰ داریم :

$$\mathbf{a}(k+1)-\alpha\hat{a}=(\mathbf{a}(k)-\alpha\hat{a})+\mathbf{y}^k$$

و از این پس :

||

از آنجا که $\mathbf{y}^k$ اشتباه دسته بندی می شود ، جمله دوم در سه بار پایان می یابد اگر $\mathbf{a}^t(k)\mathbf{y}^k \le 0$ به اندازه کافی بزرگ باشد. اگر B ماکزیمم طول از یک بردار الگو را داشته باشد:

β (۲۱)

وγ کمتر از ضرب داخلی بردارهای جواب با هر بردار الگویی است به طور مثال :

$$\gamma=\min_i[\hat{a}^t y_i] > 0, \qquad (۲۲)$$

سپس عدم تساوی زیر را داریم :

||

اگر انتخاب کنیم :

(۲۳)

α

بدست می آوریم :

||

بنابراین مربعات فاصله از $a(k)$ به $\alpha\hat{a}$ با حداقل β^2 برای هر بهسازی کاهش می یابد و بعد از k بهسازی داریم :

|| (۲۴)

از آنجا که مربعات فاصله منفی نمی شود، دنباله ای از بهسازی ها بعد از کمتر از k_0 بهسازی خاتمه می یابد

k_0 (۲۵)

از آنجا یک بهسازی وقتی اتفاق می افتد که یک نمونه اشتباه دسته بندی شده باشدو از آنجا که هر نمونه در یک دنباله محدود نمی باشد ، آن وقتی اتفاق می افتد که بردارهای وزنی نتیجه شده برای همه نمونه های به درستی دسته بندی شود. تعداد $\mathbf{a}(1)=0$ یک محدوده روی تعداد بهسازی ها داده می باشد. اگر $a(1)=0$ باشد ، عبارت نمونه ویژه ای را برای k_0 داریم :

(۲۶)

k_0

مخرج معادله ۲۶ سختی این مسئله را نشان می دهد که به صورت اساسی با نمونه های نزدیک عمود بر هم با بردارر جواب تعیین می شود. متاسفانه این کمکی نمی کند وقتی یک مسئله قابل حل نیست چرا که این مرزها در ترم هایی از بردار جواب شناخته

شده نیست.معمولا مشخص است که مشکل جدا سازی خطی می تواند با مشکلات دلخواه برای حل کردن نمونه های ساخته شده با تقریبا یک صفحه انجام شود. با این وجود ، اگر نمونه های آموزشی به صورت خطی جدایی پذیر باشند، قانون افزایشی – ثابت برای یک جواب بعد از تعداد محدودی از بهسازی ها بازدهی دارد.

۵-۵-۳- تعمیم دهی مستقیم

قانون افزایشی ثابت برای تتنوع الگوریتم ها تعمیم داده می شود. ما باید دو نوع مخصوص آن را بررسی کنیم . نوع اول با افزایش متغیر $\eta(k)$ و a مرزی برای b معرفی می شود که برای تصحیح $\mathbf{a}^t(k)\mathbf{y}^k$ می باشد زمانیکه جدا کردن مرزها به درستی انجام نمی شود.

$$\left.\begin{array}{ll} \mathbf{a}(۱) & \text{arbitrary} \\ \mathbf{a}(k+۱) = \mathbf{a}(k) + \eta(k)\mathbf{y}^k & \\ k \geq ۱, & \end{array}\right\} \qquad (۲۷)$$

که در آن شرط $\mathbf{a}^t(k)\mathbf{y}^k \leq b$ برای همه k ها برقرار است. بنابراین برای n الگو ، الگوریتم زیر را داریم :

Algorithm ۵ (Variable increment Perceptron with margin)

۱. **begin initialize** a, criterion $\theta, margin\ b, \eta(.), k = 0$

۲. **do** k ← k+۱

۳. **if** $a^t y_k + b < ۰$ *then* $a \leftarrow a - < \eta(k) y_k$

۴. **until** $a^t y_k + b \leq 0$ *for all* k

۵. **return** a

۶.**end**

می توان آن را نشان داد به شرط آنکه نمونه ها به صورت خطی جدا شدنی باشند واگر

$$\eta(k) \geq ۰ \qquad (۲۸)$$

(۲۹)

$$\lim_{n} \qquad (۳۰)$$

پس همگرایی $a(k)$ با بردار جواب a برای همه i ها با شرط پذیرفته شده $\mathbf{a}^t\mathbf{y}_i > b$ بدست می آید. با توجه به این شرایط مقدار $\eta(k)$ قابل قبول است اگر $\eta(k)$ یک مقدار مثبت ثابت باشد و یا اگر به صورت نزولی $1/k$ باشد.

یک نوع دیگر از الگوریتم نزولی شیب برای J_p عبارلت است از :

$$\left.\begin{array}{ll} \mathbf{a}(1) & \text{arbitrary} \\ \mathbf{a}(k+1) = \mathbf{a}(k) + \eta(k)\mathbf{y}^k & \\ k \geq 1 & \end{array}\right\} \qquad (31)$$

که در آن Y_k مجموعه ای از نمونه های آموزشی با دسته بندی اشتباه بوسیله $a(k)$ است . این الگوریتم همچنین برای یک جواب سازماندهی شده کارایی دارد که اگر $\hat{a}$ یک بردار جواب برای $\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n$ باشد سپس دسته بندی به درستی برای بردار بهسازی انجام می شود :

$$y$$

با توجه به جزئیات بیشتر الگوریتم به صورت زیر می شود :

Algorithm ٦ (Batch variable increment Pereceptron)

١. **begin initialize** a, $\eta(.), k = 0$
٢. **do** k ← k+1
٣. y_k = {}
٤. j = ٠
٥. **do** j ← j+١
٦. **if** y_j *is misclassified then Append* y_j **to** y_k
٧. **until** j=n
٨.a ← a +η(k) $\sum_{y \in y_k} y$
٩.**until** y_k = {}
١٠. **return** a
١١. **end**

فایده دسته نزولی شیب ، یک مسیر بردار وزنی هموار است که با الگوریتم تک نمونه ای متناظر مقایسه می شود. (الگوریتم ٥). از آنجا که بهینه سازی یک مجموعه کامل از الگو های دسته بندی شده نادرست استفاده شده است ، باتغییر آماری محلی در الگو های دسته بندی شده نادرست تمایل به صفر شدن برای این الگوها است طمانیکه یک گرایش با مقیاس گسترده وجود ندارد. بنابراین نمونه ها به صورت خطی جدا شدنی هستند و همه بردارهای تصحیح ممکن در یک مجموعه جداشدنی خطی شکل می یابند. و اگر η(k) برای معادلات ٢٨-٣٠ پذیرفته شود ، دنباله بردارهای وزنی با الگوریتم نزولی گرادیان برای $J_p(٠)$ با بردار جواب همگرا خواهد شد. این نکته جالب است که شرط η(k) پذیرفته می شود اگرη(k) یک ثابت مثبتی باشد. و اگر با η(k) کاهش یابد و یا حتی با k افزایش یابد، می توان ترجیح دادکه ١/k در هر بار کوچکتر شود. این ویژگی درست است اگر دلیلی مبنی بر اینکه نمونه ها به صورت خطی جدا نمی شوند وجود داشته باشدو این باعث کاهش اثر مخرب نمونه های نادرست و معیوب می شود. به هر حال در زمینه های جدا شدنی ، این واقعیتی است که می توان اجازه دادη(k) بزرگتر شود و با این حال هنوز یک جواب بدست بیاید. با این مشاهدات تفاوت هایی بین حالت های تئوری و عملی ایجاد می شود. از نقطه نظر تئوری ، جالب است که یک جواب را با تعداد محدودی از مراحل برای هر مجموعه از نمونه های جدا شدنی ، هر بردار وزنی اولیه a(١) ، هر مرز غیر منفی از b و هر فاکتور مقیاس مورد قبول در معادلات ٢٨-٣٠ بدست بیاوریم . از نقطه نظر عملی ، یک انتخاب

عاقلانه ای را برای هر کمیت می خواهیم ایجاد کنیم . با بررسی کردن مرز bبطور مثال . اگر b کوچکتر از $\eta(k)=\|y_k\|^2$ باشد، میزان درستی با $\mathbf{a}^t(k)\mathbf{y}^k$ افطایش می یابد، واضح است که تاثیر کمی برای همه دارد. اما اگر b بزرگتر از $\eta(k)=\|y_k\|^2$ باشد ، بهسازی های بسیاری برای پذیرفتن شرایط $\mathbf{a}^t(k)\mathbf{y}^k > b$ نیاز خواهد شد. یک مقدار نزدیک به $\eta(k)=\|y_k\|^2$ مفید می باشد. با توجه به انتخاب های $\eta(k)$ وb ، مقیاس پذیری کولفه هایی از $\mathbf{y}^k$ همچنین تاثیر بسیاری در نتایج دارند . با داشتن قضیه همگرایی نمی توان فکر استفاده ازاین تکنیک را حذف کرد. یک الگوریتم پرسپترون نزولی نزدیک ، الگوریتم Winnow است که کاربردهایی برای داده های آموزشی دارد. یک تفاوت کلیدی است زمانیکه بردارهای وزنی با الگوریتم پرسپترون بر می گردند دارای مولفه هایa_i $(i = ۰, ...d)$در Winnow است و مطابق با $۲\sinh[a_i]$ دسته بندی می شود. در یک نسخه از آن ، الگوریتم Winnow متعادل می شود که در آن بردارهای مثبت و منفی جداگانه ای وجود دارد. $\mathbf{a}^+$ و $\mathbf{a}^-$ ، که هر پیوستگی با یکی از دو کلاس ، نشان می دهد که یادگیری انجام شده است. بهسازی بردارهای مثبت فراهم می شود اگر و فقط اگر یک الگوی آموزشی با $\omega_۱$ اشتباه دسته بندی شود.بهسازی روی بردارهای منفی فراهم می شود اگر و فقط اگر یک الگوی آموزشی در $\omega_۲$ اشتباه دسته بندی شود.

Algorithm ۷ (Balanced winnow)

۱. **begin initialize** $\mathbf{a}^+$,$\mathbf{a}^-$, $\eta(.)$, $k \leftarrow 0,\ \alpha > ۱$
۲. **if sign** $[\ a^{+t}y_k - a^{-t}y_k] \neq z_k$ $(Pattern\ misclassified)$
۳. **then if** z_k= +۱ **then** $a_i^+ \leftarrow \alpha^{+yi}a_i^+;\ a_i^- \leftarrow \alpha^{-yi}a_i^-$ **for all i**
٤. **then if** z_k=-۱ **then** $a_i^+ \leftarrow \alpha^{-yi}a_i^+;\ a_i^- \leftarrow \alpha^{+yi}a_i^-$ **for all i**
٥. **return** $\mathbf{a}^+$, $\mathbf{a}^-$
٦.**end**

دو فایده اصلی برای این نسخه از الگوریتم Winnow وجود دارد. اولا در طی آموزش برای هر یک از دو بردار وزنی پیوسته ، در یک جهت یکنواخت حرکت می کنیم و این به معنی "وقفه" است ، که با دو بردار تعیین می شود، هرگز نمی توان اندازه بردارهای جداشدنی را افزایش داد.بدین ترتیب می توان همگرایی را اثبات کرد حتی زمانی که مسائل پیچیده هستند ، با این حال قضیه همگرایی پرسپترون معمول تر است. ثانیا ، همگرایی معمولا سریع تر از پرسپترون است ، از آنجا که تنظیمات مناسبی برای نرخ یادگیری وجود دارد، هر وزن تشکیل دهنده از مقدار نهایی بیشتر نمی شود. این فایده به طور خاص اعلام شده است تا یک تعداد بزرگی از ویژگی های بی ربط یا زائد شناسایی شوند (تمرین کامپیوتر ٦)

٥-٦- فرآیند ملایم سازی

٥-٦-١-الگوریتم نزولی

تابع معیار استاندارد J_p از هیچ میانگینی استفاده نمی کند.فقط تابعی می سازیم که وقتی a بردار جواب است باید مینیمم باشد . یک نسبت متمایز به صورت زیر است که :

$$J(a) = \sum_{y\in y}(a^t y)^2 \qquad (۳۲)$$

که $Y(\mathbf{a})$ مجددا یک مجموعه از نمونه های آموزشی دسته بندی اشتباه با a را نشان می دهد شبیه J_p, J_q که روی نمونه های اشتباه دسته بندی شده تمرکز می کند. لین یک تختلاف مهم است که اینجا گرادیان پیوسته است. در حالیکه گرادیان J_p پیوسته نیست. بنابراین J_q یک سطح هموازی را برای جستج. نشان می دهد. (شکل ۵-۱۱). متاسفانه J_q خیلی آرام به مرز ناحیه جواب نزدیک می شود که دنباله بردارهای وزنی به سوی این مرز می توانند همگرا شوند. بویژه در کند شدن زمان سپری شده برای گرادیان برای دستیابی به نقطه مرزی $a=0$. مشکل دیگر J_q ، مقداری است که با بردارهای نمونه طولانی غالب می شود.از هر دو این مسائل با تابع معیار استاندارد می توان اجتناب کرد :

$$y \qquad (۳۳)$$

که در اینجا $Y(\mathbf{a})$ مجموعه نمونه هایی با شرط $\mathbf{a}^t\mathbf{y} \le b$ است . بنابراین ، $J_r(\mathbf{a})$ هر گز منفی نمی شود و صفر است اگر و فقط اگر $\mathbf{a}^t\mathbf{y} \ge b$ برای همه نمونه های آموزشی باشد. گرادیان J_r به صورت زیر می باشد :

$$\nabla J_r = \sum_{y\in Y}\frac{a^t y - b}{\|y\|^2}y$$

و قانون بهینه سازی

$$\left.\begin{array}{ll} \mathbf{a}(1) & \text{arbitrary} \\ \mathbf{a}(k+1) = \mathbf{a}(k) + \eta(k)\sum_{y\in Y}\frac{b-a^t y}{\|y\|^2}y & \\ k \ge 1 & \end{array}\right\} \qquad (۳۴)$$

بنابراین الگوریتم ملایم سازی به این صورت است که :

Algorithm ۸ (Batch relaxation with margin)

۱.**begin initialize** a, $\eta(.)$, k =۰

۲. **do** k ← k+۱

۳. $y_k = \{\}$

٤. j =۰

۵. **do** j ← j+۱

٦. **if** y_j **is misclassified then Append** y_i **to** y_k

۷. **until** **j = n**

۸. **a** ← **a**+η(**k**) $\sum_{y\in y} \frac{b-a^t y}{\|y\|^2} y$

۹. **until** y_k ={}

۱۰. **return a**

۱۱. **end**

با توجه به قبل ، اثبات همگرایی ساده تر می شود وقتی نمونه ها را در یک پیوستگی بررسی می کنیم.به طور مثال : تک نمونه به حالت دسته ای ترجیح داده می شود. همچنین توجه به حالت افزایشی ثابت $\eta(k) = \eta$ محدود می شود. بنابراین ، مجددا دنباله $\mathbf{y}^1, \mathbf{y}^2, \ldots$ را بررسی می کنیم به گونه ای ه نمونه های آن برای بردارهای وزنی درست باشد.

$$\left.\begin{array}{ll} \mathbf{a}(1) & \text{arbitrary} \\ \mathbf{a}(k+1) = \mathbf{a}(k) + \eta \frac{b - a^t(k) y^k}{\|y^k\|^2} y^k & \\ k \geq 1 & \end{array}\right\} \quad (35)$$

که در آن برای همه k شرط $\mathbf{a}^t(k)\mathbf{y}^k \leq b$ وجود دارد.

Algorithm ۹ (Single – sample relaxation with margin)

۱.**begin** **initialize** **a**, η(.) , **k**=۰

۲.**do** **k** ← **k**+۱

۳. **if** y_k **is misclassified** **then** **a** **a**+η(**k**) $\frac{b-a^t y}{\|y_k\|^2} y_k$

٤. **until all patterns properly classified**

٥.**return a**

٦. **end**

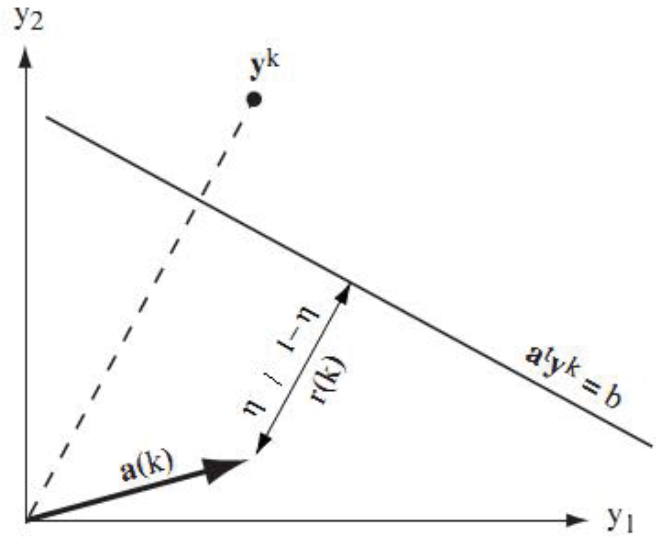
این الگوریتم به عنوان قانون ملایم سازی تک نمونه ای با مرز شناخته شده است و تفسیر هندسی ساده ای دارد. که نرخ آن عبارت است از :

$$r(k) = \frac{b - a^t(k)y^k}{\|y^k\|} \qquad (۳۶)$$

که فاصله از $a(k)$ تا ابرصفحه $a^t y^k = b$ می باشد. از آنجا که $y^k/\|y^k\|$ یک بردار نرمال واحد برای ابرصفحه است ، معادله (۳۵) برای همه $a(k)$ به عدد کسری η از فاصله $a(k)$ تا ابرصفحه حرکت داده می شود. اگر $\eta = ۱$ باشد ، $a(k)$ دقیقا به ابرصفحه منتقل می شود بطوریکه یک کشیدگی با نابرابری $a^t(k)y^k \leq b$ ایجاد می شود که ملایم سازی می باشد. (شکل ۵-۱۴). با یک بهسازی ، از معادله (۳۵) بدست می آوریم :

$$a^t(k+۱)y^k - b = (۱ - \eta)(a^t(k)y^k - b) \qquad (۳۷)$$

اگر $\eta < ۱$ باشد ، سپس $a^t(k+۱)y^k$ هنوز کمتر از b است و اگر $\eta > ۱$ باشد سپس $a^t(k+۱)y^k$ بیشتر از b است. این شرایط به ترتیب به ملایم سازی کمتر و ملایم سازی بیشتر اشاره می کند. به طور معمول ، ما باید بازه η را با $۰ < \eta < ۲$ محدود کنیم. (شکل ۵-۱۴ و ۵-۱۵)

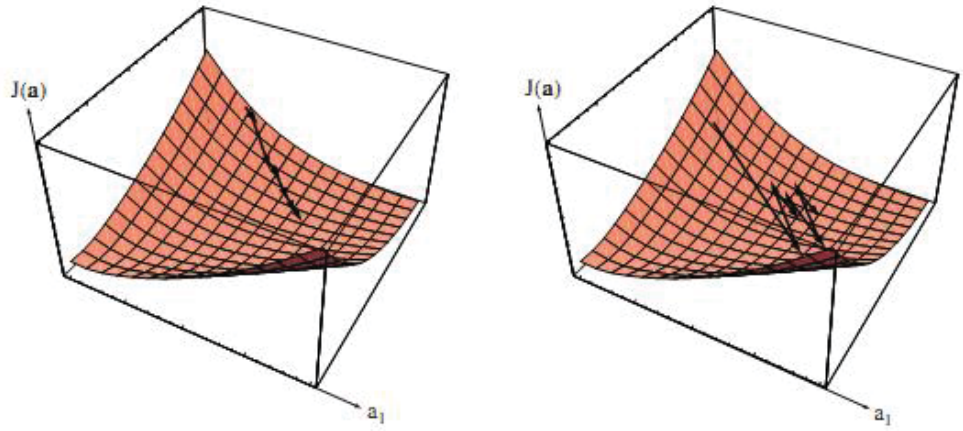


شکل ۵-۱۴ : در هر مرحله از الگوریتم ملایم سازی اصلی ، بردار وزنی با یک نسبت η به سمت ابرصفحه تعریف شده با $a^t y^k = b$ می باشد.

