مقایسه روش‌های رگرسیونی کلاسیک با شبکه عصبی و ماشین بردار پشتیبان در رده‌بندی منابع آب‌های زیرزمینی

اکرم حیدری[۱]، مهرداد نیاپرست[۲]

تاریخ دریافت: ۹۸/۱/۲۷
تاریخ پذیرش: ۹۹/۳/۱

چکیده:
در عصر حاضر رده‌بندی داده‌ها به منظور تشخیص و پیش‌بینی وقایع، یکی از موضوعات بسیار مهم در علوم مختلف است. در دیدگاه سنتی علم آمار این رده‌بندی‌ها بر اساس روش‌های کلاسیک و بر پایه مدل‌های آماری از جمله رگرسیون لوژستیک امکان‌پذیر بود. در عصر حاضر که به عبارتی عصر انفجار اطلاعات نامیده می‌شود، در اکثر موارد با داده‌هایی مواجه هستیم که نمی‌توان توزیع دقیقی را برای آن‌ها یافت؛ ازاین‌رو استفاده از روش‌های داده کاوی و یادگیری ماشین که به مدل‌های از پیش تعیین‌شده نیاز ندارند، می‌تواند مثمر ثمر باشد. در بسیاری از کشورها تشخیص دقیق نوع منابع آب‌های زیرزمینی، یکی از مسائل قابل‌توجه در زمینه علوم آب است. در این مقاله به مقایسه نتایج حاصل از رده‌بندی یک مجموعه داده مربوط به منابع آب‌های زیرزمینی با استفاده از روش‌های رگرسیونی، شبکه عصبی و ماشین بردار پشتیبان پرداخته‌ایم. نتایج این رده‌بندی‌ها نشان داد که روش‌های یادگیری ماشین در تشخیص دقیق نوع چشمه‌ها مؤثر بوده است.

واژه‌های کلیدی: شبکه عصبی [۳]، ماشین بردار پشتیبان [۴]، رگرسیون لوژستیک

۱ مقدمه

رگرسیون لوژستیک به عنوان یک ابزار رده‌بندی برای حالت‌هایی که متغیر پاسخ دو سطحی است، استفاده می‌گردد. تعمیم رگرسیون لوژستیک به حالت چند سطحی توسط گارلند و همکاران [۳]، منتل [۶] و تیل [۱۱] صورت گرفته است.

شبکه‌های عصبی و ماشین بردار پشتیبان نیز ابزارهای رده‌بندی برای مواردی هستند که توزیع دقیق داده‌ها مشخص نبوده و با یک مسئله ناپارامتری مواجه هستیم. الگوریتم شبکه عصبی نخستین بار توسط روزنبلات [۹] به صورت یک شبکه عصبی تک لایه با یک لایه پنهان که پرسپترون نامیده شد، مطرح گردید و الگوریتم SVM نیز نخستین بار توسط واپنیک و چروننکیس [۵] بیان شد. [۲]

۲ رگرسیون دنباله‌رو تصویرساز

برای بیان شبکه‌های عصبی لازم است ابتدا با مبحثی تحت عنوان "رگرسیون دنباله‌رو تصویرساز" [۶] (PPR) آشنا شویم. به این منظور مطابق با شکل ۲ فرض کنید یک بردار ورودی X با p مؤلفه، یک بردار هدف y و نیز ω_m ، $m = ۱,\ldots,M$ واحد p مؤلفه‌ای از پارامترهای نامعلوم باشند. در این صورت مدل PPR به صورت زیر است؛

$$f(X) = \sum_{m=۱}^{M} g_m(\omega_m^T X) \qquad (۱)$$

در این مدل جمعی، به جای ورودی‌ها، ویژگی‌های استخراج‌شده $V_m = \omega_m^T X$ را قرار می‌دهیم و توابع g_m نیز نامشخص هستند و توسط برخی از روش‌های هموارسازی برآورد می‌شوند. متغیر عددی $V_m = \omega_m^T X$، تصویر X را بر بردار واحد ω_m تصویر می‌کند و تابع $g_m(\omega_m^T X)$ یک تابع ستیغی [۷] در $\mathbf{R^p}$ نامیده می‌شود. این تابع تنها در مسیر ω_m تعریف‌شده است به همین

[۱] کارشناسی ارشد آمار ریاضی، دانشگاه رازی، دانشکده علوم، گروه آمار
[۲] گروه آمار، دانشگاه رازی، دانشکده علوم، کرمانشاه، ایران

[3] Neural Networks
[4] Support Vector Machine
[5] Vapnik and Chervonenkis
[6] Projection Pursuit Regression
[7] Ridge function

دلیل این مدل را مدل رگرسیون دنباله‌رو تصویرساز می‌نامند.

برای ارزیابی مدل برازش داده‌شده، با توجه به این‌که شیوه آموزش در این روش یادگیری نظارتی [۸] است، با در نظر گرفتن داده آموزشی $(x_i,y_i), i = ۱,\ldots,N$ بر روی توابع g_m و بردارهای جهت ω_m ، تابع خطای زیر را در نظر گرفته و آن را برآورد می‌کنیم؛

$$\sum_{i=۱}^{N}[y_i - \sum_{m=۱}^{M} g_m(\omega_m^T X)] \qquad (۲)$$

برای برآورد معادله فوق فرض می‌کنیم $M = ۱$ است. در این صورت باید آن را بر اساس ω، کمینه کنیم.

