

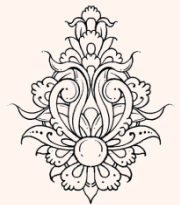
یادگیری ماشین



دانشگاه شهید بهشتی
پژوهشکده‌ی فضای مجازی
پاییز ۱۴۰۳
احمد محمودی ازناوه

فهرست مطالب

- انواع دسته‌بندی
 - بر پایه‌ی درست‌نمایی (مولد)
 - بر پایه‌ی جداساز
- جداساز قطی
 - تعمیم مدل‌های قطی
 - تعبیر هندسی جداساز قطی
 - جداسازی دودسته‌ای و چنددسته‌ای
- مروری بر جداسازی پارامتری
- نزول گرادیان
- Logistic Regression
- Softmax Regression (Multinomial Logistic Regression)
- رتبه‌بندی



دسته‌بندی بر پایه‌ی تابع درست‌نمایی

Likelihood-based classification (Generative models)

- برای دسته‌بندی یک سری «تابع جداساز $(g_i(x))$ » مناسبه می‌شود:

$$g_i(\mathbf{x}) = \max_{j=1}^K g_j(\mathbf{x})$$

اگر

کلاس C_i انتخاب می‌شود

- ابتدا احتمال پیشین $(p(C_i))$ و تابع چگالی احتمال داده‌ها در هر کلاس $(p(x|C_i))$ بر اساس تابع درست‌نمایی مناسبه شده، سپس بر اساس قانون Bayes، احتمال پسین $(P(C_i|x))$ به دست آمده و براساس آن تابع جداساز تعریف می‌شود:

$$g_i(\mathbf{x}) = \log P(C_i|x)$$

- این شیوه به «دسته‌بندی بر پایه‌ی درست‌نمایی» موسوم است. در واقع بر اساس مدلی که برای داده‌ها تخمین زده می‌شود، جداساز به دست می‌آید.

– در روش‌های پارامتری، نیمه‌پارامتری و ناپارامتری از این شیوه



دسته‌بندی بر پایه‌ی جداساز

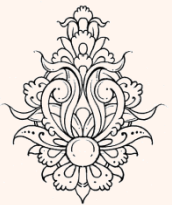
discriminant-based classification (discriminative models)

• در این روش، بدون تخمین توزیع داده‌ها، به صورت مستقیم جداساز تخمین زده می‌شود.

– در این حالت یک مدل برای جداساز تعریف می‌شود:

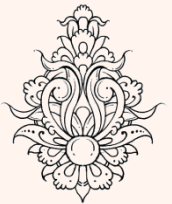
$$g_i(x|\Phi_i)$$

– پارامترهای جداساز به صورت «صریح» مشخص شده‌اند برای خلاف روش‌های مبتنی بر درست‌نمایی که به صورت «ضمنی» و براساس توزیع داده‌ها به دست می‌آیند.



دسته‌بندی بر پایه‌ی جداساز (ادامه...)

- فرآیند آموزش یافتن (بهینه‌سازی) پارامترهای جداساز بر اساس یک مجموعه‌ی آموزشی و با هدف افزایش درستی دسته‌بندی است. در این حالت به جای تخمین درست توزیع داده‌ها هر کلاس، هدف تخمین درست مرزهای بین دسته‌هاست.
- از نظر طرفداران این رویکرد، برآورد توزیع داده‌های یک کلاس از تخمین جداساز، مسأله‌ی دشوارتری است.
 - برای یک مدل مسأله، محقول نیست آن را به مسائل دشوارتر تقسیم کرد!
 - البته زمانی این گفته درست است که جداساز با یک مدل ساده تخمین زده شود.



در ادامه در مورد سه مولفه مشترک مدل‌های یادگیری ماشین برای دسته‌بندی بر پایه جداساز با تمرکز بر **logistic regression** صحبت خواهد شد: ۱- مدل ۲- تابع قطب ۳- بهینه‌سازی

جداساز خطی

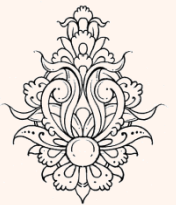
مدل

- ساده‌ترین جداسازی که می‌توان در نظر گرفت، «جداساز خطی» است:

$$g_i(\mathbf{x} | \mathbf{w}_i, w_{i0}) = \mathbf{w}_i^T \mathbf{x} + w_{i0} = \sum_{j=1}^d w_{ij} x_j + w_{i0}$$

- در بیشتر موارد استفاده از جداساز خطی ترجیح داده می‌شود، به دلایل زیر:

- سادگی: دارای پیچیدگی از مرتبه‌ی $o(d)$
- تفسیرپذیری: به راحتی می‌توان بر اساس آن به استخراج دانش پرداخت؛ خروجی مجموع وزن‌دار ورودی است. وزن هر بعد اهمیت و علامت آن اثر آن فصیصه را نشان می‌دهد.
- در بسیاری موارد جداساز خطی بهینه است: توزیع داده‌های کلاس گاوسی با ماتریس کواریانس یکسان