۵-۶-۲- اثبات همگرایی :

وقتی که قانون همگرایی برای یک مجموعه نمونه های خطی جدا شدنی بکار برده می شود، تعداد بهسازی ممکن است محدود باشد و یا نباشد. اگر محدود باشد ، سپس یک بردار جواب بدست می آوریم . اگر محدود نباشد، باید همگرایی $a(k)$ را با یک بردار محدود شده روی مرز از ناحیه جواب بررسی شود. از آنجا که این ناحیه با شرط $a^t y \geq b$ ، نواحی بزرگتر را برای $a^t y >$

۰ دارد اگر $b > ۰$ باشد ، این به این معنی است که $a(k)$ نواحی بزرگتر را حداقل یکبار استفاده می کند، سرانجام باقیمانده ها ، برای همه k بزرگتر از مقدار محدود .k می باشد.



شکل ۵-۱۵ : در سمت چپ ملایم سازی کمتر $\eta < ۱$ به یک نزولی آرام بدون اینکه نیاز باشد هدایت می شود و یا حتی همگرایی اتفاق نمی افتد. ملایم سازی بیشتر ($۱ < \eta < ۲$,) ، در وسط نشان داده شده است)یک تخطی کردن را توصیف می کند .با این حال همگرایی در نهایت بدست خواهد آمد.

اثبات آن وابسته به این حقیقت است که اگر $\hat{a}$ روی هر بردار در ناحیه جواب باشد ، بطور مثال ، بردار پذیرفته شده $\hat{a}^t y_i > b$ برای همه i ها ، سپس هر مرحله $a(k)$ به $\hat{a}$ نزدیک تر می شود. این حقیقت با استفاده از معادله (۳۵) به صورت زیر تعریف می شود :

(۳۸)

و

$$(\hat{a} - \mathbf{a}(k))^t \mathbf{y}^k > b - \mathbf{a}^t(k)\mathbf{y}^k \geq ۰ \qquad (۳۹)$$

بطوریکه

(٤۰)

$$\|$$

از آنجا که η در بازه $۰ < \eta < ۲$ محدود شده است، می توان $\|a(k+۱) - \hat{a}\| \leq \|a(k) - \hat{a}\|$ را داشت. بنابراین بردارهای دنباله $a(۱), a(۲), \ldots$ به $\hat{a}$ نزدیکتر و نزدیکتر می شوند. و در یک محدوده ای مانند k ، به سمت بی نهایت می رود فاصله $\|a(k) - \hat{a}\|$ نزدیک به فاصله محدود $r(\hat{a})$ می شود. به این معنی که وقتی k ، به سمت بی نهایت می رود $a(k)$ یک محدوده برای سطح یک ابرصفحه با مرکز $\hat{a}$ و شعاع $r(\hat{a})$ می باشد. از آنجا که برای هر $\hat{a}$ در ناحیه جواب این شرط درست می باشد ، $a(k)$ محدود شده با فصل مشترک مرکز ابرصفحه ها برای همه بردارهای جواب ممکن محدود می شود. حال می توان نشان داد که فصل مشترک این ابرصفحه ها یک نقطه تک روی مرز ناحیه جواب است . فرض کنید که ابتدا دو نقطه $\hat{a}$ و $r(\hat{a})$ روی فصل مشترک وجود دارد . سپس $\|a' - \hat{a}\| = \|a'' - \hat{a}\|$ برای هر $\hat{a}$ در ناحیه جواب برقرار است. البته این زمانی است که ناحیه جواب ابر صفحه $(\hat{d} - ۱)$ بعدی از نقاط هم فاصله از $\mathbf{a}'$ و a'' را شامل می شود ، در حالیکه می دانیم ناحیه جواب $\hat{d}$ بعدی

است. (وضعیت فرمول شده : اگر $\hat{a}^t y_i > 0$ برای $i=1, \dots, n$ برقرار باشد ، سپس برای هر بردار $\hat{d}$ بعدی به نام v ، ما $(\hat{a}+\epsilon v)^t y > 0$ برای همه $i=1, \dots, n$ داریم اگر $++$ به اندازه کافی کوچک باشد)بنابراین $a(k)$ با یک نقطه a همگرا است. این نقطه داخل ناحیه جواب نیست برای هر دنباله ای که محدود شده باشد. و اگر خارج از هیچ کدام نباشد ، از آنجا که هر بهسازی باعث می شود که بردار وزن چندین بار به سمتη حرکت کند، آن فاصله از صفحه مرزی است ، بدین ترتیب مانع دور شدن بردارها از صفحه مرزی می شود .در نتیجه نقطه حدی باید روی مرز باشد.

۵-۷- محیط های جدانشدنی

روش های ملایم سازی و پرسپترون ، یک سری روش برای پیدا کردن بردارهای تفکیک پذیر نمونه ها می دهد که می توان با آن نمونه ها را به صورت خطی جدا کرد. این روش ها را ، فرآیندهای تصحیح–خطا می نامند، چون تغییراتی را روی بردارهای وزنی ایجاد می کنند زمانیکه که خطا اتفاق می افتد. موفقیت این مسائل تفکیک پذیری ، توانایی آنها با یک جستوجی بی وقفه برای یک راه حل بدون خطا می باشد. در اینجا ، این روش ها را بررسی می کنیم با توجه به اینکه نرخ خطا برای تابع تفکیک پذیر خطی بهینه کم است. البته حتی اگر یک بردار تفکیک پذیری برای نمونه های آموزشی پیدا شده باشد دسته بند برای داده های تست مستقل است. البته نشان می دهیم که هر مجموعه با نمونه های کمتر از $2\hat{d}$ شبیه به تفکیک کننده خطی است. بنابراین با چندین باراستفاده ازنمونه های طراحی شده ، دسته بند را ایجاد می کنیم . بدین ترتیب مطمئن می شویم که برای کارایی داده های آموزشی و آزمایشی تا چه اندازه شباهت وجود دارد. متاسفانه مجموعه های طراحی شده بزرگ مطمئنا به صورت خطی تفکیک پذیر نیستند. البته مهم است که فرایند تصحیح –خطا زمانیکه نمونه ها جدایی ناپذیر هستن انجام خواهد شد. از آنجا که هیچ بردار وزنی ، به درستی هر نمونه از مجموعه جدانشدنی را نمی تواند دسته بندی کند، واضح است که بهسازی فرایند تصحیح – خطا هرگز متوقف نمی شود.برای هر الگوریتم در یک دنباله محدود از بردارهای وزنی هر عضو ممکن است به عنوان یک جواب قابل استفاده باشد و یا نباشد. در موارد جدایی ناپذیر بودن ، در زمینه های خاص مطالعاتی صورت گرفته است. به طور مثال طول بردارهای آموزشی با استفاده از قوانین افزایشی ثابت محدود شده است. قوانین تجربی برای فرایند تصحیح ، براساس گرایش طول بردارهای وزنی با نوسانات نزدیک به مقادیر محدود شده است. از نظر تئوری ، اگر مولفه های نمونه ها دارای مقادیر عددی صحیح باشند، فرایند افزایشی ثابت در یک پردازش با وضعیت محدود شده کاربرد دارد. اپر روند تصحیح برای بعضی از نقاط دلخواه خاتمه یابد، بردارهای وزنی یک حات خوب ممکن است داشته باشد و یا نه . با توجه به میانگین بردارهای وزنی که با قوانین بهسازی تولید شده است ، خطای بدست آمده از جواب های نادرست با طولانی شدن زمان خاتمه کاهش می یابد.یک تعدادی از تغییرات ابتکاری مشابه با قوانین تصحیح – خطا پیشنهاد شده است . هدف از این تغییرات، دستیابی به کاراییی قابل قبول برای مسائل جدانشدنی است .یک پیشنهاد رایج ، استفاده از متغیر افزایشی $\eta(k)$ است بطوریکه $\eta(k)$ به صفر نزدیک شود زمانیکه که k نامحدود می شود. نرخ نزدیک شدن $\eta(k)$ به صفر ، کاملا مهم می باشد. اگر خیلی آرام باشد نتایج به نمونه های آموزشی حساس هستند و یک مجموعه جدانشدنی هنوز داریم . ولی اگر این نرخ نزدیک شدن $\eta(k)$ به صفر خیلی سریع انجام شود ، بردارهای وزنی خیلی زود با نتایج بهینه همگرا می شوند. یک راه انتخاب $\eta(k)$ ، استفاده از یک تابع با کارایی فوق العاده است. یک روش دیگر انتخاب و برنامه ریزی $\eta(k)$ ، انتخاب $\eta(k) = \eta(1)/k$ می باشد . زمانیکه تکنیک های تخمین ریاضی را آزمایش می کنیم ، باید بررسی شود که انتخاب دوم یک جواب تئوری برای مسائل هم ارز می باشد . قبل از اینکه این موضوع را

شروع کنیم یک روشی را که بدست آوردن بردارهای جداشدنی برای رسیدن به کارایی خوب در هر دو روش جداشدنی و جدا نشدنی ، منجر به از دست دادن چه توانایی های دیگری می شود.

۵-۸- فرآیند مینیمم مربعات خطا

۵-۸-۱- مینیمم مربعات خطا و روش Pseudoinverse

توابع معیار استاندارد که تا کنون بررسی کردیم برای نونه های دسته بندی نشده بود. ما باید یک توابع معیار استاندارد که شامل همه نمونه ها باشد را بررسی کنیم. همان طور که قبلا گفته شد یک بردار وزنی a ساخته شده از همه جملات مثبت $\mathbf{a}^t\mathbf{y}_i$ را باید جستجو کنیم ، سپس باید سعی کنیم که $\mathbf{a}^t\mathbf{y}_i = b_i$ را بسازیم که در آن b_i مقادیر ثابت مثبت دلخواه هستند. بنابراین مسئله پیدا کردن جواب را با یک مجموعه نابرابر خطی با رشته های بیشتر پیدا می کنیم اما هدف مسئله پیدا کردن جواب یک مجموعه از معادلات خطی می باشد. بحث همزمان سازی معادلات خطی با معرفی ماتریس مورد توجه ، ساده می شود. Y یک ماتریس n-by-$\hat{d}$ می باشد که سطر i ام بردار y_i^t است و b یک بردار ستونی $\mathbf{b} = (b_1, ..., b_n)^t$. می باشد. سپس در مسئله به دنبال پیدا کردن بردارهای وزنی a پذیرفته شده به صورت زیر هستیم :

$$\begin{bmatrix} Y_{10} & Y_{11} & \dots & Y_{1d} \\ Y_{20} & Y_{21} & \dots & Y_{2d} \\ \vdots & \vdots & \dots & \vdots \\ Y_{n0} & Y_{nd} & \dots & Y_{nd} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_d \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \qquad (41)$$

$$\mathbf{Ya} = \mathbf{b}$$

اگر Y منحصر به فرد نباشد، رابطه $\mathbf{a} = \mathbf{Y}^{-1}\mathbf{b}$ را داریم و یک جواب فرمول شده دست می آوریم. اگر Y مستطیلی باشد، سطرها بیشتر از ستون ها می شود.وقتی که معادلات ناشناخته بیشتری وجود دارد ، a بیش از حد توسعه پیدا می کند و معمولا جواب دقیقی در دسترس نیست. به هر حال بردار وزنی a را جستجو می کنیم که بعضی از توابع خطا را بین Ya و b مینیمم کند. اگر بردار خطای e را به صورت زیر تعریف کنیم :

$$\mathbf{e} = \mathbf{Y_a} - \mathbf{b} \qquad (42)$$

یک روش برای مینیمم کردن طول مربعات بردار خطا داریم. به عبارتی برابر با مینیمم کردن مجموع مربعات تابع معیار استاندارد خطا می باشد.

$$J_s \qquad (43)$$

مسئله مینیمم کردن مجموع مربعات خطا ، یک مسئله کلاسیک است که می توان با یک روش جستجوی گرادیان حل شود. یک نمونه جواب مورد تایید را میتوان به شکل گرادیان پیدا کرد:

$$\nabla J_s = \sum_{i=1}^{n} 2(a^t y_i - b_i)y_i = 2Y^t(Y_a - b) \qquad (44)$$

و آن را مساوی صفر قرار دهیم. که با شرایط زیر کار می کند :

$$\mathbf{Y^tYa} = \mathbf{Y^tb} \qquad (45)$$

و با این روش حل مسئله ++ را به حل کردن +++ تبدیل می کنیم .یک معادله معروف که فواید بسیاری دارد و ماتریس ++ با ابعاد +++ مربعی است و اغلب منحصر به فرد نیست. اگر منحصر به فرد نباشد ، می توان a را به صورت یکتا حل کرد به صورت :

$$\mathbf{a} = (\mathbf{Y}^t\mathbf{Y})^{-1}\mathbf{Y}^t\mathbf{b} = \mathbf{Y}^\dagger\mathbf{b} \qquad (٤٦)$$

که ابعاد ماتریس $\hat{d}$-by-n می باشد.

$$\mathbf{Y}^\dagger \equiv (\mathbf{Y}^t\mathbf{Y})^{-1}\mathbf{Y}^t \qquad (٤٧)$$

و pseudoinverse از Y نامیده می شود. توجه کنید که اگر Y مربعی و غیر منحصر به فرد باشد ، pseudoinverse با معکوس منتظم آن یکجور می باشد. همچنین توجه کنید که اگر $\mathbf{Y}^\dagger\mathbf{Y} = \mathbf{I},$ را داشته باشیم اما $\mathbf{Y}\mathbf{Y}^\dagger \neq \mathbf{I}$ وجود ندارد. به هر حال جواب مینیمم مربعات خطا موجود است . اگر $\mathbf{Y}^\dagger$ به صورت زیر تعریف شود :

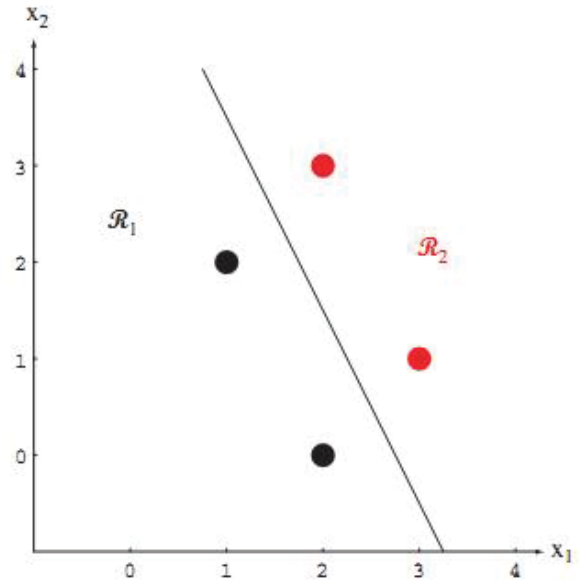
$$Y \qquad (٤٨)$$

این نشان می دهد که محدودیت وجود دارد و $\mathbf{a} = \mathbf{Y}^\dagger\mathbf{b}$ یک جواب مینیمم مربعات خطا برای Ya=b می باشد. اگر جواب مینیمم مربعات خطا به بردار مرزی b وابسته باشد، انتخاب های مختلفی برای b وجود دارد تا ویژگی های مختلف جواب بدست آید. اگر b یک مقدار ثابت دلخواه باشد، هیچ دلیلی وجود ندارد که جواب مینیمم مربعات خطا برای بردارهای جداشدنی در زمینه های تفکیک پذیری خطی باز دهی نداشته باشد. به هر حال ، قابل قبول است که امیدوار باشیم جواب مینیمم مربعات تابع معیار استاندارد خطا، یک تابع تفکیک پذیری مناسب در هر دو حالت جداشدنی و جدا ناشدنی باشد. ما باید این دو ویژگی جواب را بررسی کنیم تا از این قابل قبول بودن مطمئن شویم.

مثال ۱ : ساخت دسته بند خطی با ماتریس pseudoinverse :

فرض کنید که نقاط دو بعدی برای دو کلاس $(۱, ۲)^t$ و $(۲, ۰)^t$ داریم . و با رنگ قرمز و سیاه نشان می دهیم. (مطابق شکل). ماتریس Y به صورت زیر است :

$$Y = \begin{pmatrix} ۱ & ۱ & ۲ \\ ۱ & ۲ & ۰ \\ -۱ & -۳ & -۱ \\ -۱ & -۲ & -۳ \end{pmatrix}$$



بعد از محاسبه تعداد کمی از نمونه ها ، pseudoinverse را به صورت زیر بدست می آوریم :

چهار نقطه آموزشی و مرز تصمیم گیری $a^t \begin{pmatrix} ۱ \\ x_۱ \\ x_۲ \end{pmatrix} = ۰$ را داریم که در آن a میانگین تکنیک pseudoinverse می باشد.
ما به طور دلخواه همه مرزها را یکسان در نظر می گیریم. به طور مثال $\mathbf{b} = (۱, ۱, ۱, ۱)^t$ جواب
$\mathbf{a} = \mathbf{Y}^{\dagger}\mathbf{b} = (۱۱/۳, -٤/۳, -۲/۳)t,$ می شود و مرز تصمیم گیری در شکل نشان داده شده است. انتخاب های دیگر b مرزهای تصمیم گیری مختلفی را ایجاد می کند.