برای انجام این کار یک روش گاوس-نیوتون [۹]ابداع شد که روشی شبه-نیوتون [۱۰]است و هر بخش از مسیر شامل مشتقات دوم g است. در این صورت اگر ω_{old} برآورد حاضر برای ω باشد داریم؛

$$g(\omega^T x_i) \approx g(\omega_{old}^T x_i) + g'(\omega_{old}^T x_i)(\omega - \omega_{old})^T x_i \qquad (۳)$$

در این صورت می‌توان مقدار خطا را به صورت زیر بر اساس ω برآورد کرد؛

$$\sum_{i=۱}^{N} [y_i - g(\omega^T x_i)]^۲ \approx$$
$$\sum_{i=۱}^{N} g'(\omega_{old}^T x_i)^۲ [(\omega_{old}^T x_i + \frac{y_i - g(\omega_{old}^T x_i)}{g'(\omega_{old}^T x_i)}) - \omega^T x_i]^۲ \qquad (۴)$$

این عمل بردار ضرایب به‌روزرسانی شده ω_{new} را به وجود می‌آورد و برآورد g و ω تا زمانی که مسئله همگرا شود ادامه می‌یابد و در صورتی که در مدل PPR ، $M > ۱$ باشد، مدل در یک رویکرد روبه‌جلو ساخته می‌شود که در هر مرحله جفت (ω_m, g_m) اضافه می‌گردد. در واقع می‌توان گفت که تعداد شروط M به عنوان بخشی از استراتژی روبه‌جلو به کار می‌رود و ساخت مدل زمانی متوقف می‌گردد که شرط بعدی موجب بهبود برازش مدل نگردد. به عبارت دیگر، استراتژی روبه‌جلو زمانی متوقف می‌شود که اجرای شرط بعدی که همان جفت (ω_m, g_m) است، نتواند مدلی بهتر از مدل به‌دست‌آمده کنونی را به داده‌های موجود برازش دهد. [۵]

۳ شبکه عصبی

در این بخش نحوه عملکرد شبکه عصبی وانیلا [۱۱] که شبکه پس‌رو با یک لایه پنهان و یا پرسپترون تک لایه نامیده می‌شود و نیز پرسپترون چند لایه (MLP) [۱۲] را موردبحث قرار می‌دهیم. شبکه‌های عصبی در واقع مدل‌های آماری غیرخطی هستند که بسیار شبیه به مدل‌های PPR عمل می‌کنند. شبکه‌های عصبی در هر دوشاخه رگرسیون و رده‌بندی به کار می‌روند. برای رده‌بندی K رده‌ای، در نهایت k واحد وجود دارد طوری که k امین واحد، احتمال رده k ام را بیان می‌کند. هم‌چنین K معیار هدف $Y_k, k = ۱,\ldots,K$ وجود دارند که هر یک به صورت دو مؤلفه ۰ و ۱، برای رده k ام کد دهی شده‌اند به نحوی که مؤلفه ۱ به معنای وقوع رویداد خاص و ۰ به معنای عدم وقوع رویداد است. ویژگی‌های استخراج‌شده Z_m از ترکیبات خطی ورودی‌ها ساخته‌شده‌اند و هدف Y_k به عنوان تابعی از ترکیبات خطی Z_m مدل‌بندی شده است؛

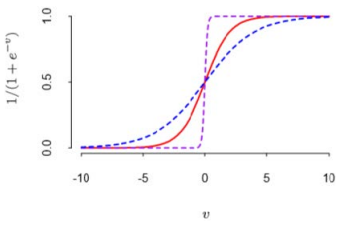
$$Z_m = \sigma(\alpha_{۰m} + \alpha_m^T X), m = ۱,\ldots,M$$

$$T_k = \beta_{۰k} + \beta_k^T Z, k = ۱,\ldots,K \qquad (۵)$$

$$f_k(X) = g_k(T), k = ۱,\ldots,K$$

که در آن $T = (T_۱,\ldots,T_K)$ و $Z = Z_۱,\ldots,Z_M$ توجه کنید که در اینجا ضرایب α مشابه با ضرایب ω در مدل PPR است.

تابع فعال‌سازی $\sigma(v)$ اغلب اوقات سیگموئید انتخاب می‌شود که به صورت $\sigma(v) = \frac{۱}{۱+\exp^{-v}}$ است. البته گاهی تابع پایه‌ای شعاعی گاوسی را برای $\sigma(v)$ در نظر می‌گیرند که در این صورت آن را شبکه پایه‌ای شعاعی می‌نامند. اما در لایه‌های پنهان شبکه‌های عصبی که مقادیر Z_m در آن محاسبه می‌شود، اغلب از توابع سیگموئید استفاده می‌کنند. شکل ۱ نمایی از تابع سیگموئید $\sigma(sv)$ را نشان می‌دهد که در آن پارامتر s ، نرخ فعال‌سازی را کنترل می‌کند. $s = \frac{۱}{۲}$ نمودار آبی رنگ و $s = ۱۰$ نمودار بنفش رنگ است. ملاحظه می‌کنید که s بزرگ در $v = ۰$ فعال‌سازی آن واحد(نورون-گره) را سخت می‌کند و $\sigma(s(v - v_۰))$ آستانه فعال‌سازی را از صفر به $v_۰$ انتقال می‌دهد. [۵]

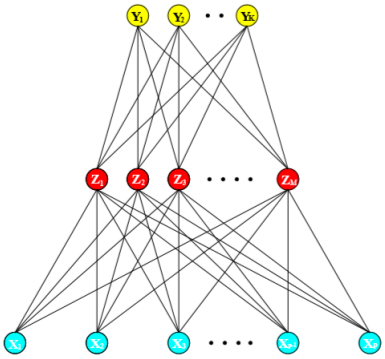


شکل ۱. نمودار تابع سیگموئید(نمودار وسط)

شکل ۲ شمایی از یک شبکه عصبی پیشرو با یک لایه پنهان را نشان می‌دهد. یک ویژگی ورودی به نام اندازه اریبی [۱۳] وجود دارد که به هر یک از

[8] Supervised learning
[9] Gauss-Newton
[10] Quasi-Newton
[11] Vanilla
[12] Multi Layer Perceptron
[13] Bias

واحدهای موجود در لایه‌های خروجی و پنهان اضافه می‌شود و می‌توان آن را برابر با ثابت «۱» در نظر گرفت. این واحد اریبی در واقع عرض از مبدأهای $\alpha_{\circ m}$ و $\beta_{\circ m}$ موجود در معادله (۵) را می‌گیرد.