- چنان‌که مدل خطی پیچیدگی لازم را نداشته باشد، می‌توان سراغ جداسازهای پیچیده‌تری رفت:

$$g_i(\mathbf{x} | \mathbf{W}_i, \mathbf{w}_i, w_{i0}) = \mathbf{x}^T \mathbf{W}_i \mathbf{x} + \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

$$g_i(\mathbf{x}) = w_0 + \sum_{i=1}^k w_i x_i + \sum_{i=1}^k \sum_{j=1}^k w_{ij} x_i x_j$$

جداساز درجه‌ی دو

$$\Psi(x) = \sum_1^N c_i \varphi_i(x).$$

- دارای پیچیدگی از مرتبه‌ی $O(d^2)$
- نیاز به داده‌های آموزشی بیشتر
- احتمال بروز overfitting بیشتر

- یک راه معادل استفاده از جملات با مرتبه بالاتر (higher order terms) است.

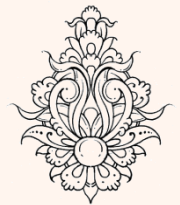
$$z_1 = x_1, z_2 = x_2, z_3 = x_1^2, z_4 = x_2^2, z_5 = x_1 x_2$$

- به جای جداساز پیچیده نگاشت غیر خطی به فضای با جداساز خطی

$$g_i(\mathbf{x}) = \sum_{j=0}^k w_{ij} \varphi_j(\mathbf{x})$$

Potential functions(1964)

Basis functions



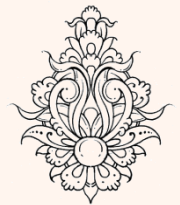
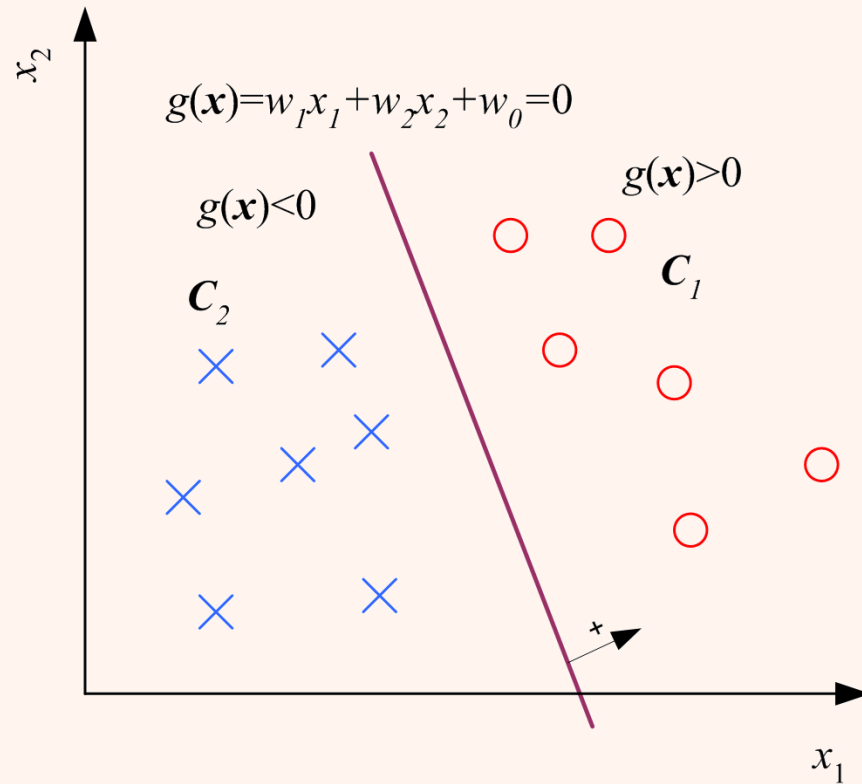
- در این حالت یک تابع جداساز (اگر سطح جداکننده) کافیست:

$$\begin{aligned}
 g(\mathbf{x}) &= g_1(\mathbf{x}) - g_2(\mathbf{x}) \\
 &= (\mathbf{w}_1^T \mathbf{x} + w_{10}) - (\mathbf{w}_2^T \mathbf{x} + w_{20}) \\
 &= (\mathbf{w}_1 - \mathbf{w}_2)^T \mathbf{x} + (w_{10} - w_{20}) \\
 &= \mathbf{w}^T \mathbf{x} + w_0
 \end{aligned}$$

بردار وزنها

مدآستانه

$$\text{choose } \begin{cases} C_1 & \text{if } g(\mathbf{x}) > 0 \\ C_2 & \text{otherwise} \end{cases}$$