۵-۸-۲- تفکیک خطی فیشر

در این بخش نشان می دهیم که با انتخاب مناسب بردار b ، تابع تفکیک پذیری مینیمم مربعات خطا $\mathbf{a}^t\mathbf{y}$ به طور مستقیم با تفکیک خطی فیشر ارتباط دارد. با استفاده از خطی بودن توابع تفکیک پذیری خطی تعمیم یافته را استفاده می کنیم. ابتدا فرض می کنیم که یک مجموعه از n نمونه d بعدی $\mathbf{x}_۱, ..., \mathbf{x}_n, n_۱$ را داریم که زیر مجموعه $D_۱$ با $\omega_۱$ برچسب می خورد و $n_۲$ در زیر مجموعه $D_۲$ با $\omega_۲$ برچسب می خورد. بنابراین ما فرض می کنیم که نمونه های y_i با افزودن مولفه آستانه x_i برای ساختن بردار الگوی افزودنی از $x_۰ = ۱$ تشکیل شده اند. بنابراین اگر نمونه ها با $\omega_۲$ برچسب بخورند، سپس بردار الگوی کاملی با یک ضرب می شود تا نرمالایز انجام شود. همان طور که در بخش ۵-٤-۱ دیدیم می توان فرض کرد که ابتدا $n_۱$ نمونه با $\omega_۱$ برچسب می خورند و سپس $n_۲$ نمونه با $\omega_۲$ برچسب می خورند. سپس ماتریس Y به صورت زیر پارتیشن بندی می شود.

$$Y=\begin{bmatrix} ۱_۱ & x_۱ \\ -۱_۲ & x_۲ \end{bmatrix}$$

که در آن ۱_i یک بردار ستونی از n_i می باشد و $\mathbf{X}_i$ یک ماتریس n_i-by-d است که درآن سطرها نمونه های برچسب خورده ω_i هستند. متقابلا a و b را پارتیشن بندی می کنیم :

$$a=\begin{pmatrix} \omega_. \\ w \end{pmatrix}$$

$$b=\begin{bmatrix} \frac{n}{n_۱} l_۱ \\ \frac{n}{n_۲} l_۲ \end{bmatrix}$$

ما باید نشان بدهیم که این انتخاب ویژه b با حل MSE برای تفکیک خطی فیشر رابطه دارد. معادله (٤٧) را بابه صورت ماتریس تقسیم شده از ترم ها می نویسیم.

(٤٩)

با تعریف میانگین نمونه ها m_iو ماتریس پراکندگی نمونه های مورد نظر S_w معادله زیر را داریم :

(٥٠)

n

و

(٥١)

S

اگر ماتریس معادله (٤٩) را ضرب کنیم به صورت زیر می شود:

این به صورت یک جفت معادله دیده می شود که می توانیم $w_.$ را به صورت w حل کنیم :

(٥٢)

m میانگین همه نمونه ها است . با جانشین کردن در بخش دوم معادله و کمی دستکاری ریاضی ، رابطه زیر بدست می آید:

(٥٣)

از آنجا که بردار $(m_۱ - m_۲)(m_۱ - m_۲)^t w$ در جهت $m_۱ - m_۲$ برای هر مقدار w است می توان نوشت :

که α یک مقدار اسکالر است . معادله (٥٣) به صورت زیر می شود:

$$W= \alpha n S_w^{-۱}(m_۱ - m_۲) \qquad (٥٤)$$

به استثنای یک مقدار عددی کم اهمیت ، این معادله با حل تفکیک خطی فیشر یک جور است. علاوه بر این ، وزن آستانه $w_.$ را داریم و که برای آن قوانین تصمیم گیری زیر را دنبال می کنیم :

اگر رابطه $w^t(x-m) > 0$ برقرار باشد کلاس w_1 را داریم ، در غیر اینصورت کلاس w_2را داریم.

۵-۸-۳-تخمین هندسی برای یک تفکیک بهینه

یک ویژگی دیگر از راه حل MSE که توصیه شده است زمانی است که $b = l_n$ می باشد که تخمین مینیمم حداقل مربعات خطا برای تابع تفکیک بیزین می باشد.

g (۵۵)

محدودیت در تعداد نمونه روش های بی شماری است. برای اثبات آن ، ابتدا فرض می کنیم نمونه ها ،با یک توزیع یکسان مطابق با کمترین احتمال مستقل هستند.

p (۵۶)

برای ترم هایی با بردار افزایشی y ، راه حل MSE با گسترش رابطه $g(x) = a^t y$ بدست می آید که در y=y(x) می باشد. اگر تخمین میانگین مربعات خطا را به صورت زیر تعریف کنیم :

ε (۵۷)

سپس می توان نشان داد که ε^2 با راه حل $a = Y^\dagger l_n$ مینیمم می شود. در جهت اثبات ساده سازی ، ما توزیع بین نمونه های کلاس های w_1و w_2را حفظ می کنیم. در این ترم ها داده ها نرمال نشده اند وتابع تصمیم گیری J_S به صورت زیر می شود:

$$J_S(a) = \sum_{y\in y_1}(a^t y - 1)^2 + \sum_{y\in y_2}(a^t y - 1)^2 = \quad (۵۸)$$
$$n[\frac{n_1}{n}\frac{1}{n_1}\sum_{y\in y_1}(a^t y - 1)^2 + \frac{n_2}{n}\frac{1}{n_2}\sum_{y\in y_2}(a^t y + 1)^2 \quad .$$

بنابراین، با توجه به مقادیر بزرگ ، تعداد n مشابه بیشمار $(\frac{1}{n})J_S(a)$ بدست می آید.

j (۵۹)

با یک احتمال داریم :

ε

و

ε

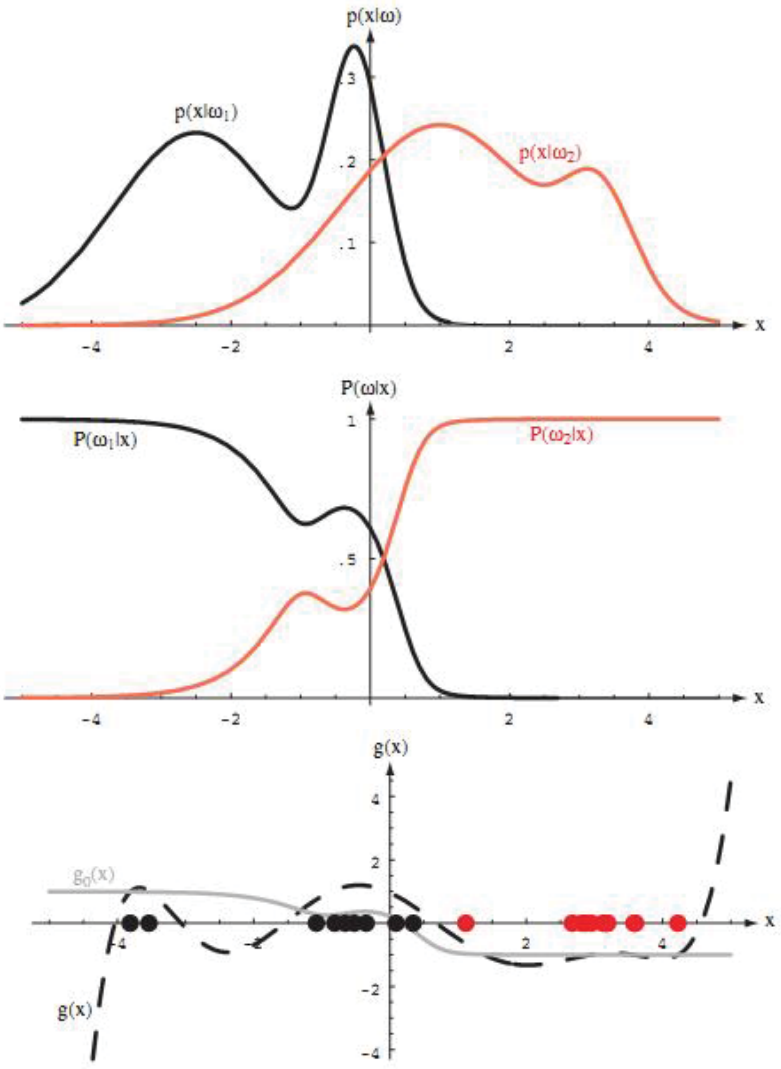
حالا با پذیرفتم معادله (۵۵) داریم:

g

بنابراین می بینیم که :

(۶۰)

جمله دوم معادله فوق ، وابسته به بردار وزنی a است. از آنجا که مینیمم J_S ، همچنین مینیمم ε^2 می باشد میانگین مربعات خطا بین $a^t y$ و $g(x)$ است . در شکل ۵-۱۶ ، ویژگی های هم ارزی برای تعداد زیادی از شبکه های چند لایه ای نشان داده شده است.



شکل ۵-۶: شکل اول چگالی ۲ کلاسی شرطی را نشان می دهد ، شکل دوم احتمال ثانویه است. فرض می کنیم معادل احتمال اولیه است. برای مینیمم کردن خطای MSE ، میانگین مربعات خطای بین $a^t y$ و تابع تفکیک $g(x)$ روی توزیع داده ها اندازه گیری می شود(چند جمله ای مرتبه ۷ ام). که در شکل سوم نشان داده شده است. توجه کنید که نتیجه$g(x)$ با بهترین تخمین $g_0(x)$ در نواحی که داده اشتباه وجود دارد.

این نتایج روند MSE را نشان می دهد. با تخمین $g_0(x)$ ، تابع تفکیک $a^t y$ اطلاعات درستی را برای احتمال ثانویه $p(w_1|x) = (1 + g_0)/2$ و $p(w_2|x) = (1 - g_0)/2$ می دهد. تخمین کیفیت به تابع $y_i(x)$ و تعداد جملات توسعه یافته $a^t y$ بستگی دارد.متاسفانه حل استاندارد میانگین مربعات خطا به نقاطی اشاره می کند که $p(x)$ بزرگتر

هستند و به سطح تصمیم گیری $g.(x) = ۰$ نزدیک تر هستند.بنابراین تابع تفکیک کننده بیزین برای مینیمم کردن خطای احتمالی ستفاده می کنیم. علی رغم این ویژگی ، راه حل MSE ویژگی ها را جذب می کند. ما باید تخمین میانگین مربعات $g.(x)$ را مجددا زمانیکه روش های تخمین تصادفی و شبکه عصبی چند لایه را بررسی می کنیم ، بدست آوریم.

۵-۸-٤- روش Widrow-Hoff

ابتدا مشاهده می کنیم که با تابع گرادیان نزولی $J_S(a) = ||Ya - b||^۲$ می تواند مینیمم شود. در این روش دو فایده با محاسبات pseudoinverse داریم :

۱- از مسائلی که مشکلات را زیاد می کند ، زمانیکه Y^tY منحصر به فرد است ، اجتناب می کنیم

۲- از مسائلی که نیاز به کار کردن با ماتریس بزرگتر دارد ، اجتناب می کنیم.

علاوه بر این ، محاسبات موجود با یک طرح بازخورد موثر و کارا می باشد که به طور اتوماتیک تعدادی از مسائل محاسباتی را با توجه به کوتاه سازی و یا roundoff کپی می کند. از آنجا که $\nabla J_s = ۲Y^t(Ya - b)$ داریم ، قانون بهینه سازی آشکار زیر را داریم :

$$a(۱) \quad \text{arbitrary}$$
$$a(k+۱) = a(k) + \eta(k)Y^t(Ya - b)$$

در تمرین ۲٤ ، خواسته شده است که نشان دهیم که اگر $\eta(k) = \frac{\eta(۱)}{k}$ برقرار باشد و $\eta(۱)$ هر مقدار ثابت مثبتی می باشد ، سپس این قانون یک دنباله ازبردارهای وزنی را تولید می کند که با بردار محدودیت a مورد نظر همگرا می شود.

Y

بنابراین ، الگوریتم نزولی معمولا یک راه حل بدون دقت فراهم می کندو Y^tY منحصر به فرد نیست. در حالیکه ماتریس $\hat{d} * \hat{d}$ برای Y^tY معمولا کمتر از ماتریس $\hat{d} * n$ برای $Y^\dagger$ می باشد. ذخیره سازی مورد نیاز می تواند با بررسی دنباله نمونه ها و با استفاده از قوانین , LMS و یا Widrow-Hoff بیشتر کاهش یابد.

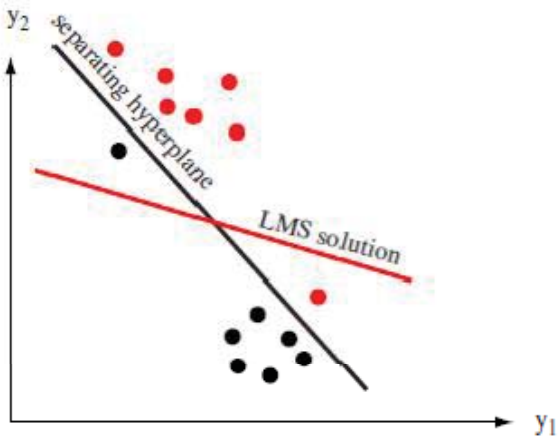
$$a(۱) \quad \text{arbitrary} \qquad (٦۱)$$
$$a(k+۱) = a(k) + \eta(k)(b_k - a(k)^t y^k)y^k$$

الگوریتم LMS به شکل زیر می باشد:

[۱] Begin initialize,**b , criterion** θ **,**$\eta(0), k = 0$
[۲] Do $k = k+۱$
[۳] $a = a + \eta(k)(b_k - a^t y^k)y^k$
[٤] Until $\eta(k)(b_k - a^t y^k)y^k < \theta$
[۵] Return **a**
[٦] End

با بررسی اولیه الگوریتم نزولی به نظر می رسد که ماهیت یکسان با قانون ملایم سازی دارد. اختلاف اول با این قانون در قانون بهسازی خطا می باشد، بطوریکه $a(k)^t y^k$ با b_k مساوی نباشد و در نتیجه بهسازی هرگز متوقف نمی شود. بنابراین $\eta(k)$ باید با k برای بدست آوردن همگرایی کاهش یابد. انتخاب $\eta(k) = \frac{\eta(۱)}{k}$ ساده می شود. آنالیز دقیق قانون Widrow-

Hoff در زمینه تفکیک پذیری هم پیچیده تر می شود و کاملا نشان می دهد که دنباله بردار وزنی همگرایی به بهترین راه حل را حفظ می کند. به جای تعقیب بیشتراین موضوع ، باید از قانون تشابه بیشتر برای روش نزولی احتمالی استفاده کرد. در هر صورت برای این راه حل به بردارهای جداشده نیاز نداریم حتی اگر موجود نباشد همان طور که در شکل ۵-۱۷ نشان داده شده است.



شکل ۵-۱۷ : الگوریتم LMS با صفحه جدا کننده همگرا نشده است. از آنجا که راه حل LMS مجموعه مربعات نمونه های آموزشی از صفحه است در این مثال ، صفحه ای که در جهت عقربه های ساعت می چرخد با یک صفحه جدا کننده مقایسه می شود.

۵-۸-۵- روش های تخمین احتمالی

همه روش های نزولی تکراری برای همه جملات تکراری توصیف شده بررسی می شود. یک مجموعه ویژه از نمونه ها را داریک که دنباله ویژه ای از بردارهای وزنی را تولید می کند. در این بخش کمتر درباره روش MSE با نمونه های تصادفی صحبت می شود و به تئوری تخمین تصادفی می پردازیم و در اینجا هدف اصلی بدون اثبات نشان داده خواهد شد. فرض کنید نمونه ها با توجه به وضعیت انتخاب احتمالی $p(w_i)$ و سپس یک x مطابق با قانون احتمالی $p(x|w_i)$ انتخاب می کنیم. هر x با یک **مقدار**θ برچسب زده می شود.اگر θ=+۱ باشد x با w_1 برچسب زده می شود واگر θ=−۱ باشد x با w_2برچسب زده می شود.سپس داده ها از یک دنباله محدود از جفت های.... $(x_3,\theta_3),\dots(x_2,\theta_2),(x,\theta_1)$ شامل می شوند. متغیر برچسب شده θ مقدار باینری است . به نظر می توان یک نسخه نویزی از توابع تفکیک بیزین را $g_.(\mathrm{x})$ استدلال کرد. این به صورت زیر دیده می شود :

$$p$$

$$p$$

و میانگین شرطی θ به صورت زیر است :

$$\varepsilon_{\theta|x}[\theta] = \sum_{\theta} \theta P(\theta|x) = p(w_1|x) - p(w_2|x) = g_.(\mathrm{x}) \qquad (۶۲)$$

فرض می کنیم که تخمین $g_.(\mathrm{x})$ با توسعه سری محدود به صورت زیر باشد :

$$\mathrm{g(x)} = a^t y = \sum_{i=1}^{\hat{a}} a_i\, y_i(x)$$

هر دو تابع زیرساختی $y_i(x)$ و تعداد جملات $\hat{d}$ هر دو شناخته شده اند. سپس ما می توانیم بردار وزنی $\hat{a}$را جستجو کنیم که تخمین میانگین مربعات خطا را مینیمم کنیم .

ε (۶۳)

با مینیمم کردن ε^2 به نظر خواهد رسید که به دانشی از تفکیک بیزین $g_i(x)$ نیاز داریم .به هر حال یک وضعیت مشابه با بخش ۳-۸-۵ حدس زده می شود که نشان می دهد که بردار وزنی $\hat{a}$ که ε^2 را مینیمم می کند ، تابع مقیاس زیر را نیز مینیمم می کند.