شکل۲. شمایی از یک شبکه عصبی پیشرو با یک لایه پنهان

برای انجام محاسبات لازم برای Z_m ها به توابعی به نام توابع تشخیص [۱۴] نیاز است. این تابع در شبکه عصبی همان تابع سیگموئید است. مطابق با شکل ۱ و نیز معادله (۵) ملاحظه می‌کنید که نرخ فعال‌سازی سیگموئید به نرم α_m بستگی دارد به‌گونه‌ای که اگر $||\alpha_m||$ خیلی کوچک باشد، بخش خطی تابع فعال‌سازی را فراهم می‌آورد. با توجه به این مطالب می‌توان گفت که شبکه عصبی بسیار شبیه به مدل PPR است و تفاوت آن‌ها در این است که مدل PPR از توابع ناپارامتری $g_m(v)$ استفاده می‌کند درحالی‌که شبکه عصبی از یک تابع بسیار ساده‌تر بر اساس $\sigma(v)$ با سه پارامتر آزاد نرخ فعال‌سازی، α و β استفاده می‌کند. در این صورت اگر از دیدگاه مدل‌های PPR به شبکه عصبی نگاه کنیم داریم؛

$$g_m(\omega_m^T X) = \beta_m \sigma(\alpha_{\circ m} + \alpha_m^T X) = \beta_m \sigma(\alpha_{\circ m} + ||\alpha_m||(\omega_m^T X)) \qquad (۶)$$

طوری که $\omega_m = \frac{\alpha_m}{||\alpha_m||}$ ، m امین بردار واحد است. در واقع مزیت شبکه عصبی بر مدل PPR را می‌توان این‌گونه بیان نمود:

با توجه به این‌که $g(v)$ نسبت به یک $\sigma_{\beta,\alpha_\circ,s(v)} = \beta\sigma(\alpha_\circ + sv)$ ناپارامتری، پیچیدگی کم‌تری دارد، لذا شبکه عصبی می‌تواند برای تحلیل مدل‌هایی با تعداد ورودی‌های بسیار بالا، حتی تا ۱۰۰ متغیر نیز مورداستفاده قرار می‌گیرد. این در حالی است که مدل PPR از قیود کم‌تری استفاده می‌کند به‌عنوان‌مثال $M = ۵$ یا ۱۰. [۵]

۴ برازش دادن شبکه‌های عصبی

شبکه‌های عصبی این امکان را برای کامپیوترها فراهم می‌آورند که به اطلاعات در مقیاس بزرگ دست یابند و نیز برای ساخت مدل‌های بسیار پیچیده به کار می‌روند. در واقع می‌توان گفت شبکه‌های عصبی «نبض» اصلی الگوریتم‌های یادگیری ماشین است. [۷] شبکه عصبی پارامترهای نامعلومی به نام وزن دارد که باید مقادیری را برای آن‌ها در نظر گرفت که برازش مدل به داده‌های آموزشی را به خوبی انجام دهد. می‌توان یک مجموعه کامل از وزن‌ها را با θ نمایش داد که شامل مقادیر زیر است؛

$$\{\alpha_{\circ m}, \alpha_m;\ m = ۱,\dots,M\}\ ,\ \{\beta_{\circ k}, \beta_k;\ k = ۱,\dots,K\} \qquad (۷)$$

در این صورت برای رده‌بندی داده‌ها از یک خطای درجه دوم به نام آنتروپی-متقابل [۱۵] استفاده می‌کنیم؛ [۵]

$$R(\theta) = -\sum_{i=۱}^{N}\sum_{k=۱}^{K} y_{ik}\log f_k(x_i) \qquad (۸)$$

در این صورت جداکننده متعلق به آن برابر است با؛

$$G(x) = \arg\max_k f_k(x)$$

با در نظر گرفتن تابع فعال‌سازی و تابع خطای آنتروپی-متقابل، مدل شبکه عصبی در واقع یک مدل رگرسیون لوژستیک خطی در واحدهای پنهان است که تمام پارامترهای آن با روش درستنمایی ماکسیمم برآورد می‌شوند. حال می‌توان گفت که یک روش عمومی برای مینیمم کردن $R(\theta)$ از طریق کاهش گرادیان [۱۶] است که تنظیمات انتشار پس‌رو [۱۷] نامیده می‌شود. فرض کنید $\sigma(\alpha_{\circ m} + \alpha_m^T x_i)$ ، از معادله (۵) و نیز با فرض $z_i = (z_{۱i},\dots,z_{Mi})$ داریم؛ [۵]

$$R(\theta) = \sum_{i=۱}^{N} R_i = \sum_{i=۱}^{N}\sum_{k=۱}^{K}(y_{ik} - f_k(x_i))^۲ \qquad (۹)$$

$$\frac{\partial R_i}{\partial \beta_{km}} = -۲(y_{ik} - f_k(x_i))g_k'(\beta_k^T z_i)z_{mi},$$

$$\frac{\partial R_i}{\partial \alpha_{ml}} = -\sum_{k=۱}^{K} ۲(y_{ik} - f_k(x_i))g_k'(\beta_k^T z_i)\beta_{km}\sigma'(\alpha_m^T x_i)x_{il}. \qquad (۱۰)$$