J_1 (۶۴)

این باید همچنین احتمال این حقیقت که θ یک نسخه نویزی از باشد و از آنجا که گرادیان به صورت زیر است :

∇ (۶۵)

ما می توانیم که راه حل closed-form بدست بیاوریم :

$\hat{a}$ (۶۶)

بنابراین ، یک راه استفاده از نمونه ها ، تخمین $\varepsilon[yy^t]\varepsilon[\theta y]$ و می باشد و از معادله (۶۶) برای تفکیک خطی بهینه MSE استفاده می کنیم . با یک تکرار ، بوسیله روش نزولی گرادیان $J_m(a)$ را مینیمم می کنیم . فرض می کنیم در جایی که گرادیان درست می باشد ما یک نسخه نویزی از $2(a^t y_k - \theta_k)y_k$ را جایگزین می کنیم که به صورت قانون زیر می باشد :

$$a(k+1) = a(k) + \eta(k)(\theta_k - a^t(k)y_k y_k \quad (۶۷)$$

که در اینجا قانون Widrow-Hoff است که $\varepsilon[yy^t]$غیر منحصر به فرد است **و همگرایی $\eta(k)$برآورده می شود :**

$$\lim_{m\to\infty}\sum_{k=1}^{m}\eta(k) = +\infty \quad (۶۸)$$

و

$$\lim_{m\to\infty}\sum_{k=1}^{m}\eta^2(k) < +\infty \quad (۶۹)$$

سپس a(k) با $\hat{a}$ **در مربعات خطا همگرا می شود :**

$$\lim_{m\to\infty}\varepsilon[||a(k) - \hat{a}||^2] = 0 \quad (۷۰)$$

با دلایلی می توان $\eta(k)$را ساده کرد. شرط اول بردارهای وزنی برای همگرایی را حفظ می کنیم به طوریکه یک خطای سیستمی خاص برای همیشه اشتباه باقی خواهد ماند. شرط دوم مطمئن هستیم که نوسانات تصادفی سرانجام ازبین می روند . هر دو این شرایط با انتخاب معمولی $\eta(k) = \frac{\eta(1)}{k}$ فراهم می شود.متاسفانه با کاهش $\eta(k)$ ، اغلب همگرایی کمتر می شود. البته ، این فقط بهترین الگوریتم نزولی برای مینیمم کردن J_m نیست ،*البته با یک ماتریس از مشتق ناتمام ثانویه برای J_m می توانیم نشان می دهیم که :*

L

با قانون نیوتن برای مینیمم کردن J_m (معادله ۱۵) داریم :

$$a(k+1) = a(k) + \varepsilon[yy^t]^{-1}\varepsilon[(\theta - a^t y)y]$$

و با مقایسه احتمال این قانون داریم :

$$a(k+1) = a(k) + R_{k+1}\,(\theta_k - a^t(k)y_k)y_k] \qquad (۷۱)$$

و با

$$R_{k+1}^{-1} = R_k^{-1} + y_k y_k^t \qquad (۷۲)$$

و معادل با آن داریم :

$$R_{k+1} = R_k - \frac{R_k y_k (R_k y_k)^t}{1 + y_k^t R_k y_k} \qquad (۷۳)$$

این قانون یک دنباله از بردارهای وزنی را تولید می کند که با راه حل بهینه در میانگین مربعات همگرا می شود. این همگرایی سریع تر است اما برای محاسبات هر مرحله بیشتر مورد نیاز است.

این روش گرادیان به عنوان روش هایی برای مینیمم کردن تابع مقیاس و یا پیدا کردن مقدار صفر برای گرادیان با وجود نویز دیده شده است. از نظر ریاضی توابعی مانند J_m و ∇J_m که به شکل $E[f(\mathbf{a}, \mathbf{x})]$ هستند را توابع رگرسیون می نامیم و الگوریتم های تکراری را تخمین احتمالی می نامیم.

دو روش مشهور داریم که عبارتند از روش Kiefer-Wolfowitz برای مینیمم کردن تابع رگرسیون و روش Robbins-Monro برای پیدا کردن ریشه تابع رگرسیون. اغلب راه ساده تر برای اثبات همگرایی نزولی بودن ویژه و یا روش تخمین بدست می آید که شرایط همگرایی را برای توابع معمولی تر نشان می دهد. متاسفانه توضیح این روش با تعمیم کامل بعید بنظر می رسد که ما راهدایت کند و ما باید این انحراف را با ارجاع به خواندن مقالات بیشتر بدست آوریم.

۵-۹- روش های Ho-Kashyap

۵-۹-۱-روش نزولی

روش هایی که تا کنون بررسی کردیم در چندین زمینه با هم اختلاف دارند. روش پرسپترون و ملایم سازی بردارهای جداگانه ای را پیدا خواهد کرد اگر نمونه ها به صورت خطی جدایی پذیر باشند . اما مسائل غیرجدایی پذیر همگرا نمی شوند. روش MSE با بردارهای وزنی برای آن دسته از نمونه هایی که بکار برده شده است به صورت خطی جدایی پذیرند و یا می توانند نباشند اما هیچ ضمانتی وجود ندارد که این بردارها برای بردارهای جدایی پذیری در زمینه تفکیک پذیری باشند (شکل ۵-۱۷). اگر بردار مرز b به صورت اختیاری انتخاب شود می توان گفت که در روش MSE ، $||Ya - b||^2$ مینیمم می شود. حالا اگر نمونه های آموزشی به صورت خطی جدایی پذیر باشد رابطه بین $\hat{a}$ و $\hat{b}$ به صورت زیر است :

$$\mathbf{Y}\hat{a} = \hat{b} > 0,$$

که در آن $\hat{b} > 0$ می باشد و ما می دانیم که هر مولفه $\hat{b}$ ، مثبت می باشد.واضح است که $\boldsymbol{b} = \hat{b}$ را داریم و روش MSE را بکار می بریم. یک بردار تفکیک کننده بدست می آوریم.البته مقدار $\hat{b}$ را از قبل نمی دانیم. به هر حال می بینیم که چگونه روش های MSE استفاده از بردارهای تفکیک پذیری a و بردار مرزی b را تغییر می کند. هدف اصلی از این مشاهده بدست می آید که اگر نمونه ها تفکیک پذیر باشند و هم a و هم b در تابع مقیاس استاندارد باشد :

$$J_s \qquad (۷۴)$$

با یک تغییر مقدار مینیمم J_S صفر می شود و a از مینیمم کردن بردارهای تفکیک پذیری بدست می آیند.برای مینیمم کردن J_Sاز یک روش شیب نزولی تغییر یافته استفاده می کنیم. گرادیان J_S با توجه به a به صورت زیر است :

∇ (۷۵)

و گرادیان J_S با با توجه به b را به صورت زیر داریم :

∇ (۷۶)

و برای هر مقدار از b معمولا می توانیم رابطه زیر را داشته باشیم :

a (۷۷)

بدین طریق $\nabla_a J_S = ۰$ بدست می آوریم و J_Sبا توجه به a در هر مرحله مینیمم می کنیم. ما نمی توانیم b را تغییر دهیم ، به هر حال باید الزام $b > ۰$ را رعایت کنیم ، b همگرا به صفر است وقتی که $b > ۰$ باشد و کاهش هر مولفه را رد می کنیم . می توان این را انجام داد و گرادیان منفی را دنبال کرد اگر ابتدا همه مولفه های مثبت $\nabla_b J_S$ با صفر تنظیم شود. بنابراین اگر $|V|$ یک بردار باشدکه مولفه های آن از نظر کمیت مزابق با مولفه های V باشد ، می توان یک قانون بهینه سازی برای مرزها به شکل زیر داشته باشیم :

b (۷۸)

با استفاده از معادله(۷۶) و (۷۷) ، می توانیم قانون Ho-Kashyap را برای مینیمم کردن $J_S(a,b)$بدست آورد.

(۷۹) زمانیکه دلخواه نیست $b(۱)>۰$

$b(k+۱) = a(k) + ۲\,\eta(k)e^{+}(k)$

که در آن e(k) یک بردار خطا است

$e(k) = Ya(k) - b(k)$ (۸۰)

$e^{+}(k)$ یک بخش مثبت از بردار خطا است

e (۸۱)

و

$a(k) = Y^{\dagger}b(k)$ (۸۲)

بنابراین اگر b_{min} یک مقیاس همگرایی کوچک باشد و Abs[e] یک بخش کوچکی از e باشد ، الگوریتم زیر را داریم :

Algorithm ۱۱ (Ho-Kashyap)

[۱] Begin initialize $\boldsymbol{a, b, \eta(0) < ۱}$, **criterion** , $\mathbf{b_{min}}$, $\mathbf{k_{max}}$
[۲] **Do k = k+۱**
[۳] **e= Ya-b**
[٤] $\boldsymbol{e^{+} = \frac{1}{2}(e + Abs[e])}$
[٥] $\boldsymbol{a = Y^{\dagger}b}$
[٦] If **Abs[e]** $\leq\leq$ $\boldsymbol{b_{min}}$ then return **a,b and** exit
[۷] **Until** $\boldsymbol{k = k_{max}}$
[۸] Print No SOLLUTION FOUND
[۹] **End**

از آنجا که بردار وزنی a(k) به طور کامل با بردار مرزها b(k) مشخص می شود ، این یک الگوریتم اصلی برای تولید یک دنباله از بردارهای مرزها است. بردار اولیه b(۱) برای شروع کردن مثبت است و اگر $\eta > ۰$ باشد همه بردارهای بعدی b(k) مثبت هستند . ما فقط نگران هستیم اگر هیچ کدام از مولفه های e(k) مثبت نباشد ، بطوریکه b(k) با تغییرات متوقف می شودو برای بدست آوردن راه حل کافی نباشد. به هر حال باید ذیذ که برای هر یک از e(k)=۰ یک راه حل داریم و یا e(k)<=۰ دلیل داریم که نمونه ها به صورت خطی جدایی ناپذیر باشند.

۵-۹-۲- اثبات همگرایی

ما باید نشان دهیم که اگر نمونه ها به صورت خطی جدایی پذیرند و اگر $۰ < \eta < ۱$ برقرار باشد ، سپس الگوریتم Ho-Kashyap یک بردار جواب برای تعدادی از مراحل تولید می کند. برای خاتمه داده الگوریتم ما باید یک وضعیت شرط پایانی اضافه کنیم که هر بهسازی هر بار ، با یک بردار جواب متوقف شود و یا تعداد مقیاس بزرگی از تکرارها اتفاق بیافتد. به هر حال از نظر ریاضی مناسب تر است که بهسازی ادامه پیدا کند و نشان دهیم که هر یک از بردارهای خطای e(k) برای تعداد محدودی از k همگرا می شود و یا همگرا به صفر می شود زمانیکه k نامحدود است .

کاملا واضح است که هر یک از e(k)=۰ برای بعضی از k را میتوان k. نامید و یا هیچ بردار صفر در دنباله e(۱),e(۲),… وجود ندارد. در حالت اول ، برای هر بردار صفر که بدست آمده است هیچ تغییر اضافه ای برای a(k) ، b(k) و یا e(k) اتفاق نمی افتد و Ya(k) = b(k)>۰ برای همه $k \geq k$. می باشد. بنابراین ، اگر یک بردار با خطای صفر بدست آوریم ، الگوریتم به طور اتوماتیک با یک بردار جواب متوقف می شود. فرض کنید که e(k) هرگز برای هر k محدودی صفر نباشد، با این وجود e(k) باید همگرا به صفر باشد. سئوالی که مطرح می شود این است که آیا احتمال دارد که e(k) با هیچ مولفه مثبتی بدست آید. متاسفانه ، از آنجا که Ya(k) <=b(k) و $e^+(k)$ باید صفر باشد ، ما هیچ تغییری بیشتری در a(k) ، b(k) و e(k) نداریم. خوشبختانه ، این هرگز اتفاق نمی افتد اگر نمونه ها به صورت خطی جدایی پذیر باشند. این اثبات ساده است و بر این اساس است که اگر $Y^tYa(k) = Y^tb$ باشد سپس $Y^te(k) = ۰$ وجود دارد. اما اگر نمونه ها به صورت خطی جدایی پذیرند ، سپس یک $\hat{a}\,,\ a\hat{b} > ۰$ وجود دارد به طوریکه

Y

بنابراین

e

و از انجا که همه مولفه های $\hat{b}$ مثبت هستند ، برای هریک از e(k) = ۰ و یا حداقل یکی از مولفه های e(k) باید مثبت باشد. از آنجا که مانع هر e(k) = ۰ می شویم ، $e^+(k)$ برای هر k محدود صفر می شود. برای اثبات بردار خطا که معمولا همگرا به صفر می شود، به مواردی از جمله اینکه ماتریس $YY^\dagger$ متقارن است ، نیمه معین مثبت است و داریم :

((۸۳)

اگرچه این نتایج به طور معمول درست است ، برای ساده سازی اثبات می کنیم که فقط برای جایی که $YY^\dagger$ غیر منحصر به فرد است ، می باشد. در این وضعیت $YY^\dagger = Y(Y^\dagger Y)^{-۱}Y^t$ تقارن کاملا واضح است . از آنجا که $Y^\dagger Y$ مثبت معین است به طوری که $(Y^\dagger Y)^{-۱}$ را داریم ، بنابراین $bY(Y^\dagger Y)^{-۱}Y^tb \geq ۰$ برای هر b داریم : و $YY^\dagger$ حداقل یک نیمه معین مثبت است. سرانجام معادله (۸۳) به صورت زیر می شود :

(

از آنجا e(k) باید به صفر همگرا شود ؛ a(k) را با معادلات (۸۰) و (۸۲) تخمین می زنیم و معادله زیر بدست می آید:

e

بنابراین ؛ با استفاده از نرخ یادگیری و معادله (۷۹) ما رابطه بازگشتی زیر را بدست می آوریم :

e (۸٤)

بنابراین

۱
٤

برای جمله دوم و سوم ساده سازی انجام می شود. از آنجا که $e^t(k)Y = ۰$ برقرار است جمله دوم به صورت زیر می شود :

η

مولفه های غیر صفر $e^+(k)$ ، مولفه های مثبت e(k) می شوند. از آنجا که $YY^\dagger$ متقارن است و با $(YY^\dagger)^t(YY^\dagger)$ برابر است ، جمله سوم به صورت زیر ساده می شود :

||
=

بنابراین داریم:

۱
٤ (۸۵)

با فرض اینکه $e^+(k)$،غیر صفراست وازآنجا که $YY^\dagger$ نیمه معین مثبت است ، داریم: $||e(k)||^۲ > ||e(k+۱)||^۲$ اگر $۰ < \eta < ۱$. بنابراین دنباله $||e(k)||^۲, ||e(۲)||^۲, \dots$ یک کاهش دهنده یکنواخت است و باید با مقادیر محدودی از $||e||^۲$ همگرا شود. اما برای همگرایی که اتفاق می افتد $e^+(k)$باید با صفر همگرا شود ، بطوریکه همه مولفه های مثبت e(k) باید به صفر همگرا شود. از آنجا که برای همه k، $e^+(k)\hat{b} = ۰$ برقرار است ، همه مولفه های e(k) باید همگرا به صفر شود. بنابراین اگر $۰ < \eta < ۱$ باشد و اگر نمونه ها به صورت خطی جدایی پذیر باشند ، a(k) به یک بردار جواب همگرا خواهد شد با وجود اینکه kبه سمت بی نهایت می رود. اگر علامت مولفه های Ya(k) در هر مرحله را تست کنیم و الگوریتم را زمانیکه همه آنها مثبت هستند خاتمه دهیم ، به این واقعیت می رسیم که یک بردار جداگانه در یک تعداد محدودی از مراحل بدست می آوریم که با توجه به Ya(k)=b(k)+e(k) و اینکه مولفه های b(k) هرگز صفر نمی شوند، بدست می آید.بنابراین اگر b_{min}، کوچکترین مولفه از b(۱) باشد و اگر e(k) همگرا به صفر باشد، سپس e(k) باید بعد از طی تعداد محدودی از مراحل یک صفحه کروی $||e(k)|| = b_{min}$ باشدکه در آن Ya(k)>۰ است. اگر چه شرط خاتمه را برای ساده سازی این اثبات نادیده بگیریم ، مانند شرط پایانی که معمولا در عمل استفاده می شود.

۵-۹-۳- رفتارهای تفکیک ناپذیر

اثبات همگرایی با فرض جدایی پذیر بودن مورد بررسی قرار گرفت،را بکار می بریم ، که در دو مرتبه باید انجام شود. اول $e^t(k)\hat{b} = ۰$ برای نشان دادن e(k)=۰ برای هر k محدود استفاده شده بود و یا $e^+(k)$هرگز صفر نمی شود و برای همیشه بهسازی تکرار شود. دوم ، یک الزام یکسان برای نشان دادن همگرایی به صفر $e^+(k)$ است که e(k) همچنین باید همگرا به صفرباشد.اگر نمونه ها به صورت خطی جدایی پذیر نباشند، آن مناسب نیست به طوریکه دنبال می کنیم $e^+(k)$ صفر است ، سپس e(k) باید صفر باشد. البته در مسائل جدایی ناپذیر یک بردار خطای غیر صفر بدون هیچ مولفه غیر مثبتی بدست می اوریم . اگر این اتفاق بیافتد، الگوریتم به طور اتوماتیک خاتمه می یابد و ما اثبات کرده ایم که نمونه ها جدایی ناپذیرند.

اگر الگوها جدایی ناپذیر باشند اما $e^+(k)$ هرگز صفر نباشد چه اتفاقی می افتد؟ در این وضعیت معادله زیر را داریم :

$$e \qquad (۸۶)$$

و

$$\frac{۱}{٤} \qquad (۸۷)$$

بنابراین دنباله $||e(k)||^۲, ||e(۲)||^۲, \dots$ باید همگرا باشد ، با آنکه مقادیر محدود $||e||^۲$ نمی تواند صفر باشد. از آنجا که همگرایی $e^+(k) = ۰$ ، هر مقدار محدود k مورد نیاز است، و یا $e^+(k)$ همگرا به صفر باشد زمانیکه ||e(k)|| کرانی از صفر است . بنابراین الگوریتم Kashyap یک بردار تفکیک کننده در وضعیت های تفکیک پذیری و با نشانه تفکیک ناپذیری در زمینه تفکیک ناپذیری فراهم می کند.به هر حال هیچ مرزی روی تعداد مراحل مورد نیاز برای تفکیک ناپذیری مشخص شده وجود ندارد.