با استفاده از این مشتقات ، یک بروز رسانی کاهش گرادیان در $(r+۱)$ امین تکرار برابر است با؛

$$\beta_{km}^{(r+۱)} = \beta_{km}^{(r)} - \gamma_r \sum_{i=۱}^{N} \frac{\partial R_i}{\partial \beta_{km}^{(r)}},$$

$$\alpha_{ml}^{(r+۱)} = \alpha_{ml}^{(r)} - \sum_{i=۱}^{N} \frac{\partial R_i}{\partial \alpha_{ml}^{(r)}}. \qquad (۱۱)$$

[14] Identity functions
[15] Cross-Entropy
[16] Gradian decay
[17] Back-propagation

طوری که γ_r نرخ یادگیری نامیده می‌شود. می‌توان معادلات (۱۰) را به صورت زیر بازنویسی نمود؛

$$\frac{\partial R_i}{\partial \beta_{km}} = \delta_{ki} z_{mi}$$

$$\frac{\partial R_i}{\partial \alpha_{ml}} = s_{mi} x_{il} \qquad (۱۲)$$

طوری که δ_{ki} و s_{mi} «خطاهای» مدل اصلاح‌شده به ترتیب در واحدهای خروجی و پنهان هستند و داریم؛

$$s_{mi} = \sigma'(\alpha_m^T x_i) \sum_{k=1}^{K} \beta_{km} \delta_{ki} \qquad (۱۳)$$

این معادلات را معادلات انتشار پس‌رو می‌نامند و با استفاده از به‌روزرسانی‌ها در معادلات (۱۱) یک الگوریتم دو گذرگاهی [18] با دو گذرگاه پیشرو [19] و پس‌رو [20] ایجاد می‌گردد به نحوی که در گذرگاه پیشرو، وزن‌های اصلاح‌شده ثابت‌شده‌اند و مقادیر برآورد شده $\hat{f}_k(x_i)$ از معادله (۵) به دست می‌آید و نیز در گذرگاه پس‌رو، خطاهای δ_{ki} محاسبه می‌شوند. انتشار پس‌رو از طریق معادله (۱۳) خطاهای s_{mi} را به دست می‌آورد، سپس هر دو مجموعه خطا را برای محاسبه گرادیان‌ها برای به‌روزرسانی‌ها در معادلات (۱۱) از طریق معادلات (۱۲) مورداستفاده قرار می‌گیرند. این روش در واقع همان روش انتشار پس‌رو است که قانون دلتا نیز نامیده می‌شود. به‌روزرسانی‌ها در معادلات (۱۱) نوعی دیگر از یادگیری دسته‌ای [21] است. اغلب نرخ یادگیری γ_r برای یادگیری دسته‌ای را یک مقدار ثابت در نظر می‌گیریم طوری که در هر مرحله، تابع خطا را مینیمم کند، به صورتی که هرگاه داشته باشیم $r \to \infty$ آنگاه $\gamma_r \to 0$. بنابراین همگرایی در صورتی حتمی خواهد بود که

$$\gamma_r \to 0 \ , \ \sum_r r\gamma_r = \infty \ , \ \sum_r \gamma_r^2 < \infty$$

(به‌عنوان‌مثال برای $\gamma_r = \frac{1}{r}$ صدق می‌کند). [۵]

نکته‌ای که باید به آن توجه داشت این است که اگر وزن‌ها نزدیک به صفر باشند نمودار سیگموئید موجود در شکل ۱ تقریباً خطی است، در نتیجه شبکه عصبی به مدل خطی نزدیک می‌شود. معمولاً مقادیر شروع‌کننده وزن‌ها برای لایه اول، مقادیر تصادفی نزدیک به صفر انتخاب می‌شوند؛ به همین دلیل هم مدل با شکلی نزدیک به خطی شروع می‌شود و با افزایش وزن‌ها غیرخطی می‌شود. استفاده از وزن‌های صفر واقعی منجر به مشتقات صفر شده و الگوریتم هرگز حرکت نمی‌کند، [۵] زیرا علت اصلی فعال شدن شبکه در واقع فعال بودن اتصالات بین نورون‌های موجود در دو لایه مختلف است که این اتصالات، همان وزن‌ها هستند و توابع فعال‌سازی موجود در واحدهای درون هر لایه، جمع وزن دهی شده به‌دست‌آمده از نورون‌های لایه قبلی را محاسبه کرده و در صورتی که مقدار آن از یک حد آستانه معین بیش‌تر شود، آن واحد(نورون) فعال خواهد شد. [۷]

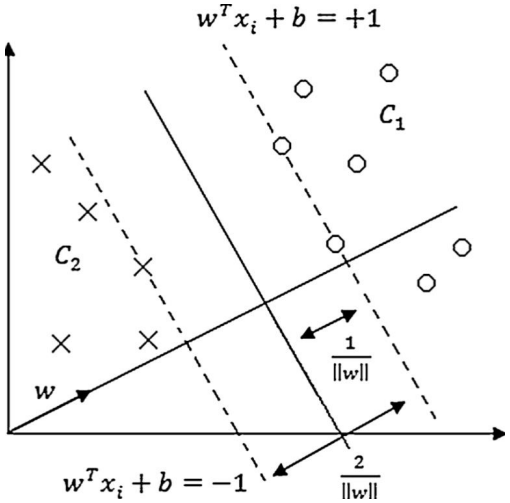
۵ ماشین بردار پشتیبان

یکی از شاخه‌های یادگیری ماشین، SVM است و برای رده‌بندی داده‌ها به کار می‌رود که بنا بر تعداد سطوح متغیر پاسخ به دو حالت دو رده‌ای و چند رده‌ای تقسیم می‌گردد. فرآیند جداسازی رده‌ها توسط توابعی به نام تابع تصمیم انجام می‌شود. اگر داده‌ها در فضای ورودی اولیه به صورت خطی جداپذیر باشند تابع تصمیم را خطی خوانده و SVM را سخت-حاشیه می‌نامند. تابع تصمیم در حالت دو رده‌ای را به صورت زیر نمایش می‌دهند

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \quad , \quad i = 1, 2, \ldots, N$$

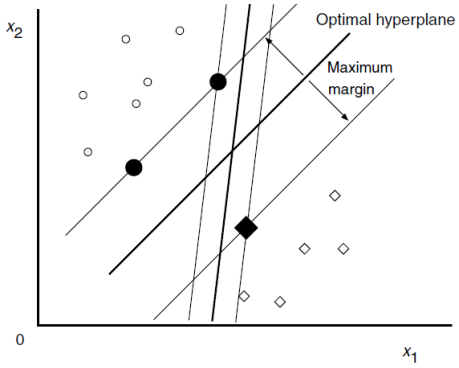
به طوری که $\mathbf{w}$ یک بردار p بعدی که p بیان‌گر بعد داده مشاهده‌شده $\mathbf{x}$ (داده مشاهده‌شده i ام را با $\mathbf{x}_i$ نمایش می‌دهیم) و b اندازه اریبی و N تعداد داده‌های آموزشی است. در این صورت تخصیص داده $\mathbf{x}$ به هر یک از دو رده به این صورت انجام می‌شود؛

$$D(\mathbf{x}) \begin{cases} > 0 & \mathbf{x} \text{ متعلق به رده یک} \\ < 0 & \mathbf{x} \text{ متعلق به رده دو} \end{cases}$$



شکل۳. SVM جداپذیر خطی

[18] Two-pass algorithm
[19] Forward-pass
[20] Backward-pass
[21] Batch learning



شکل ۴. ابرصفحه جداکننده بهینه در حالت دوبعدی

در شکل ۳ نمایی از جداپذیری خطی در حالت دو رده‌ای را مشاهده می‌کنیم طوری که خط‌چین‌ها ابرصفحه‌های مثبت و منفی هستند. [۱] مطابق با شکل ۴ انتخاب این تابع تصمیم برای جداسازی دو رده باید به‌گونه‌ای انجام شود که بیشینه حاشیه ممکن (h) بین نزدیک‌ترین داده‌های مربوط به دو رده مختلف ایجاد گردد. اگر $\mathbf{x}^+$ و $\mathbf{x}^-$ به ترتیب داده‌هایی بر دو ابرصفحه مثبت و منفی باشند، حاشیه بین دو ابرصفحه برابر

$$h = |\mathbf{x}^+ - \mathbf{x}^-| = \frac{۲}{||\mathbf{w}||}$$

خواهد بود. در این صورت مسئله بهینه‌سازی محدب زیر را داریم

$$minimize \quad Q(\mathbf{w},b,\alpha) = \frac{۱}{۲}\mathbf{w}^T\mathbf{w} - \sum_{i=۱}^{N}\alpha_i(y_i(\mathbf{w}^T\mathbf{x}_i + b) - ۱) \qquad (۱۴)$$

به طوری که

$$(\alpha_i \geq ۰) \quad , \quad \alpha = (\alpha_۱, \alpha_۲, \ldots, \alpha_M)^T$$

که در آن α_i ها ضرایب لاگرانژ نامنفی مربوط به $\mathbf{x}_i$ ها هستند. تحت شرایط "کاروش-کان-توکر" [۲۲] زیر داریم

$$\frac{\partial Q(\mathbf{w},b,\alpha)}{\partial \mathbf{w}} = \mathbf{w} - \sum_i \alpha_i y_i \mathbf{x}_i = ۰ \quad \Rightarrow \quad \mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$
$$\frac{\partial Q(\mathbf{w},b,\alpha)}{\partial \mathbf{b}} = -\sum_i \alpha_i y_i = ۰ \quad \Rightarrow \quad \sum_i \alpha_i y_i = ۰$$

$\mathbf{x}_i$ هایی که α_i مثبت دارند، بردارهای پشتیبان مسئله را تشکیل می‌دهند و روی ابرصفحه‌ها قرار دارند بنابراین با جایگذاری مقادیر فوق در معادله (۱۴) خواهیم داشت

$$maximize \quad Q(\alpha) = \sum_{i=۱}^{N}\alpha_i - \frac{۱}{۲}\sum_{i=۱}^{N}\sum_{\substack{j=۱ \\ i\neq j}}^{N}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j$$

به طوری که

$$\sum_{i=۱}^{N}\alpha_i y_i = ۰ \quad , \quad \alpha_i \geq ۰ \quad i = ۱,۲,\ldots,N$$

اما گاهی داده‌ها در فضای ورودی به صورت خطی قابل جداسازی نیستند. در این شرایط با دو حالت کلی مواجه هستیم

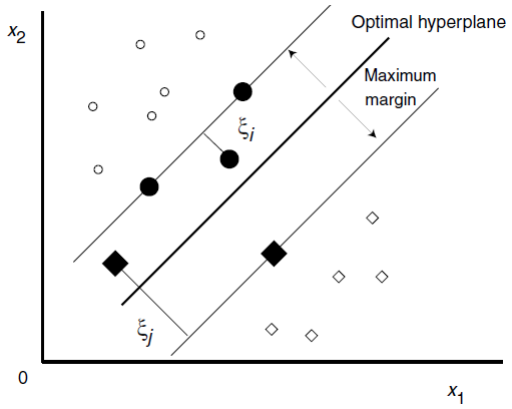
حالت اول، در این حالت با افزودن متغیر ضعف [۲۳] (ξ_i) به تابع تصمیم، به صورت