۵-۹-٤- چند روش وابسته

اگر $Y^\dagger = (YY^\dagger)^{-۱}Y^\dagger$ را داشته باشیم و از این حقیقت $Y^\dagger e(k) = ۰$ استفاده کنیم می توانیم قانون Ho-Kashyap را به صورت زیر تغییر دهیم :

$$b(۱)>۰ \quad \text{درغیر این صورت دلخواه است} \qquad (۸۸)$$
$$a(۱) = Y^\dagger b(۱)$$

و از انجا که داریم :

$$e(k)=Ya(k)-b(k) \qquad (۸۹)$$

الگوریتم با نرخ یادگیری ثابت را داریم :

Algorithm ۱۲ (Ho-Kashyap)

[۱] Begin initialize $\boldsymbol{a}, \boldsymbol{b}, \eta < ۱$, **criterion**, $\mathbf{b_{min}}$, $\mathbf{k_{max}}$
[۲] **Do** $\mathbf{k = k+۱}$
[۳] $\mathbf{e = Ya-b}$
[٤] $e^+ = \frac{۱}{۲}(e + Abs[e])$
[۵] $b = b + ۲\eta(k)(e + Abs[e]$
[٦] $a = Y^\dagger b$

[۷] If **Abs[e]** $\leq\leq$ b_{min} then return **a,b and** exit
[۱۰] Until $k = \mathbf{k_{max}}$
[۱۱] Print No SOLLUTION FOUND

این الگوریتم با الگوریتم پرسپترون و ملایم سازی برای حل نامعادله خطی در حاقل سه روش متفاوت است :

۱- در بردار وزنی a و بردار مرزی b متفاوت است.

۲- علامت هایی از تفکیک ناپذیری را فراهم می کند.

۳- به محاسبات pseudoinverse از Y نیاز داریم.

اگر چه آخرین محاسبات مورد نیاز فقط یکبار انجام شده است، آن می تواند زمان مورد نظر باشد و به یک بحث ویژه ای نیاز است اگر $Y^{\dagger}Y$ منحصر به فرد باشد. یک الگوریتم تکراری مورد توجه که با معادله (۸۸) مقایسه شده است اما از محاسبات $Y^{\dagger}$ اجتناب می کنیم.

$b(۱)>۰$ درغیر این صورت دلخواه است (۹۰)

$a(۱) =$ دلخواه

$$b(k+۱) = b(k) + \eta\,(e(k) + |e(k)|)$$

$$a(k+۱) = a(k) + \eta R Y^{t}|e(k)|$$

در اینجا R دلخواه ، ثابت و ماتریس $\hat{d} * \hat{d}$ مثبت معین است . ما نشان می دهیم که اگر η ویژگی انتخاب شده است ، این الگوریتم یک بردار جواب در یک تعداد محدودی از مراحل بازدهی دارد و جواب موجود را فراهم می کند. بنابراین ، اگر هیچ راه حلی وجود نداشته باشد ، بردار $Y^{t}|e(k)|$ برداشته می شود و یا تحت اثر تفکیک ناپذیری قرار می گیرد و یا همگرا به صفر می شود. اثبات آن کاملا واضح است.اگرچه نمونه ها به صورت خطی تفکیک پذیر باشند یا نباشند. همان طور که درمعادله (۸۹) و (۹۰) نشان داده شده است.

e

ما می توانیم بدست آوریم که کمیت مربعات می شود :

و بنابراین

| (۹۱)

و در آن داریم :

A (۹۲)

واضح است اگر η مثبت باشد اما به قدر کافی کوچک باشد، A به طور تخمینی $۲\eta R$ خواهد شد و از اینجا به بعد مثبت معین است. بنابراین اگر $Y^{t}|e(k)| \neq ۰$ باشد ما $||e(k)||^{۲} > ||e(k+۱)||^{۲}$ را خواهیم داشت. بدین ترتیب ما باید وضعیت تفکیک پذیری و تفکیک ناپذیری را تشخیص بدهیم. در وضعیت تفکیک پذیری ، آنجا که $\hat{a}\,,\ \hat{b} > ۰$ را داریم باید $Y\hat{a} = \hat{b}$ برقرار باشد.بنابراین اگر $|e(k)| \neq ۰$ ، سپس :

|

بطوریکه $Y^t|e(k)|$صفر کمتر از زمانیکه e(k) صفر است نمی تواند باشد. بنابراین دنباله $||e(k)||^2, ||e(2)||^2, \ldots$ به صورت یکنواخت کاهش می یابد و باید به برخی از مقادیر همگرا شود. اما برای اینکه همگرایی اتفاق بیافتد ، $||e||^2$باید همگرا به صفر باشد و مستلزم |e(k)| است و از آنجا به بعد باید همگرا به صفر شود. از آنجا که e(k) با غیر مثبت شروع می شودو هرگز کاهش نمی یابد ، نشان می دهد که a(k) باید همگرا به بردارهای تفکیک پذیری همگرا شود. علاوه بر این با بحث یکسانی که قبلا بیان شد ، یک راه حل باید بعد از یک تعداد محدودی از مراحل بدست بیاید.

در وضعیت تفکیک ناپذیری e(k) هیچ گاه صفر نمی شود و یا همگرا به صفر نمی شود. البته $Y^t|e(k)| = 0$ در چندین مرحله ممکن است اتفاق بیافتد ، که در آن اثبات تفکیک ناپذیری فراهم می شود. به هر حال این ممکن است برای دنباله ای از بهسازی ها برای همیشه ادامه یابد. در این زمینه ، مجددا دنباله $||e(k)||^2, ||e(2)||^2, \ldots$ باید همگرا به مقدار محدود $||e||^2 \neq 0$ باشد و $Y^t|e(k)|$ باید همگرا به صفر باشد. بنابراین مجددا نشانه هایی از تفکیک ناپذیری در وضعیت جدا نشدنی را بدست می آوریم.

قبل از بستن این بحث ، یک نگاه خلاصه ای به سدوالاتی در باره انتخاب η و R داریم. ساده ترین انتخاب برای R ، ماتریس همانی است که به صورت رابطه $A = 2\eta R - \eta^2 Y^t Y$ می باشد. این ماتریس همچنین مثبت معین است. به این ترتیب از همگرایی مطمئن می شویم اگر $0 < \eta < \frac{2}{\lambda_{max}}$ و در آن λ_{max} بزرگترین بردار ویژه از $Y^t Y$ می باشد. از آنجا که اثر $Y^t Y$ جمع بردار ویژه $Y^t Y$ و جمع مربعات عناصر Y است ، ما از بدبینانه ترین مرز $\hat{d}\lambda_{max} \leq \sum_i ||y_i||^2$ در انتخاب مقادیر η استفاده می کنیم.

یک روش جالب تر ، تغییر دادن η در هر مرحله است انتخاب مقادیری که $||e(k)||^2 - ||e(k+1)||^2$ را ماکزیمم می کند. معادلات (۹۱) و (۹۲) به صورت زیر می شود:

|| (۹۳)

با توجه به اختلاف با η ، مقدار بهینه را بدست می آوریم :

(۹٤)

η

که آن را با R=I ساده می کنیم :

(۹٥)

η

یک روش یکسان برای انتخاب ماتریس R استفاده شده است. با جایگزین کردن R در معادله (۹۳) با ماتریس متقارن $R + \delta R$و صرف نظر کردن از جملات مرتبه دوم ، ما بدست می آوریم : .

δ

بنابراین کاهش بردار مربعات خطا با انتخاب زیر ماکزیمم می شود:

$$R = \frac{1}{\eta}(Y^t Y)^{-1} \qquad (۹٦)$$

و از انجا که $\eta R Y^t = Y^\dagger$ ، الگوریتم نزولی تقریبا با الگوریتم اصلی Ho-Kashyap یکسان می شود.

۵-۱۰- الگوریتم برنامه ریزی خطی

۵-۱۰-۱- برنامه ریزی خطی

در روش های پرسپترون ، ملایم سازی و Ho-Kashyap ، روش های کاهش شیب برای حل نابرابری خطی همزمان می باشند. تکنیک برنامه ریزی خطی ، روشی برای ماکزیمم یا مینیمم کردن تابع خطی به برابری خطی یا الزام نابرابری می باشد. در هر یک از این پیشنهادات ، ممکن است نابرابری خطی را با استفاده از آنها ، به عنوان محدودیتی در مشکل برنامه ریزی خطی مناسب حل کند. در این بخش دو روش را بررسی می کنیم . برای فهمیدن فرمالاسیون برنامه ریزی خطی نیاز به دانشی نیست. اگرچه دانش ما در بکاربردن تکنیک ها مفید می باشد.

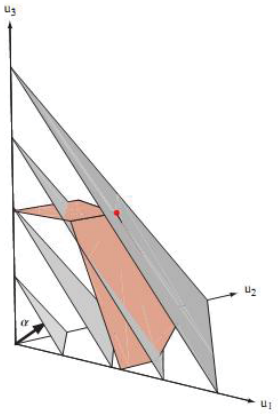
یک مشکل برنامه ریزی خطی کلاسیک می تواند این وضعیت باشد که پیدا کردن بردار $u = (u_1, \dots, u_m)^t$ که تابع اسکالر هدف خطی باشد.

z (۹۷)

با توجه به شرط الزام :

A (۹۸)

α یک بردار ارزشی m*۱ ، β یک بردار l*۱ و A ماتریس l*m بعدی می باشد. الگوریتم سیمپلکس ، یک روش تکراری کلاسیک برای حل این مشکل است.(شکل ۵-۱۸). با دلایل این تکنیک ، به تحمیل بیش از یک الزام نیاز داریم : U>=۰ .



شکل ۵-۱۸- سطوح ثابت $z = \alpha^t u$ به رنگ خاکستری نشان داده می شود، شکل الزام $Au.\beta$ به رنگ قرمز می باشد.

الگوریتم سیمپلکس ، حد نهایی z برای تعیین نابرابری پیدا می کند. به طور مثال صفحه خاکستری ، صفحه قرمز را از یک نقطه تک قطع می کند.

اگر u را به عنوان بردار وزنی a تصور کنیم ، این الزام قابل قبول نیست ، از آنجا که بیشترین زمینه حل بردار برای مولفه های هم مثبت و هم منفی خواهد بود. در هر حال با این فرض می نویسیم :

$$a = a^+ - a^-$$ (۹۹)

که در آن :

a (۱۰۰)

و

$$a \qquad (۱۰۱)$$

که a^+و a^-هر دو غیر مثبت هستند. و مولفه های u با مولفه های a^+و a^-شناسایی می شوند. برای مثال می توان نابرابری را با u>=۰ قبول کرد.

۵-۱۰-۲- وضعیت تفکیک پذیری خطی

فرض کنید یک مجموعه از n نمونه $y_۱,\ldots,y_n$داریم و بردار وزنی a با شرط $a^t y_i \geq b_i > ۰$برای همه i می خواهیم. چگونه می توان مشکل برنامه ریزی خطی را فرموله کرد؟ یک روش آن است که یک متغیر غیر واقعی بنام $\tau \geq ۰$بسلزیم و آنرا به صورت زیر بنویسیم :

$$a$$

اگر τ به قدر کافی بزرگ باشد ، هیچ مشکلی برای این نابرابری ها بوجود نمی آید. برای مثال اگر a=۰ و $\tau = max_i b_i$.* باشد آن الزامات انجام می شود. به هر حال ، حل مشکل اصلی سخت می باشد. آنچه که ما می خواهیم یک راه حل با $\tau = ۰$ می باشد که در آن کمترین مقدار τ می تواند $\tau \geq ۰$ باشد. بنابراین بنابراین ما مشکل را به این صورت دنبال می کنیم که :
مینیمم کردن τ طی همه مقادیر a با شرط $\tau \geq ۰$ و $a^t y_i \geq b_i$ بررسی می شود.
اگر جواب صفر است، نمونه ها به صورت خطی تفکیک پذیرند و یک راه حل داریم . اگر جواب ها مثبت هستند، هیچ بردار تفکیک کننده ای وجود ندارد، اما اثبات می کنیم که نمونه ها تفکیک پذیرند. از نظر شکل، مشکل پیدا کردن بردار u است که تابع اسکالر هدف $z = \alpha^t u$ را با الزام $u \geq ۰$, $Au \geq \beta$ مینیمم کند ، که در آن

$$A$$

بنابراین مشکل برنامه ریزی خطی متغیر $m = ۲\hat{d} + ۱$ و الزام l=n را شامل می شود. علاوه بر این ، الزام الگوریتم سیمپلکس u>=۰ می باشد. الگوریتم سیمپلکس مقدار مینیمم تابع هدف اسکالر $z = \alpha^t u = \tau$ در تعداد محدودی از مراحل پیدا می کند و نشان می دهد که بازدهی بردار $\hat{u}$یک مقدار است. اگر نمونه ها به صورت خطی تفکیک پذیر باشند، مینیمم مقدار τ صفر خواهد شد و بردار راه حل $\hat{a}$از $\hat{u}$بدست می آید. اگر نمونه ها جدایی پذیر نباشند، مینیمم مقدار τ مثبت خواهد شد. نتیجه $\hat{u}$معمولا برای تخمین راه حل ، خیلی مفید نمی باشد ، اما حداقل یک اثبات تفکیک ناپذیری بدست می آید.

۵-۱۰-۳-مینیمم کردن تابع مقیاس پرسپترون

در یک وسعت بزرگی از کاربردهای دسته بندی الگو ها نمی توان فرض کرد که نمونه ها ، جدایی پذیر خطی هستند. به ویژه زمانی که الگوها تفکیک ناپذیرند. فقط می خواهیم یک بردار وزنی بدست بیاوریم که دسته بندی را برای بیشترین نمونه های درست طبقه بندی کند. متاسفانه تعداد خطاها، تابع خطی مولفه های یک بردار وزنی نیست و مینیمم شدن آن مسئله برنامه ریزی خطی نیست. به هر حال مینیمم کردن تابع مقیاس پرسپترون ، به عنوان یک مشکل برنامه ریزی خطی مطرح می شود. از انجا که

مینیمم کردن تابع مقیاس یک بردار تفکیک پذیری را در وضعیت تفکیک پذیری تولید می کند و یک راه حل قابل قبول در وضعیت تفکیک ناپذیری است، این روش کاملا مورد توجه است. بیاد می آوریم که تابع مقیاس پرسپترون به صورت زیر داده شده است :

$$J_i \qquad (۱۰۲)$$

که در آن y(a) یک مجموعه از نمونه های آموزشی دسته بندی شده با a است . برای جلوگیری از راه حل های بدون اثر a=۰ ، یک بردار مرزی مثبت b معرفی می کنیم و می نویسیم :

$$J_i \qquad (۱۰۳)$$

که در آن $y_i \in Y'$ اگر $a^t y_i \leq b_i$. واضح است که J_p'یک تابع خطی تکه ای از a است، نه یک تابع خطی و تکنیک های برنامه ریزی خطی مستقیما بکار برده نمی شوند. به هر حال با تولید n متغیر غیرواقعی و الزام آنها ، می توان یک تابع هدف خطی برابر ساخت. با توجه به مسئله پیدا کردن بردارهای a و τ ، تابع خطی به صورت زیر مینیمم می شود:

$$a$$

با توجه به الزام

$$\tau$$

برای هر مقدار ثابت a ، مینیمم مقدار z دقیقا برابر با $J_p'(a)$است. با توجه به این الزام بهترین مقدار به صورت $\tau_i = \max[۰, b_i - a^t y_i]$می باشد. اگر z را بیشتر از t و a مینیمم کنیم ، باید مقدار مینیمم ممکن از $J_p'(a)$را بدست آوریم. بنابراین مسئله مینیمم سازی $J_p'(a)$برای مینیمم کردن تابع خطی z به الزام نابرابری تبدیل می کنیم. u_nبه بردار واحد n بعدی اشاره می کند. ما بدست می آوریم مسئله را با متغیرهای $m = ۲\hat{d} + n$و الزام l=n و مینیمم کردن $\alpha^t u$ با $u \geq ۰$, $Au \geq \beta$و ، که در آن :

با انتخاب a=۰ و $\tau_i = b_i$یک راه حل منصفانه اساسی برای شروع الگوریتم سیمپلکس تولید می کنیم و الگوریتم سیمپلکس یک $\hat{a}$برای مینیمم کردن $J_p'(a)$در تعداد مراحل محدود تهیه می کند.دو راه برای فرموله کردن مسئله پیدا کردن یک تابع تفکیک خطی به عنوان مشکلی برای برنامه ریزی خطی نشان می دهیم. فرمول های ممکن دیگری وجود دارد مانند مسئله دوئل (dual problem) که از نقطه نظر ریاضی بررسی می شود. به طور کلی صحبت کردن در مورد روش هایی مانند روش سیمپلکس ، روش های نزولی گرادیان پیچیده برای قرینه کردن تابع خطی به الزام خطی هستند.کد کردن الگوریتم های برنامه ریزی خطی پیچیده تر از کد کردن روش های نزولی ساده شده است. این روش های نزولی به طور طبیعی به شبکه عصبی چند لایه تعمیم داده می شود. به هر حال بسته های برنامه ریزی خطی پیشنهاد شده به طور مستقیم یا اصلاح شده مورد نظر با تلاش نسبتا کمی بدست می آید. وقتی این کار را انجام می دهیم از سود همگرایی تضمین شده برای مسائل تفکیک پذیر و تفکیک ناپذیر مطمئن

هستیم. الگوریتم های متنوعی برای پیدا کردن تابع تفکیک خطی در این فصل نشان داده شده است که در جدول ۵-۱ خلاصه شده است. البته سئوال می شود اینکه کدامیک بهتر است ، اما برتر بودن هیچ کدام یکسان نیست. انتخاب به مواردی نظیر بررسی مشخصات مورد نظر ، سهولت برنامه ریزی ، تعداد نمونه ها و فراوانی نمونه ها بستگی دارد. اگر یک تابع تفکیک پذیری خطی برای یک نرخ خطای کم بدست می آید. هر یک از این روش ها به طور هوشمندانه بکار برده می شود تا کارایی خوبی را فراهم کند.

جدول ۵-۱ : روش های نزولی برای توابع تفکیک پذیری خطی بدست آمده.

نام	مقیاس استاندارد	الگوریتم	شرایط
افزایشی ثابت			———
افزایشی متغیر		(	
ملایم سازی		(=	
Widrow-Hoff (LMS)		(	
تقریبی / احتمالی		(	
		(	
Pseudo-inverse			___
Ho-Kashyap		$e(k) = Ya(k) - b(k)$ $a(k) = Y^{\dagger}b(k)$	$b(۱)>۰$
		η = $a(k+۱) = a(k) + \eta RY^{t}\|e(k)\|$	R sym., pos.def. , $b(۱)>۰$

برنامه ریزی خطی	الگوریتم سیمپلکس	
	الگوریتم سیمپلکس	

۵-۱۱ – ماشین بردار پشتیبان

یک سری ماشین های خطی آموزشی با مرزبندی داریم. ماشین بردار پشتیبان (SVM) در بسیاری از مطالعات با تکیه بر پیش پردازش داد ها با نمایش الگوها در ابعاد بالا مورد توجه قرار گرفته است. با نگاشت غیر خطی اختصاص داده شده $\varphi()$به ابعاد بالای مورد نظر ، داده ها از دو کلاس با یک صفحه جدا می شوند (مسئله ۲۷).از انجا که فرض می کنیم هر الگوی x_kبه $y_k = \varphi(x_k)$تبدیل خواهد شد ، انتخاب $\varphi()$به صورت زیر است. برای هر n الگو ، k=۱,۲,…,n ما $z_k = \pm 1$داریم مطابق با هر الگویی از k که w_1 یا w_2 است. یک تفکیک کننده خطی در افزایش فضای y به صورت زیر است :

g (۱۰٤)

که در آن بردارهای وزنی و بردارهای الگوی تبدیل یافته افزایش یافته اند. (با $a_0 = w_0$و $y_0 = 1$ ، به ترتیب). بنابراین یک صفحه تفکیک پذیری به صورت زیر است :

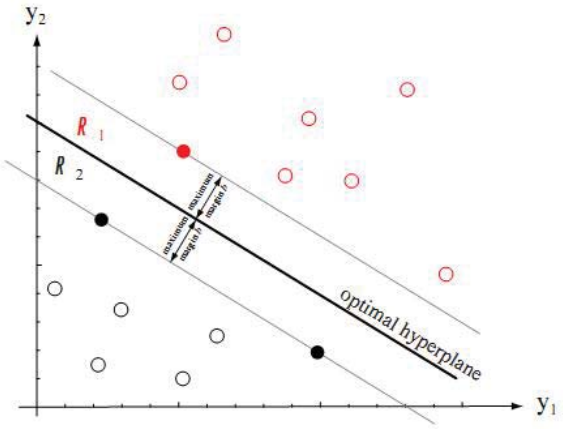
z (۱۰۵)

که در شکل ۵-۸ نشان داده شده است.

در این بخش مرزها فاصله مثبتی از صفحه تصمیم گیری بود. هدف ماشین بردار پشتیبان آموزشی پیدا کردن صفحه تفکیک پذیری با بزرگترین مرزها است. ما انتظار داریم که مرزهای بزرگتر ، تعمیم بهتری برای دسته بندی داشته باشند. همان طور که در شکل ۵-۲ نشان داده شده است، فاصله هر ابرصفحه با الگوهای y ، به صورت $\frac{|g(y)|}{||a||}$ است و می دانیم که حاشیه مثبت b موجود است (۱۰۵) به این دلالت دارد:

$\underline{z}$ (۱۰٦)

هدف از پیدا کردن بردار وزنی a ماکزیمم کردن b است.البته بردار جواب به صورت داخواه اندازه می شود و هنوز ابر صفحه حفظ می شود و بنابراین به یکتا بودن آن مطمئن هستیم که الزام $b||a|| = 1$ را می پذیریم. ما می خواهیم که جواب با معادلات (۱۰٤) و (۱۰۵) به صورت $||a||^2$مینیمم شود. بردارهای پشتیبان ، الگوهای آموزشی برای معادله (۱۰۵) را با تشابه نمایش می دهند. بردارهای پشتیبان به ابرصفحه نزدیک هستند(شکل ۵-۱۹).



شکل ۵-۱۹ : آموزش ماشین بردار پشتیبان شامل شده از ابرصفحه بهینه. به طور مثال ، ماکزیمم کردن فاصله از نزدیک ترین الگوهای آموزشی. بردارهای پشتیبان ، الگوهایی هستند که در فاصله b از صفحه می باشند و سه ماشین بردار پشتیبان در نقاط توپر نشان داده شده اند.

بردارهای پشتیبان نمونه های آموزشی هستند که صفحه تفکیک پذیری بهینه را تعریف می کنند و الگوهای بسیار مشکل را دسته بندی می کنند. با توجه به یک توضیح خصوصی ، آنها الگوهای بسیار آموزنده برای کارهای دسته بندی هستند. اگر N_Sبه مجموعه تعداد بردارهای پشتیبان اشاره کند، سپس برای الگوهای اموزشی n با ارزش منتظره از نرخ خطاهای تعمیمی کراندار است ، بنا به این گفته داریم :

$$\varepsilon \qquad (۱۰۷)$$

امیدواریم که همه بردارهای اموزشی با سایز n با توصیف توزیع شده این گروه بندی ترسیم شوند. این کران وابسته به ابعاد فضای بردار تبدیل یافته است که با $\varphi()$تعیین می شود. حالا به معادلات این بخش می پردازیم ، اما می دانیم که این شکل با میانگین خارج از محدوده مرزی می باشد. فرض کنید که n نقطه در بردار آموزشی داریم و یک ماشین بردار پشتیبان روی n-۱ تا از آنها انجام می دهیم و نقاط باقیمانده یکتا را تست می کنیم. اگر نقاط باقیمانده برای بردارهای پشتیبان با n نمونه کامل اتفاق بیافتد، سپس آنجا یک خطا خواهد بود در غیر اینصورت خزا نخواهیم داشت. اگر یک تبدیل $\varphi()$پیدا کنیم که همه داده ها را جدا کندبه طوریکه انتظار تعداد بردارهای پشتیبان کوچک را داریم سپس در معادله (۱۰۷) انتظار داریم که نرخ خطا کمتر شود.

۵-۱۱-۱- آموزش ماشین بردار پشتیبان

ما حالا مسئله آموزش SVM را بررسی می کنیم.در مرحله اول تابع φ-غیر خطی را انتخاب می کنیم که ورودی را با فضایی با ابعاد بالا نگاشت می کند. البته این انتخاب با دانش طراحی دامنه مسائل صورت خواهد گرفت. درصورت کمبود اطلاعات ، هر یکی ممکن است برای استفاده از چند جمله ای ، گوسین و یا توابع اصلی دیگر انتخاب شود. نگاشت ابعاد فضا می تواند به صورت دلخواه بالا باشد. محاسبات را مجددا برای مسائل مینیمم کردن بردارهای وزنی قابل سنجش انجام می دهیم تا با جدا کردن مسائل مشخص با روش های ضربی نامعین لاکرانژ. بنابراین با معادله (۱۰۶) و مینیمم کردن $\|a\|$ ما توابع را می سازیم :

(۱۰۸)

L

و مینیمم L() را باتوجه به بردارهای وزنی a جستجو می کنیم و با توجه به ضرب نامعین $\alpha_k \geq ۰$ ماکزیمم می کنیم.
آخرین جمله معادله (۱۰۸) هدف دسته بندی نقاط درست را بیان می کند. این استفاده از روشی را نشان می دهد که ساختن Kuhn-Tucker می نامیم. (تمرین ۳۰). این بهینه کردن می تواند برای ماکزیمم کردن به صورت زیر فرموله شود :

(۱۰۹)

L

و با توجه به الزام

(۱۱۰)

۲
۷
k

داده های آموزشی داده می شود. از آنجا که این معادلات با برنامه ریزی درجه دوم حل خواهد شد ، تعداد طرح های متناوب حدس زده می شود.

مثال ۲ : ماشین بردار پشتیبان برای مسئله XOR

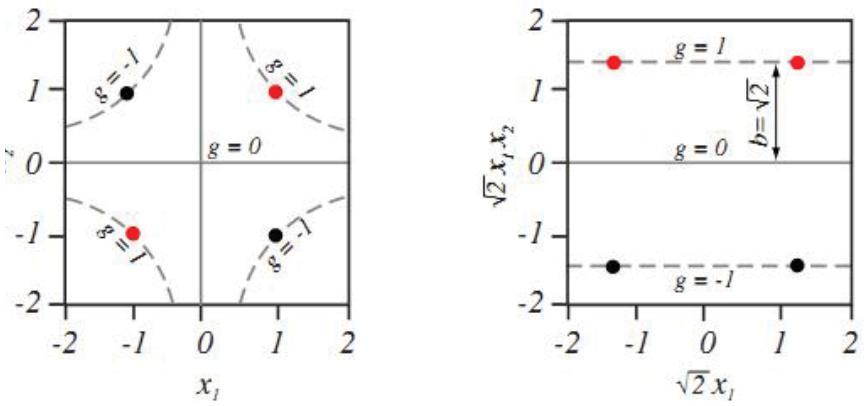
XOR یک مسئله ساده است که برای استفاده از عملگرهای تفکیک کننده خطی روی ویژگی ها به طور مستقیم نمی توان استفاده کرد. نقاط k=۱,۳ در $x = (۱,۱)^t$ و $x = (-۱,-۱)^t$در گروه $w_۱$هستند ، نقاط k=۲,٤ در $x = (۱,-۱)^t$ و $x = (-۱,۱)^t$در گروه $w_۲$هستند. با توجه به روش های ماشین بردار پشتیبان ، ما پیش پردازش می کنیم ویژگی هایی برای نگاشت به فضا با ابعاد بالا که آنها به صورت خطی می توانند جدا شوند. وقتی توابع φ زیادی استفاده می شوند، از ساده ترین بسط مرتبه دوم استفاده می کنیم : $۱, \sqrt{۲}x_۱, \sqrt{۲}x_۲, \sqrt{۲}x_۱x_۲, x_۱^۲, x_۲^۲$، که در آن $\sqrt{۲}$به آسانی نرمالیزه می شود. با ماکزیمم کردن معادله (۱۰۹) داریم :

۲
۷
k

و با توجه به الزام معادله (۱۱۰) داریم :

a
۰

تقارن مسئله با $\alpha_۱ = \alpha_۳$و $\alpha_۲ = \alpha_٤$در جواب کاملا واضح است. وقتی از نزولی گرادیان تکراری توصیف شده در بخش ۵-۹ برای مسئله کوچک استفاده می کنیم ، می توانیم از تکنیک تجزیه و تحلیل به جای آن استفاده کرد. جواب برای k=۱,۲,۳,٤ ، $\alpha_k^* = \frac{۱}{۸}$است. جملات آخر در معادله (۱۰۸) به این معنی می باشد که هر چهار الگوی آموزشی بردارهای پشتیبان هستند، در یک زمینه غیر معمول،یک تقارن طبیعی بالایی برای مسئله XOR داریم.تابع تفکیک نهایی $g(x) = g(x_۱, x_۲) = x_۱x_۲$ است و صفحه تصمیم گیری با g=۰ تعریف می شود بطوریکه دسته بندی مناسب برای همه الگوهای آموزشی را داریم. این مرز به آسانی برای راه حل ||a|| محاسبه می شود و با $b = \frac{۱}{||a||=\sqrt{۲}}$ بدست آمده است. شکل سمت راست تصویر مرزهای دو بعد از فضای تبدیل پنج بعدی را نشان می دهد. تمرین ۲۸ ، این مرزرا برای فضای ۲ بعدی دیگر به زیر فضا تخمین می زند.



یک هدف مهم دیگر از ماشین بردار پشتیبان ، پیچیدگی نتایج دسته بندی با تعداد بردارهای پشتیبان وابسته به ابعاد فضای تبدیل یافته مشخص می شود. مسئلهXOR در فضای ویژگی $x_1 - x_2$در شکل سمت چپ نشان داده شده است. دو الگوی قرمز در گروه w_1 هستند و دو الگوی سیاه در گروه w_2هستند. این چهار الگوی آموزشی X در یک فضای شش بعدی با $1, \sqrt{2}x_1, \sqrt{2}x_2, \sqrt{2}x_1x_2, x_1^2, x_2^2$نگاشت می شوند. در این فضا ابرصفحه بهینه با $g(x_1, x_2) = x_1x_2 = 0$پیدا می شود و مرز $b = \sqrt{2}$ است. یک تخمین دو بعدی از این فضا در شکل سمت راست نشان داده شده است. ابرصفحه بین بردارهای حمایت شده $\sqrt{2}x_1x_2 = \pm 1$است و مطابق با هذلولی $x_1x_2 = \pm 1$در فضای ویژگی اصلی است همان طور که نشان داده شده است.

۵-۱۲- تعمیم دهی چند کلاسی

۵-۱۲-۱ – ساختن Kesler

هیچ راه ثابتی برای برای گسترش دادن روش های دو کلاسی وجود ندارد به همین خاطر روش های چند گروهی بحث می شوند. در بخش ۵-۲-۲ ، دسته بندی چند گروهی را یک ماشین خطی می نامیم که دسته بندی الگو ها با محاسبه c تابع تفکیک خطی انجام می شود :

$$g$$

X را به گروهی مطابق با بزرگترین تفکیک تعیین می کنیم. این یک تعمیم طبیعی در زمینه چند کلاسی است به خصوص با نتایجی که در این فصل برای مسائل نرمال چند متغیره دیده شده است. به سادگی تابع تفکیک خطی تعمیم یافته را می توان بسط داد که در آن y(x) یک بردار $\hat{d}$بعدی از توابع x است که به صورت زیر نوشته می شود :

$$g \qquad (۱۱۱)$$

که x مجددا با w_iتعیین می شود اگر برای همه i ها بجز i≠j شرط $g_i(x) > g_j(x)$ برقرار باشد. تعمیم روش دسته بندی دو کلاسی برای ماشین خطی چند گروهی ساده ترین حالت جداکننده خطی می باشد. فرض کنید که یک مجموعه از نمونه های برچسب گذاری شده $y_1, y_2, \ldots, y_n$: با n_1 در زیر مجموعه برچسب گذاری شده Y_1، w_1باشد ، با n_2در زیر مجموعه برچسب گذاری شده Y_2، w_2 باشد و با n_c در زیر مجموعه برچسب گذاری شده Y_c، w_c باشد . ما می گوییم که این مجموعه

یک تفکیک کننده خطی است اگر یک ماشین خطی وجود داشته باشد که دسته بندی همه آنها درست باشد. اگر این نمونه ها به صورت خطی تفکیک پذیر باشند سپس یک مجموعه از بردارهای وزنی $\hat{a}_1 , \ldots . , \hat{a}_c$ وجود دارد که اگر $y_k \in Y_c$ سپس

$\hat{a}$ (۱۱۲)

برای همه i و j بجز i≠j .

یکی از موارد خوشایند درباره این تعریف این است که دستکاری کردن این نامعادله و کاهش مسائل چند گروهی به مسائل دو گروهی ممکن است. فرض کنید برای هر لحظه ای که $y \in Y_1$ بطوریکه معادله (۱۱۲) می شود :

$\hat{a}$ (۱۱۳)

یک مجموعه با c–۱ عضو نامساوی می تواند به عنوان نیازی از بردار وزنی $\widehat{cd}$ بعدی باشد.

$\hat{a}$

برای دسته بندی درست همه c–۱ عضو از مجموعه نمونه های $\widehat{cd}$ بعدی داریم :

η

به عبارت دیگر هر η_{1j} مطابق با نرمال کردن الگو های w_1 و w_2 است. به طور عمومی تر ، اگر $y \in Y_i$ را داشته باشیم ، یک η_{ij}، نمونه های آموزشی (c–۱) $c\hat{d}$ –بعدی با افراز η_{ij} در داخل زیر بردارهای $c\hat{d}$–بعدی می سازیم که زیربردار i ام y می شود وزیر بردار j ام y- می شود و مابقی صفر می شوند. واضح است که اگر $\hat{a}^t\eta_{ij} > 0$ برای i≠j باشد سپس ماشین خطی مطابق با مولفه های $\hat{a}$، y را به طور درست طبقه بندی می کند. روش ساختن Kesler داده های c بعد و تعدادی از نمونه های c–۱ را ضرب می کند که بطور مستقیم نمی توان آن را بدست آورد. تبدیل روش های بهسازی خطا چندگروهی به روش های دو گروهی برای بدست آوردن اثبات همگرایی بسیار مهم است.

۵–۱۲–۲ – همگرایی قانون افزایش ثابت

ما از روش ساختن Kesler برای بهبود دادن همگرایی تعمیم قانون افزایشی ثابت جهت یک ماشین خطی استفاده می کنیم . فرض کنید یک مجموعه n تایی از نمونه های تفکیک پذیر خطی $y_1, y_2, \ldots, y_n$ داریم و و از آنها برای شکل دهی دنباله نامحدود که هر نمونه نامحدود به نظر می رسد ، استفاده می کنیم. L_k ماشین خطی را نشان می دهد که بردارهای وزنی آن $a_1(k), \ldots, a_c(k)$ هستند. با شروع از یک ماشین اولیه دلخواه L_1 یک دنباله از نمونه ها را برای ساختن یک دنباله از ماشین خطی می خواهیم که به ماشین جواب همگرا شود تا نمونه ها را به صورت درست دسته بندی کند. ما باید پیشنهاد کنیم که قانون تصحیح خطا با تغییر وزن ها ایجاد می شود اگر و فقط اگر ماشین خطی یک نمونه را اشتباه دسته بندی کرده باشد. y^k نمونه صحیح k ام است و فرض کنید که $y^k \in Y_i$ از آنجا که y^k نیاز می باشد که صحیح باشد، حداقل برای یک i≠j داریم :

a (۱۱٤)

سپس قانون افزایشی ثابت برای تصحیح L_k می شود :

(۱۱۵)

a

بردارهای وزنی برای گروه مورد نظر با الگوها به صورت افزایشی است، بردارهای وزنی برای گروه های انتخاب شده اشتباه کاهشی است. بردارهای وزنی دیگر بدون تغییر هستند.

حال باید نشان دهیم که این قانون باید ما را بعد از تعداد محدودی از بهسازی ها به یک ماشین جواب هدایت کند. اثبات ساده است . برای هر ماشین خطی L_kمطابق با بردار وزنی داریم :

a

برای هر نمونه $y \in Y_i$، ۱–c نمونه η_{ij}وجود دارد که در این بخش توصیف شده است. بویژه مطابق با هر بردار y^kدر معادله (۱۱٤) یک بردار وجود دارد که

$$\eta^k{}_{ij} = \begin{bmatrix} \vdots \\ y^k \\ \vdots \\ -y^k \end{bmatrix} \begin{matrix} \\ \leftarrow \text{ i} \\ \\ \leftarrow \text{ j} \end{matrix}$$

که برآورده می کند :

علاوه بر این ، قوانین افزایشی ثابت برای تصحیح L_k، قانون افزایشی ثابت برای تصحیح $\alpha(k)$ می باشد، بطور نمونه :

بنابراین یک تطابق کامل بین وضعیت چندگروهی و وضعیت دو گروهی بدست می آوریم که در آن روش چند گروهی یک دنباله از نمونه ها $\eta^1, \eta^2, \ldots, \eta^k, \ldots$ و یک دنباله از بردارهای وزنی $\alpha_1, \alpha_2, \ldots, \alpha_k, \ldots$را تولید می کند. با نتایج بدست آمده در حالت دو گروهی ، دنباله نهایی نامتناهی نیست، اما باید به یک بردار جواب خاتمه یابدو از آنجا که ، دنباله $L_1, L_2, \ldots, L_k, \ldots$باید با یک ماشین جواب بعد از یک تعداد محدودی از تصحیح شده ها خاتمه یابد. از روش ساختن Kesler برای ایجاد کردن برابری بین روش های چندگروهی و دو گروهی استفاده می کنیم که یک ابزار تئوری قدرتمند است. این همچنین می تواند برای گسترش دادن همه نتایج با روش های پرسپترون و ملایم سازی در زمینه چند پروهی باشد وقوانین تصحیح خطا را برای روش های توابع پتانسیلی بکار ببریم. متاسفانه ، این روش به طور مستقیم برای تعمیم دادن MSE یا روش های برنامه ریزی خطی مفید نیست.