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq ۱ - \xi_i \quad , \quad i = ۱,۲,\ldots,N$$

مسئله مینیمم سازی (۱۵) را داریم و SVM را نرم-حاشیه می‌نامند. متغیر ضعف برای داده‌هایی که درون حاشیه بین دو ابرصفحه قرار می‌گیرند به کار می‌رود. زیرا برای سایر داده‌ها که به درستی رده‌بندی شده‌اند، $\xi_i = ۰$ است. با توجه به این‌که مرز تصمیم دارای معادله $D(\mathbf{x}) = ۰$ و ابرصفحه مثبت دارای معادله $D(\mathbf{x}) = +۱$ و نیز ابرصفحه منفی معادله $D(\mathbf{x}) = -۱$ را دارا است، لذا مطابق با شکل ۵ می‌توان برای داده آموزشی $\mathbf{x}_i$ دو حالت زیر را در نظر گرفت

۱- اگر $۰ < \xi_i < ۱$ باشد، داده $\mathbf{x}_i$، ماکسیمم حاشیه را ندارد اما همچنان به درستی قابل رده‌بندی است،

۲- اگر $\xi_i > ۱$ باشد،(مانند $\xi_j > ۱$)، در این صورت باید گفت داده ورودی به وسیله ابرصفحه بهینه به درستی رده‌بندی نشده است. [۱]



شکل۵. استفاده از متغیر ضعف در رده‌بندی داده‌ها در حالت دو رده‌ای

$$minimize\, Q(\mathbf{w},b,\xi,\alpha,\beta) = \frac{۱}{۲}||\mathbf{w}||^۲ + C\sum_{i=۱}^{N}\xi_i$$
$$-\sum_{i=۱}^{N}\alpha_i(y_i(\mathbf{w}^T\mathbf{x}_i + b) - ۱ + \xi_i) - \sum_{i=۱}^{N}\beta_i\xi_i \qquad (۱۵)$$

[22] Karush-Kuhn-Tucker(KKT)

[23] Slack Variable

در این حالت نیز با استفاده از شرایط KKT داریم

$$\frac{\partial Q(\mathbf{w},b,\xi,\alpha,\beta)}{\partial \mathbf{w}} = \mathbf{0} \Rightarrow \mathbf{w} - \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i = \mathbf{0} \Rightarrow \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$$

$$\frac{\partial Q(\mathbf{w},b,\xi,\alpha,\beta)}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0$$

$$\frac{\partial Q(\mathbf{w},b,\xi,\alpha,\beta)}{\partial \xi} = \mathbf{0} \Rightarrow C - \alpha_i - \beta_i = \mathbf{0} \Rightarrow \alpha_i + \beta_i = C$$

به طوری که C را پارامتر هزینه خوانده و رابطه بین ماکسیمم سازی حاشیه بین دو رده و مینیمم سازی خطای رده‌بندی را ارزیابی می کند. با جایگذاری مقادیر فوق در معادله (۱۵) داریم

$$maximize \quad Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{\substack{i=1 \\ i \neq j}}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j$$

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \quad \text{و} \quad 0 \leq \alpha_i \leq C \quad i = (1,2,\ldots,N)$$

در این حالت نیز که تنها تفاوت آن با SVM سخت–حاشیه در کران بالای α_i است، تابع تصمیم دقیقاً برابر با این تابع در حالت سخت–حاشیه است

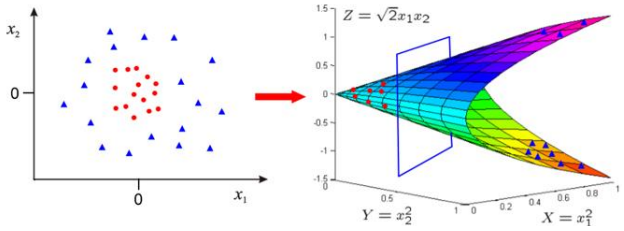
$$D(\mathbf{x}) = \sum_{i \in S} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b$$

که در آن S مجموعه‌ای از $\mathbf{x}_i$ ها با ضرایب لاگرانژ نامنفی است و برای رده‌بندی مشاهده $\mathbf{x}$ داریم

$$D(\mathbf{x}) \begin{cases} > 0 & \mathbf{x} \text{ متعلق به رده یک} \\ < 0 & \mathbf{x} \text{ متعلق به رده دو} \end{cases}$$

حالت دوم، مربوط به زمانی است که حتی با اضافه نمودن متغیر ضعف به مسئله باز هم داده‌ها به صورت خطی قابل رده‌بندی نیستند؛ در این حالت می‌توان داده‌ها را توسط توابعی به نام کرنل که خود دارای انواع مختلفی است به فضایی با ابعاد $l > p$ به نام فضای مشخصه انتقال داد طوری که داده‌ها در این فضای جدید، جداپذیر خواهند بود. این عمل را حیله کرنل می‌نامند.

$$\Phi : \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow \begin{pmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2}x_1 x_2 \end{pmatrix} \quad \mathbb{R}^2 \rightarrow \mathbb{R}^3$$



شکل ۶. نمایی از انتقال داده‌ها به فضای مشخصه

شکل ۶ نمایی از انتقال داده‌ها توسط کرنل چندجمله‌ای، از فضای دوبعدی به فضای مشخصه سه‌بعدی و جداسازی خطی داده‌ها در این فضا را نمایش می‌دهد.