۵–۱۲–۳–تعمیم روش های MSE

شاید ساده ترین راه بدست آوردن تعمیم روش های MSE در زمینه های چند کلاسی بررسی کردن مسئله یک مجموعه از c تا مشکل دو کلاسی است. مشکل i ام با بردارهای وزنی a_i بدست می آید راه حل مینیمم مربعات خطا است با معادله :

a

a

هدف این نتایج در بخش ۵-۸-۳ ، برای تعدادی از نمونه هایی است که خیلی بزرگ هستند ، یک تخمین مینیمم میانگین مربعات خطا را با تابع تفکیک پذیری بیزین بدست خواهیم آورد.

$$F$$

مشاهده شده است که دو نتیجه بدست آمده است . اولا ، یک تغییر جزئی پیشنهاد شده است که یک بردار وزنی a_iرا جستجو کنیم که یک جواب مینیمم مربعات خطا است با معادله :

$$a \qquad (۱۱۶)$$

بطوریکه $a^t y$خمین مینیمم میانگین مربعات خطا با $P(w_i|x)$خواهد شد . ثانیا ، استفاده از نتایج توابع تفکیک پذیری در ماشین خطی را توجیه می کند بطوریکه y برای w_iتعیین می شود اگر برای همه i ها بجز i≠j رابطه $a_i^t y > a_j^t y$ برقرار باشد. جواب pseudoinverse برای مسائل چند کلاسی MSE در یک قالب متناظر با قالب دو کلاسی نوشته شده است. اجازه بدهید که y یک ماتریس $n * \hat{d}$ از نمونه های آموزشی باشد که فرض می کنیم به صورت پارتیشن بندی به صورت زیر باشد :

(۱۱۷)

$$Y$$

نمونه ها با w_iشامل سطرهایی از Y_iبرچسب گذاری می شوند. به طور مشابه A ، یک ماتریس $\hat{d} * c$بعدی از بردارهای وزنی است.

$$A = [a_۱ a_۲ \ \dots \ \ a_c], \qquad (۱۱۸)$$

و B یک ماتریس n*c می باشد

(۱۱۹)

$$B$$

همه عناصر B_iصفر هستند بجز آنهایی که در ستون i ام هستند .سپس اثر ماتریس خطای مربعات $(YA-B)^t * (YA-B)$با راه حل زیر مینیمم می شود

$$A \qquad (۱۲۰)$$

به طور معمول ، $Y^\dagger$ با محاسبات pseudoinverse از Y است. این نتیجه یک روش تئوری را تعمیم می دهد.λ_{ij} تاثیر کمی برای تصمیم گیری w_i دارد وقتی که وضعیت درست w_j است. زیر ماتریس j ام از B به صورت زیر است :

$$B_j = -\begin{bmatrix} \lambda_{۱j} & \lambda_{۲j} & \dots & \lambda_{cj} \\ \lambda_{۱j} & \lambda_{۲j} & \dots & \lambda_{cj} \\ \vdots & \vdots & \dots & \vdots \\ \lambda_{۱j} & \lambda_{۲j} & \dots & \lambda_{cj} \end{bmatrix} \begin{matrix} \uparrow \\ n_j \\ \downarrow \end{matrix} \qquad \text{j=۱, ... , c.} \qquad (۱۲۱)$$

سپس به عنوان تعدادی از نمونه های غیرپایانی ، جواب $A = Y^\dagger B$ توابع تفکیک پذیری $a_i^t y$ یک تخمین مینیمم میانگین مربعات خطا را با توابع تفکیک پذیری بیزین فراهم می کند.

(۱۲۲)

g

توزیع مستقیم اثبات در بخش ۵-۸-۳ داده شده است (تمرین۳٤)

خلاصه :

در این فصل توابع تفکیک پذیری بررسی می شوند که یک تابع خطی از یک مجموعه پارامترها هستند، که معمولا وزن نامیده می شوند. در همه وضعیت های دو گروهی مانند تفکیک پذیری ، یک مرز تصمیم گیری ابرصفحه داریم که هر یک در فضای ویژگی خودش می باشد و یا در فضایی است که ویژگی ها با توابع غیر خطی نگاشت شده اند (تفکیک پذیری خطی معمولی).پیبا این نقطه نظر اصلی ، تکنیک هایی نظیر الگوریتم پرسپترون ، پارامترها را با افزایش ضرب داخلی به الگوهایی در کلاس w_1تطبیق می دهد و با کاهش ضرب داخلی در کلاس w_2تطبیق می دهد.یک روش بسیار معمولی با توابع مقیاس شکل گرفته است و شیب نزولی را فراهم می کند. توابع مقیاس متفاوتی با میزان پایداری متفاوت داریم که نقطه ضعف ها مرتبط با محاسبات و همگرایی است. یکی از روش های جبری خطی برای حل وزن ها به طور مستقیم با میانگین ماتریس pseudoinverse برای مسائل کوچک استفاده می شود. در ماشین های بردار پشتیبان ، ورودی های با توابع خطی به فضایی با ابعاد بالا نگاشت می شوند و یک ابرصفحه بهینه پیدا می شود که بزرگتریم مرز را دارد. بردارهای گشتیبان آن دسته از الگوهایی هستند که با مرزها تعیین می شوند؛ آنها مخصوص سخت ترین الگوها برای دسته بندی هستند و بیشترین آموزندگی را برای طراحی دسته بندی دارند. یک مرز بالایی نرخ خطای مورد نظر از دسته بندی به صورت خطی به تعداد بردارهای پشتیبان مورد نظر وابسته است. برای مسائل چند کلاسی ، ماشین های خطی مرزهای تصمیم گیری شامل بخش هایی نظیر ابرصفحه را بوجود می آورند. همگرایی الگوریتم های چند کلاسی را می توان با تبدیل آنها به الگوریتم های دو کلاسی آمتحان نمود و با حالت دو کلاسی اثبات نمود. الگوریتم سیمپلکس ، توابع خطی بهینه را برای الزام پیدا می کند و برای دسته بندی خطی آموزشی می تواند استفاده شود . تفکیک کننده خطی ، جایی که مورد استفاده است به اندازه کافی برای مسائل بازشناسی الگو مورد چالش قرار نمی گیرد. مگر اینکه یک نگاشت غیر خطی مناسب (تابع φ)پیدا شده باشد.

ملاحظات تاریخی و فهرست کتب :

از آنجا که توابع تفکیک کننده خطی موظف به آنالیز هستند ، هر مقاله نوشته شده ای درباره آنها مناسب برای ارائه می باشد. از نظر تاریخی ، همه کارهای انجام شده در مقاله کلاسیک بوسیله Ronald A. Fisher شروع شده است. کاربرد توابع خطی دسته بندی الگوها به خوبی در[۷] توصیف شده است که در آن مسئله بهینه سازی تفکیک خطی مطرح شده است و روش های نزولی شیب محتمل برای تعیین یک حل از نمونه ها پیشنهاد شده است. متاسفانه به طور مغرضانه درباره بسیاری از روش ها بدون توزیع بنیادی می توان نظر داد و سپس این مسائل تجزیه و تحلیل پیچیده ای دارند. طراحی دسته بندی چند گروهی مورد استفاده در روش های دو گروهی در [۱۲] آمده است. روش پرسپترون Minskyand Papert [۱۱] برای نشان دادن ضعف دسته بندی خطی موثر بود. این نقطه ضعف با توجه به روش هایی که در این فصل مطالعه شد ، برطرف گردید. الگوریتم Winnow [۸] در وضعیت خطای آزاد و [۹,٦] و کارهای متعاقب آن در زمینه های معمولی در محیط های یادگیری محاسباتی

مفید می باشد بطوریکه اجازه مشتق شدن کران های همگرایی را می دهد. از آنجا که این کار از نظر آماری مورد بررسی قرار می گیرد ، بسیاری از مقالات بازشناسی الگو بعد از ۱۹۵۰ و قبل از ۱۹۶۰ با این دیدگاه اتخاذ می شوند. یک دیدگاه در شبکه های عصبی به این گونه بود که نرون های جداگانه به عنوان عناصر آستانه ای مدل شده اند. . ماشین های خطی دو گروهی ، مبدا مقالات مشهور توسط McCullo ch and Pitts بوده است. یک ماشین خطی برای مجموعه داده های بزرگتر و بزرگتر در ابعاد بالاتر وبالاتر بکار برده شده است ، بار محاسباتی برنامه ریزی خطی [۲] روش هایی با محبوبیت کمتر می باشد. تخمین های احتمالی ، مثال [۱۵] . مقالات اخیر ، در زمینه های کلیدی با ماشین بردار پشتیبان [۱] است. یک رفتار گسترش یافته ، شامل کنترل پیچیدگی در [۱٤] دیده شده است. مطالب است که در فصل ؟؟ باید دده شود. یک نمایش قابل قبول از این روش ها در [۳] آمده است که در مثال ۲ بیان شده است. روش ساختن Kuhn-Tucker ، در روش های آموزش بردار ماشین پشتیبان در متن آمده است و در تمرین (۳۰) مورد بررسی و امتحان قرار می گیرد. کل در [۳] آمده است و در [۱۳] استفاده شده است. نتایج بنیادی در سه زمینه مطرح شده است : ۱- شرایط اصلی و اولیه یک راه حل بهینه دارد. در این زمینه وضعیت های زوجی همچنین انجام شده است و مقادیر اصلی آنها مساوی است. ۲- شرایط اصلی و اولیه اجرا نشدنی ها است ، در این زمینه ، هریک از عدم وجود کران ها ، زوجی می شود و یا اجرانشدنی است . ۳- شرایط اصلی و اولیه ، بدون کران دار بودن است که در این زمینه زوجی ها اجرا نشدنی است.

تمرینات

⊕ بخش ۵-۲

۱- یک ماشین خطی با تابع تفکیک پذیری $g_i(x) = w^t x + w_{i.} \quad i = ۱, \dots, c$ را بررسی کنید. نشان دهید که ناحیه تصمیم گیری ، محدب است اگر $x_۱ \in R_i$ و$x_۲ \in R_i$باشد و سپس $\lambda x_۱ + (۱ - \lambda)x_۲ \in R_i$را داریم. اگر $۰ \leq \lambda \leq ۱$را داشته باشیم.

۲- شکل ۵-۳ نشان می دهد که دو روش مشهور برای طراحی دسته بندی c-کلاس با بخش های مرزی خطی است. یک روش دیگر استفاده کامل $\binom{c}{۲}$با مرز خطی $\frac{w_i}{w_j}$ است و دسته بندی هر نقطه با بردارهای مبتنی بر مرز می باشد.اثبات کنید نتایج ناحیه تصمیم گیری محدب است. اگر محدب نباشد، یک مثال غیر پزشکی در حداقل یک ناحیه تصمیم گیری غیر محدب بیاورید.

۳- ابر صفحه ای را برای تابع تفکیک پذیری بررسی کنید.

(a) نشان دهید که فاصله ابرصفحه $g(x) = w^t x + w_{.} = ۰$ با نقطه x_a، $\frac{|g(x_a)|}{||w||}$است که با مینیمم کردن $||x - x_a||^۲$برای الزام g(x)=۰ می باشد.

(b) نشان دهید که تخمین x_aدر ابر صفحه به صورت زیر است :

٤- یک ماشین خطی ۳ کلاسی با تابع تفکیک $g_i(x) = {w_i}^t x + w_{i.} \quad i = ۱, ۲, ۳$ بررسی کنید.

(a) برایس جایی که x دو بعدی است و آستانه وزنی w_i، صفر است . طرح اولیه بردارهای وزنی با دنباله اصلی است. خطوط سه تایی برای اتصالات استفاده می شود و مرز تصمیم گیری داریم.

(b) چگونه تغییرات طرح اولیه وقتی که بردار ثابت c به هر بردار وزنی سه تایی اضافه می شود؟

۵- در زمینه چند کلاسی ، یک مجموعه از نمونه ها ، جدا کننده خطی می باشد، اگر یک ماشین خطی داشته باشیم که دسته بندی را به درستی انجام دهد. اگر همه نمونه ها با w_i برچسب داشته باشند ، می توان آن ها را با ابر صفحه جدا کرد. این نمونه ها تفکیک پذیری خطی کلی هستند. اما محدب بودن لازم نیست.

۶- یک مجموعه از نمونه ها تفکیک پذیری خطی دوبدو گفته می شوند اگر $\frac{c(c-1)}{2}$ ابرصفحه H_{ij} بطوریکه H_{ij} برای جدا کردن نمونه های w_i از w_j باشد. نشان دهید که مجموعه تفکیک پذیری خطی جفتی الگو ها تفکیک پذیر نیستند.

۷- اگر $y_1, y_2, \dots, y_n$ یک مجموعه متناهی از نمونه های آموزشی خطی جدا کننده باشد و a را بردار جواب بنامیم .اگر $a^t y_i \geq 0$ برای همه i باشد. نشان دهید که بردار جواب مینیمم طول مینیمم منحصر به فرد است.

۸- convex hull یک مجموعه از بردار های $x_i , i = 1, \dots, n$، مجموعه ای از بردار های به صورت زیر است:

که در آن ضریب α_i غیر منفی است و مجموعه آن یک است. دو مجموعه از بردار ها را داریم : نشان دهید که تفکیک پذیر خطی هستند و convex hulls آنها را قطع می کند.

۹-یک دسته بندی یک ماشین خطی خطی تکه ای نامیده می شود اگر توابع تفکیک پذیری به صورت زیر باشد :

که در آن

(a) نشان دهید مه چگونه ماشین خطی جفتی در ترم هایی از ماشین خطی برای دسته بندی زیر کلاس های الگو ها دیده می شود.

(b) نشان دهید که ناحیه نصمیم گیری برای ماشین خطی جفتی می تواند غیر محدب باشد .

(c) طرح اولیه نقشه $g_{ij}(x)$ برای مثال های یک بعدی در $n_1 = 2$ و $n_2 = 1$ برای جواب بخش b نشان دهید.

۱۰- d مولفه از x به صورت ۰ یا ۱ داریم . فرض کنید که که x را با w_1 علامت گذاری می کنیم اگر تعداد مولفه های غیر صفر x اضافه شده باشد. در غیر این صورت w_2 را داریم.

(a) نشان دهید اگر d>۱ را داشته باشیم ، دوبخشی سازی ، تفکیک پذیر خطی نیست.

(b)نشان دهید که این مسئله با ماشین خطی جفتی با d+۱ بردار های وزنی w_{ij} حل می شود

بخش ۵-۳

۱۱- تابع تفکیک مربعی را بررسی کنید

که متقارن باشد و ماتریس غیر منحصر بفرد $W = [w_{ij}]$را داریم . نشان دهید که ویژگی اصلی مرز تصمیم گیری در ترم هایی از ماتریس عددی $\bar{W} = W/(w^t W^{-1} w - \varepsilon w_.)$را به صورت زیر می توان توصیف کرد :

(a) اگر $\bar{W} \propto I$، سپس مرز تصمیم گیری hypersphere است.

(b) اگر $\bar{W}$مثبت باشد، سپس مرز تصمیم گیری hyperellipsoid است.

(c) اگر مقدار ویژه $\bar{W}$مثبت و با بعضی منفی باشند، سپس مرز تصمیم گیری ، hyperhyperboloid است.

(d) فرض کنید $W = \begin{pmatrix} ۵ \\ ۲ \\ -۳ \end{pmatrix}$و $W = \begin{pmatrix} ۱ & ۲ & ۰ \\ ۲ & ۵ & ۱ \\ ۰ & ۱ & -۳ \end{pmatrix}$را داریم . ویژگی های جواب چیست؟

(e) بخش d را برای $W = \begin{pmatrix} ۲ \\ -۱ \\ ۳ \end{pmatrix}$و $W = \begin{pmatrix} ۱ & ۲ & ۳ \\ ۲ & ۰ & ٤ \\ ۳ & ٤ & -۵ \end{pmatrix}$تکرار کنید؟

۱۲- استنباط کنید که در معادله (۱٤) $J(۰)$ به مراحل تکرار k وابسته است.

۱۳- تابع مقیاس مجمعه مربعات خطا را در معادله (٤۳) بررسی کنید.

$b = b_i$را داریم و شش نقطه آموزشی به صورت زیر داریم :

ν

ν

(a) ماتریس Hessian را برای این مسئله بررسی کنید.

(b)مشخص کنید تابع استاندارد مربعی با نرخ یادگیری بهینه η محاسبه می شود.

بخش ۵-۵

۱٤- در اثبات همگرایی الگوریتم پرسپترون فاکتور مقیاس αبه صورت β^2/γبوده است.

(a) با توجه به بخش ۵-۵ نشان دهید که اگر αبزرگتر از $\beta^2/2\gamma$ ماکزیمم تعداد بهسازی باشد، رابطه زیر را داریم :

(b) اگر $a_۱ = ۰$، مقدار αکه $k_.$رامینیمم می کند چیست؟

۱۵- با اثبات همگرایی در بخش ۵-۵-۲ ، اثبات همگرایی روش بهسازی زیر را بدست آورید : با شروع از بردارهای وزنی اولیه دلخواه $a_۱$ و تصحیح a(k) مطابق :

a

اگر و فقط اگر $a^t(k)y^k$ مرز b به اندازه کافی صحیح نباشدو $\eta(k)$ با $0 < \eta_a \leq \eta(k) \leq \eta_b < \infty$ مرز بندی می شود.

اگر bمنفی باشد چه اتفاقی می افتد؟

۱۶- اگر $y_1, y_2, \ldots, y_n$ یک مجموعه متناهی از نمونه های تفکیک پذیری خطی d-بعدی باشد :

(a) یک روش کامل و دقیق پیشنهاد کنید که یک بردار تفکیک پذیری در تعداد محدودی از مراحل پیدا شود.

(b) پیچیدگی محاسباتی این روش چیست؟

۱۷- با بررسی کردن تابع مقیاس

که در آن $Y(a)$ یک مجموعه از نمونه ها با $a^t y < b$ است . فرض کنید که y_1 یک نمونه در $Y(a(k))$ است. نشان دهید که $\nabla J_q(a(k)) = 2[(a^t(k)y_1 - b)y_1]$ و ماتریس ناتمام دوم اشتقاق یافته با $D = 2y_1y_1^t$ وجود دارد . توجه کنید که بهینه بودن $\eta(k)$ با الگوریتم نزولی شیب استفاده شده است.

$$a(k+1) = a(k) + \eta \frac{b - a^t y_1}{\|y_1\|^2} y_1$$

۱۸- با توجه به شرایط معادلات ۲۸-۳۰ ، نشان دهید که $a(k)$ در افزایش متغیرهای قانون نزولی ، عملا برای $a^t y_i > b$ همگرا می شود.

⊕ بخش ۵-۶

۱۹- طرح اولیه یک شکل برای اثبات در بخش ۵-۶-۲ نشان داده شده است.مطمئن شوید که یک وضعیت معمولی بدست آورده ایم و همه متغیرها برچسب دارند.

⊕ بخش ۵-۷

⊕ بخش ۵-۸

۲۰- نشان دهید که فاکتور مقیاس α در حل MSE مطابق با تفکیک خطی فیشر بخ صورت زیر داده شده است :

۲۱- نتیجه بخش ۵-۸-۳ را برای نشان دادن بردار a که تابع مقیاس را مینیمم می کند تعمیم دهید :

$$J$$

مجانب را برای تخمین خطای مینیمم مربعات خطا با تابع تفکیک بیزین $(\lambda_{21} - \lambda_{11})P(w_1|x) - (\lambda_{12} - \lambda_{22})P(w_2|x)$ تولید کنید.

۲۲- با بررسی تابع مقیاس $J_m(a) = \varepsilon[(a^t y - z)^2]$ و تابع تفکیک بیزین نشان دهید که :

(a) نشان دهید که :

$$J_1$$

(b)با استفاده از اینکه میانگین شرطی z ، $g_.(x)$است نشان دهید که J_mو $\varepsilon[(a^t y - z)^2]$را مینیمم می کند.

۲۳-یک تحلیل عددی از ارتباط $R^{-1}(k+1) = R^{-1}(k) + y_k y_k^t$در تخمین احتمالی $\eta^{-1}(k+1) = \eta^{-1}(k) + y_k^2$ است .

(a) نشان دهید که جواب closed form را داریم :

η

(b) فرض کنید که $\eta(1)>0$ و $0 < a \le y_i^2 \le b < \infty$را داریم ، نشان دهید که چرا دنباله ضرائب به صورت $\sum \eta^2(k) \to L < \infty$ و $\sum \eta(k) \to \infty$می باشد.

۲٤- نشان دهید که برای قانون LMS یا Widrow-Hoff ، اگر $\eta(k) = \frac{\eta(1)}{k}$باشد سپس دنباله بردارهای وزنی به یک بردار محدود شده a با توجه به $Y^\dagger(Ya - b) = 0$ همگرا می شود.

بخش ۵-۹

۲۵- بررسی کنید که شش نقطه داده زیر :

ν
ν

(a) آیا تفکیک پذیر خطی هستند؟

(b) با استفاده از روش داده شده در متن ، فرض کنید R=I ، ماتریس همانی ، نرخ یادگیری بهینه معادله (۸۵) را محاسبه کنید.

۲٦- مسئله برنامه ریزی خطی فرموله شده در بخش ۵-۱۰-۲ شا مل مینیمم کردن متغیر غیرواقعی تک τ تحت الزام $b \ge 0$و $a^t y_i + \tau \ge b_i$می باشد. نشان دهید که بردارهای وزنی نتیجه شده تابع مقیاس را مینیم می کند

بخش ۵-۱۱

۲۷ – توضیح دهید که از نظر کیفیت اگر نمونه ها دو کلاسی مجزا باشند یک سری نگاشت غیر خطی داریم با ابعاد بالاتر که نقاط تفکیک پذیر خطی را رد می کنند.

۲۸- شکل مثال ۲ ، مرز ماکزیمم ماشین بردار پشتیبان بکار رفته برای XOR را به فضای ۵-بعدی نگاشت می کند. شکل الگو های آموزشی و کانتورهای تابع تفکیک به عنوان تخمینی از زیرفضای دو بعدی تعریف شده با ویژگی های $\sqrt{2}x_1, \sqrt{2}x_1x_2$ نشان می دهد. بدون در نظر گرفتن ویژگی های ثابت و با بررسی چهار ویژگی دیگر ، برای هر یک از جفت های ویژگی $5 = 1 - \binom{4}{2}$که د مثال نشان داده شده است ، نشان دهید که الگو ها و خط ها مطابق با تفکیک $g = \pm 1$می باشند. آیا مرزها در شکل شما یکسان هستند؟ علت را توضیح دهید.

۲۹- یک ماشین بردار پشتیبان با داده های آموزشی دو کلاسی را در نظر بگیرید.

Category		

۱ ۲ ۰	۱ ۲ ۲	
۰ ۰ ۱	۰ ۱ ۰	

(a) این شش نقطه آموزشی را رسم کنیدو با بازبینی بردار وزنی برای ابرصفحه بهینه و مرزهای بهینه.

(b)بردارهای پشتیبان کدام هستند؟

(c)جواب را در فضای دوگان با پیدا کردن ضرب کننده لاکرانژ تعیین نشده α_i . نتایج را با بخش (a) مقایسه کنید.

۳۰- این مسئله درباره قضیه Kuhn-Tucker برای انتقال مسائل بهینه اجباری ماشین بردار پشتیبان به دوگان می باشد. برای ماشین بردار پشتیبان، هدف پیدا کردن بردار وزنی مینیمم a با محدوده دسته بندی

z

که ۱$\pm = z_k$ به کلاس هدف از هر n الگو y_k می باشد . توجه کنید که a و y افزوده شده اند. ($a.$ و $y.$ تکرار می شوند)

(a) بهینه سازی غیر اجباری وابسته به ماشین بردار پشتیبان را بررسی کنید:

L

در فضای تعیین شده با مولفه های a و n ضرب کننده مشخص نشده α_k . جواب های مورد نظر یک نقطه تحمیلی نسبت به ماکزیمم یا مینیمم کلی است. توضیح دهید.

(b)تخمین بعدی وابسته به تابع نزدیک a به طور مثال بهینه سازی فرموله شده در فرم دوگان ، با دنباله ای از مراحل انجام می شود. توجه کنید که نقطه تحمیلی از مراحل اولیه است ، ما داریم :

$\acute{c}$

مشتق جزئی را حل کنید و خاتمه دهید:

$\mathbf{\Sigma}$

$\mathbf{\Sigma}_k$

(c) اثبات کنید که نقطه گوس دار ، ابرصفحه بهینه است که یک ترکیب خطی از بردارهای آموزشی می باشد :

c

(d) با توجه به قضیه Kuhn-Tucker theorem ، یک ضرب کننده مشخص نشده a^* غیر صفر است اگر و فقط اگر نمونه های تطبیقی y_k ، $z_k \alpha^*{}_k y_k = ۰$ را ارضا کند.نشان دهید که به صورت زیر بیان می شود :

c

(نمونه های برای a^* غیر صفر هستند به طور مثال $z_k \alpha^{*t} y_k = ۱$ بردارهای پشتیبان هستند.)

(e) از نتایج بخش (b) و (c) برای تخمین بردارهای وزنی به طور عملی استفاده کنید وبنابراین دوگان عملی را بسازید :

$\bar{L}$

(f) حل a^* از بخش (c) را برای پیدا کردن دوگان عوض کنید:

$\bar{L}$

۳۱- مثال ۲ را تکرار کنید.

بخش ۵-۱۲

۳۲- فرض کنید که هر نقطه آموزشی ۲ بعدی y_i در کلاس w_1 و یک نقطه تطبیقی متقارن w_2 در $-y_i$ وجود دارد.

(a) اثبات کنید که ابرصفحه جدا کننده یا جواب LMS باید اصلی و تازه باشد.

(b)نظیر تقارن شش نقطه با نقاط زیر را بررسی کنید :

и

и

شرایط ریاضی y را نظیر حل LMS برای مسئله یک ابر صفحه جدا کننده نیست.

(c) با تعمیم دادن نتایج . فرض کنید که w_1 از y_1 و y_2 تشکیل شده است. و نسخه متقارنی در w_2 است. شرایط y_3 مانند جواب LMS با این نقاط جدا نمی شود.

۳۳- pseudocode را برای یک الگوریتم چندکلاسی افزایشی ثابت بر مبنای معادله ۱۱۵ بنویسید. نقاز قوت وضعف را پیدا سازی کنید.

۳٤- با تعمیم دادن بحث بخش ۵-۸-۳ را به ترتیب ، جواب مشتق شده از معادله ۱۲۰ را اثبات کنید که با تخمین مینیمم مربعات خطای میانگین با تابع تفکیک بیز داده شده در معادله ۱۲۲ ایجاد می شود.

تمرینات کامپیوتری

چندین تمرین از داده های جدول زیر استفاده می کنند :

	ω_1		ω_2		ω_3		ω_4	
sample	x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2
1	0.1	1.1	7.1	4.2	-3.0	-2.9	-2.0	-8.4
2	6.8	7.1	-1.4	-4.3	0.5	8.7	-8.9	0.2
3	-3.5	-4.1	4.5	0.0	2.9	2.1	-4.2	-7.7
4	2.0	2.7	6.3	1.6	-0.1	5.2	-8.5	-3.2
5	4.1	2.8	4.2	1.9	-4.0	2.2	-6.7	-4.0
6	3.1	5.0	1.4	-3.2	-1.3	3.7	-0.5	-9.2
7	-0.8	-1.3	2.4	-4.0	-3.4	6.2	-5.3	-6.7
8	0.9	1.2	2.5	-6.1	-4.1	3.4	-8.7	-6.4
9	5.0	6.4	8.4	3.7	-5.1	1.6	-7.1	-9.7
10	3.9	4.0	4.1	-2.2	1.9	5.1	-8.0	-6.3

⊕ بخش ۵-٤

۱- گرادیان نزولی اصلی (الگوریتم اول) را بررسی کنیدو الگو ریتم نیوتن (الگوریتم دوم) را برای داده های جداول بکار برید.

(a) دو داده ۳ بعدی را در کلاس w_1 و w_3 بکار برید. برای گرادیان نزولی ، از $\eta(k) = ۰.۱$ استفاده کنید. تابع معیار استاندارد را به عنوان تابعی از تعداد تکرار رسم کنید.

(b) تعداد کل عملیات ریاضی را در دو الگوریتم تخمین بزنید.

(c) زمان همگرایی را ئر مقابل نرخ همگرایی رسم کنید. مینیمم نرخ یادگیری که برای رد همگرایی است چیست؟

⊕ بخش ۵-۵

۲۰- یک برنامه برای پیاده سازی الگوریتم پرسپترون بنویسید.

(a) با a=۰ شروع کنید. برنامه ها را برای داده های آموزشی w_1 و w_2 بکار ببرید. به تعداد تکرار مورد نیاز برای همگرایی توجه کنید.

(b) برنامه را برای w_2 و w_3 بکار ببرید. مجددا ، تعداد تکرارهای مورد نیاز برای همگرایی را توجه کنید.

(c) اختلاف بین الزام تکرار در دو حالت را توضیح دهید.

۳- الگوریتم Pocket را با معیار استاندارد از طولانی ترین دنباله ها با بهسازی نقاط دسته بندی بکار ببرید و همبستگی الگوریتم یادگیری اصلی را را استفاده کنید. برای نمونه الگوریتم Pocket را در همبستگی با الگوریتم پرسپترون در مرتب کردن طرح چرخ دنده ای دنبال کنید. دو مجموعه از وزن ها وجود دارد.یکی الگوریتم پرسپترون نرمال ویکی دیگر جدا شدنی (به طور مستقیم برای آموزش استفاده نمی شود.) چیزی که "in your pocket". هر دو در شروع تصادفی انتخاب می شوند. وزن های pocket روی دیتا ست کامل برای پیدا کردن بیشترین اجرا از دسته بندی مناسب آزمایش می شوند. (در ابتدا این اجرا باید کوتاه باشد.) وزن های پرسپترون به طور معمول آموزش می بینند اما بعد از هر وزن بهینه شده (یا بعد از تعداد محدودی از وزن های بهینه شده) ، وزن های پرسپترون روی نقاط داده ای تست می شوند، تصادفی انتخاب می شوند وبا بیشترین اجرا از نقاط دسته بندی شده مناسب تعیین می شوند. اگر این طول ، بزرگتر از وزن های pocket باشد وزن های پرسپترون با وزن های پرسپترون جایگزین می شوندو آموزش پرسپترون پیوسته است. در این زمینه ، وزن های pocket و به طور پیوسته بهبود می یابند و دسته بندی از نقاط انتخاب شده تصادفی طولانی مدت تر می شود.

(a) یک الگوریتم pocket برای بکاربردن الگوریتم پرسپترون بنویسید.

(b)داده هایی از w_1 و w_3 را بکار ببرید. چگونه وزن های pocket بروزرسانی می شوند؟

٤- با انتخاب تصادفی a شروع کنید.β^2 را محاسبه کنید(معادله ۲۱ در انتهای آموزش γ معادله ۲۲ را محاسبه کنید. k را برای معادله ۲۵ اثبات کنید).

۵- نشان دهید که نقطه xx اول از کلاس w_x و $w_x$xx است. به صورت دستی یک نگاشت غیر خطی از فضای ویژگی را برای جداسازی خطی آنها بسازید. یک دسته بند پرسپترون روی آنها را آموزش دهید.

٦- یک نسخه از الگوریتم آموزشی Balanced Winnow را بررسی کنید(الگوریتم ۷). دسته بندی داده های تست با خط ۲ داده شده است. نرخ همگرایی Balanced Winnow را با افزایشی ثابت ، پرسپترون ساده واحد(الگوریتم ٤) روی مسئله ای با تعداد بزرگی از ویژگی های زائد مقایسه کنید . به صورت زیر دنبال کنید:

(a) یک مجموعه آموزشی ۱۰۰۰ تایی از الگو های ۱۰۰ بعدی تولید کنید که فقط در ابتدا ۱۰ ویژگی اول آموزنده هستند. برای الگو هایی در کلاس w_1 ، با هر ۱۰ ویژگی اول تصادفی انتخاب می شوند و بطور یکسان از بازه $+1 \leq x_i \leq 2$ for i = ۱,...,۱۰ . به طور معکوس برای الگو هایی در w_2 ، هر ۱۰ ویژگی به صورت تصادفی انتخاب می شوند و در بازه $-1 \leq x_i \leq -2$ یکسان هستند.همه ویژگی های دیگر از هر دو کلاس با بازه $+2 \leq x_i \leq -2$ انتخاب شده اند.

(b) به صورت دستی ابرصفحه جداسازی مشاهده شده را بسازید.

(c) تطبیق نرخ یادگیری را برای دو الگوریتم به طور تقریبی با نرخ همگرایی روی مجموعه آموزشی کامل یکسان است وقتی که فقط ۱۰ ویژگی اول بررسی می شود. فرض کنید که هر ۲۰۰۰ الگوی آموزشی فقط از۱۰ ویژگی اول هستند.

(d) حال دو الگوریتم را برای ۲۰۰۰ الگوی ۵۰ بعدی بکار ببرید. که در آن ۱۰ ویژگی اول آموزنده هستند و ٤٠ تای باقیمانده نیستند. تعداد کل خطاها را در تکرارهای مختلف بدست آورید.

(e) حال دو الگوریتم از مجموعه آموزشی کامل از ۲۰۰۰تا الگوی ۱۰۰ بعدی را بکار ببرید.

(f)جواب های بخش های (e)–(c) را خلاصه کنید.

بخش ۵-٦

روش های ملایم سازی را بررسی کنید.

(a) ملایم سازی دسته ای با مرز را پیاده سازی کنید(الگوریتم ۸) و با b=۰.۱ و a(۱)=۰ تنظیم کنید و آن را برای داده های w_1 و w_3 بکار ببرید . تابع معیار استاندارد را به عنوان یک تابع با تعداد مجموعه های آموزشی مجهول رسم کنید.

(b) با b=۰.۵ و a(۱)=۰ تکرار کنید. از لحاظ کیفی هر اختلاف که شما در نرخ همگرایی پیدا می کنید را توضیح دهید.

(c) برنامه را با استفاده از یادگیری ساده واحد تغییر دهید. مجددا یک معیار استاندارد به عنوان یک تابع از تعدادی از مجهولات از نمونه های آموزشی را رسم کنید.

(d) هر اختلافی را توضیح دهید.مطمئن باشید که نرخ یادگیری را بررسی کرده اید.

بخش ۵-۸

۸- یک الگوریتم ملایم سازی ساده واحد بنویسید و از چه معادله باید برای بروزرسانی R استفاده می کنید. برنامه را برای داده های w_2 و w_3 بکار ببرید.

بخش ۵-۹

۹- الگوریتم Ho-Kashyap را پیاده سازی کنید(الگوریتم ۱۱). و داده ها در کلاس w_1 و w_3 را بکار ببرید. برای کلاس های w_2 و w_4 را تکرار کنید.

بخش ۵-۱۰

۱۰- مثال بزنید جایی که قانون LMS بردارهای جداگانه لازم نیست حتی اگر یکی موجود باشد.

بخش ۵-۱۱

۱۱- ماشین بردار پشتیبان xxx را داریم. آن را برای دسته بندی w_3 و w_4 بکار ببرید.

PDF-XChange
Click to buy NOW!
www.docu-track.com

بخش ۵-۱۲

۱۲- یک برنامه پیاده سازی تعمیم چندکلاسی از ملایم سازی نمونه های واحد بدون مرز بنویسید.

(a) آن را برای داده های هر ٤ کلاس در جدول بکار بگیرید.

(b) الگوریتم دو کلاسی ساخته شده به شکل مرزهای $w_i/notw_i$ را برای i= ۱,۲,۳,٤ استفاده کنید. یک ناحیه دسته بندی با سیستم های چند رده (مبهم) پیدا کنید.