این ایده به بیش از دو رده نیز قابل‌تعمیم است. برای جداسازی رده‌ها در این حالت می‌توان از چهار ماشین بردار پشتیبان متفاوت تحت عناوین یک در مقابل همه (OAA)، زوجی $(PSVM)$ ، اصلاح خطای کدهای خروجی $(ECOC)$ و همه در یک‌زمان بهره گرفت. [۵]

۶ رگرسیون لوژستیک

با توجه به این‌که متغیر پاسخ کیفی و دارای چند سطح است لذا این مدل به منظور تخمین احتمال وقوع یک رویداد خاص(احتمال تعلق بردار داده $\mathbf{x}$ به رده i $(i = 1,\ldots,n)$ ، به کار می‌رود، به طوری که n تعداد رده‌ها است. برای تخصیص داده $\mathbf{x}$ به یکی از دو رده i و j از بین n رده موجود، از شاخصی تحت عنوان نسبت بخت‌ها (OR) به صورت زیر استفاده می‌شود

$$OR = \frac{\frac{\pi_i}{1-\pi_i}}{\frac{\pi_j}{1-\pi_j}}$$

به طوری که $\pi_i = Pr(G = i|\mathbf{X} = \mathbf{x})$ و $\pi_j = Pr(G = j|\mathbf{X} = \mathbf{x})$ ، $(i,j = 1,\ldots,n; i \neq j)$. در این صورت π_i را احتمال تعلق بردار داده $\mathbf{x}$ به رده i می‌نامند. بر اساس مقدار حاصل‌شده OR، $\mathbf{x}$ را به یکی از دو رده i و j نسبت می‌دهند طوری که اگر $OR > 1$ باشد آنگاه $\mathbf{x}$ عضو رده i، اگر $OR < 1$ باشد آنگاه $\mathbf{x}$ متعلق به رده j و اگر $OR = 1$ باشد، $\mathbf{x}$ غیرقابل رده‌بندی است. با در نظر گرفتن تابع ربط کانونی لوجیت داریم

$$\log \frac{\pi_i}{\pi_n} = \frac{Pr(G = i|\mathbf{X} = \mathbf{x})}{Pr(G = n|\mathbf{X} = \mathbf{x})} = \beta_{i0} + \beta_i \mathbf{x} \; ; \; i = 1,\ldots,n-1$$

توجه شود که مقدار فوق در واقع همان آماره OR برای دو رده است و رده مخرج نیز قراردادی است. [۵]

۷ بحث و نتیجه‌گیری

در این پژوهش داده‌های مربوط به ۴۲ منبع آب زیرزمینی، به منظور بررسی و مقایسه رده‌بندی چشمه‌ها به دو دسته تحول‌یافته (A) و تحول

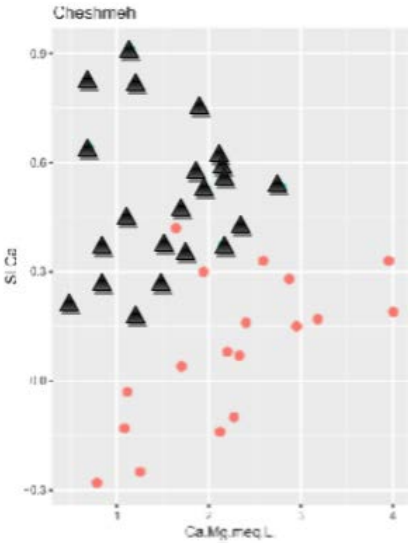
جدول ۱: رده‌بندی مجموعه داده منابع آب با استفاده از روش SVM

هسته	$cost$	$gamma$	$degree$	$Coef_0$	SV	تعداد رده‌بندی اشتباه	دقت
radial	۱	۰٫۲۵	_	_	۱۸	۰	۱
linear	۱	۰٫۲۵	_	_	۱۱	۲	۰٫۹۵
polynomial	۱	۰٫۲۵	۳	۰	۱۹	۶	۰٫۸۶
sigmoid	۱	۰٫۲۵	_	۰	۱۵	۲	۰٫۹۵

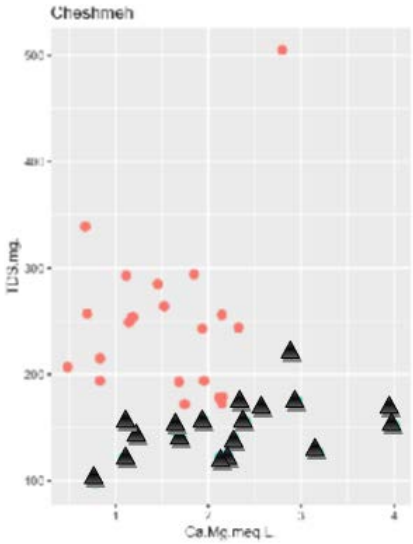
جدول ۲: ضرایب رگرسیونی برای دو مدل لجیت و پروبیت

	لوجیت			پروبیت		
متغیر مستقل	$Estimate$	$ErrorStd.$	$p-value$	$Estimate$	$ErrorStd.$	$p-value$
Intercept	$۱٫۷۲۷e+۱۵$	$۶٫۶۵۶e+۰۷$	$<۲e-۱۶$	$۵٫۵۶۲e+۱۵$	$۶٫۶۵۶e+۰۷$	$<۲e-۱۶$
Ca/Mg	$۵٫۹۹۱e+۱۴$	$۱٫۳۸۲e+۰۷$	$<۲e-۱۶$	$۱٫۶۴۵e+۱۵$	$۱٫۳۸۲e+۰۷$	$<۲e-۱۶$
SI.Ca	$-۱٫۲۴۸e+۱۵$	$۵٫۶۳۶e+۰۷$	$<۲e-۱۶$	$-۳٫۷۸۵e+۱۵$	$۵٫۶۳۶e+۰۷$	$<۲e-۱۶$
TDS	$-۸٫۹۷۸e+۱۲$	$۱٫۸۴۶e+۰۵$	$<۲e-۱۶$	$-۲٫۴۰۱e+۱۳$	$۱٫۸۴۶e+۰۵$	$<۲e-۱۶$
EC	$-۲٫۲۳۴e+۱۲$	$۱٫۱۵۴e+۰۵$	$<۲e-۱۶$	$-۷٫۵۲۱e+۱۲$	$۱٫۱۵۴e+۰۵$	$<۲e-۱۶$
AIC	۸۲٫۰۸۷			۱۵۴٫۱۷۰		
accuracy	۰٫۹۷۶۱۹۰۵			۰٫۹۵۲۳۸۱		

نیافته (B) با استفاده از روش رگرسیون لوژستیک و دو روش ماشین بردار پشتیبان و شبکه عصبی به‌کاررفته است. در ادامه جدول‌ها و نتایج حاصل از این بررسی ارائه‌شده است.



شکل ۷. رده‌بندی منابع آب با در نظر گرفتن دو متغیر Ca/Mg و $SI.Ca$



شکل ۸. رده‌بندی منابع آب با در نظر گرفتن دو متغیر Ca/Mg و TDS

جدول ۳: دقت به‌دست‌آمده از سه روش شبکه عصبی، SVM و رگرسیون

روش	دقت
شبکه عصبی	۱
SVM	۱
رگرسیون(لوژستیک)	۰/۹۷

در شکل‌های ۷ و ۸ نقاط دایره شکل نشان‌دهنده داده‌های رده A با کد ۱ و نقاط مثلث شکل نشان‌دهنده داده‌های رده B با کد ۰ هستند. مطابق با این دو شکل ملاحظه می‌کنید، زمانی که الگوریتم شبکه عصبی برای رده‌بندی داده‌ها تنها دو متغیر Ca/Mg و $SI.Ca$ در شکل ۷ و یا دو متغیر Ca/Mg و TDS در شکل ۸ را برای جداسازی دو رده A و B در نظر می‌گیرد، جداسازی آن‌ها با استفاده از روش شبکه عصبی بسیار دقیق صورت گرفته است البته این امر برای تمام جفت متغیرها در این مسئله برقرار است.

دو عامل مهم در تنظیم و افزایش یا کاهش نرخ خطا در شبکه‌های عصبی عبارت است از تعداد لایه‌های پنهان و تعداد واحدهای موجود در هر لایه، [۷] طوری که هر میزان تعداد لایه‌های پنهان بیش‌تر باشد، انعطاف‌پذیری شبکه افزایش‌یافته و دقت محاسبات افزایش می‌یابد، از طرفی این تعداد را نمی‌توان تا هر میزان دلخواه افزایش داد زیرا ممکن است مسئله به جواب درست همگرا نشود. بنابراین برای یافتن تعداد لایه‌های پنهان بهینه، مانند الگوریتم SVM، باید یک پارامتر جریمه را در قالب عمل ارزشیابی متقابل به مسئله اعمال کرد، که البته در جریان این پژوهش، این امر در نظر گرفته‌شده است. مطابق با نتایج حاصل‌شده از جدول ۳ می‌توان گفت با استفاده از روش‌های آموزش ماشین به راحتی و با میزان دقت ۱ می‌توان عمل رده‌بندی را انجام داد. پس از انجام محاسبات با استفاده از شبکه عصبی، ملاحظه می‌کنیم که تعداد لایه‌های پنهان برابر با ۱۳، حد آستانه برابر با ۰/۰۱ و تعداد تکرارهای لازم برای همگرا شدن مسئله برابر با ۳۵۰ مرحله است. مطابق با جدول ۱ دقت محاسبات با روش SVM با استفاده از چهار هسته نام برده در جدول بالا بوده طوری که بالاترین دقت مربوط به هسته رادیال با ۱۸ بردار پشتیبان و با دقت ۱ است. مطابق با جدول ۲ دقت در روش‌های رگرسیونی نیز بالا است. این میزان دقت در رگرسیون لوژستیک با تابع ربط لجیت که دارای ملاک آکائیکه ۸۲/۰۸۷ و دقت ۰/۹۷ است نسبت به تابع ربط پروبیت با $AIC = ۱۵۴/۱۷$ و دقت ۰/۹۵ ، بالاتر است.

مراجع

[1] Abe, S. (2010). *Support Vector Machines for Pattern Classification* (pp. 331-341). Springer, London.

[2] Boser, B. E., Guyon, I. M., and Vapnik, V. N. (2003). A training algorithm for optimal margin classifiers. *In Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, 144-152.

[3] Gurland, J., Lee, I., and Dahm, P. A. (1960). Polychotomous quantal response in biological assay. *Biometrics*, **16**(3), 382-398.

[4] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: prediction, inference and data mining.* Springer-Verlag, New York.

[5] Hastie, T., Tibshirani, R., Friedman, J., and Franklin, J. (2005). The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, **27**(2), 83-85.

[6] Mantel, N. (1966). Models for complex contingency tables and polychotomous dosage response curves. *Biometrics*, **22(1)**, 83-95.

[7] Muller, A. C., and Guido, S. (2017). *Introduction to machine learning with Python: a guide for data scientists.* O'Reilly Media.

[8] Robbins, H., and Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, **22**(3), 400-407.

[9] Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, **65**(6), 386-408.

[10] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, **323**, 533-536.

[11] Theil, H. (1969). A multinomial extension of the linear logit model. *International economic review*, **10**(3), 251-259.

[12] Widrow, B., and Hoff, M. E. (1960). Adaptive switching circuits. 1960 IRE WESCON convention record (pp. 96-104). *New York: IRE. Reprinted in Anderson and Rosenfeld (1988)*, 126-134.