دسته‌بندی دوتایی-تعبیر هندسی

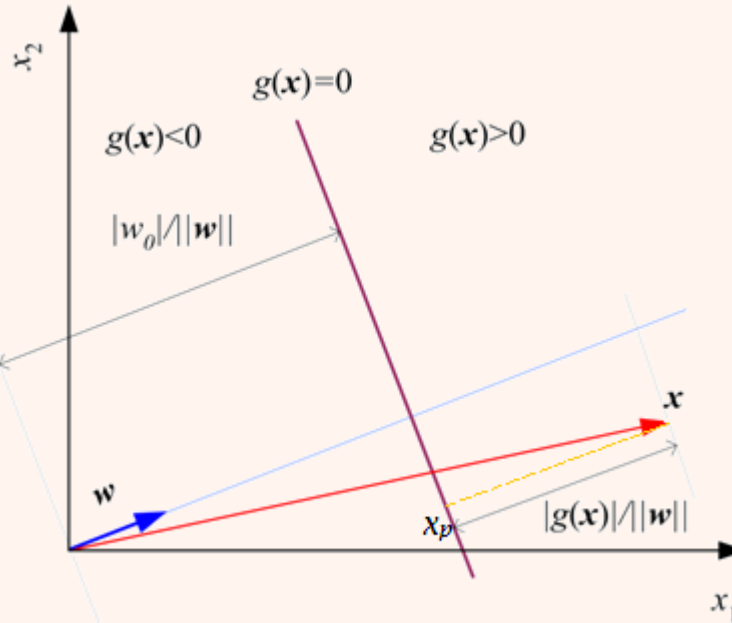
- در صورتی که r فاصله‌ی x از [ابر] سطح جداکننده باشد و x_p نگاشت x بر روی سطح، خواهیم داشت:

$$x = x_p + r \frac{w}{\|w\|}$$

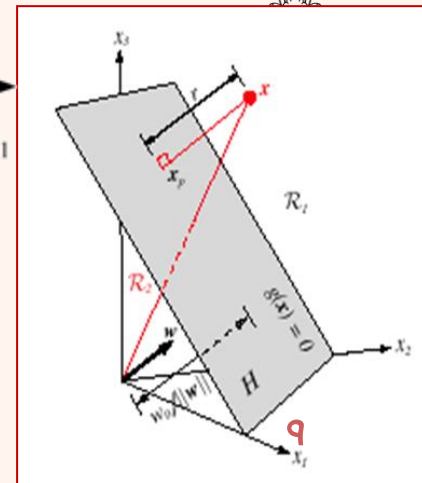
$$g(x) = w^T \left[x_p + r \frac{w}{\|w\|} \right] + w_0$$

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

$$g(x) = r \|w\| \quad r = \frac{g(x)}{\|w\|} \quad r_0 = \frac{w_0}{\|w\|}$$



Duda et al.



• در این حالت به k تابع جداساز نیاز است:

$$g_i(\mathbf{x} | \mathbf{w}_i, w_{i0}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

در نظر گرفتن چنین جداسازی به معنای این است که همگی دسته‌ها جدایی‌پذیر فطی در نظر گرفته شوند.

$$g_i(\mathbf{x} | \mathbf{w}_i, w_{i0}) = \begin{cases} > 0 & \text{if } \mathbf{x} \in C_i \\ \leq 0 & \text{otherwise} \end{cases}$$

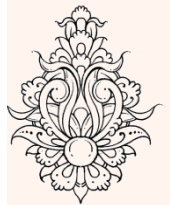
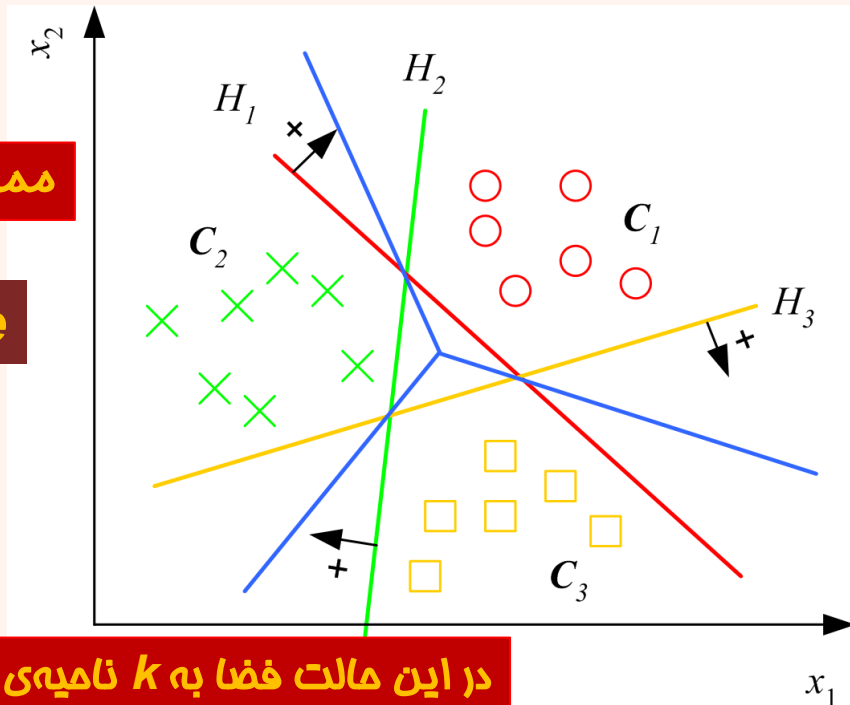
Linearly separable

ممکن است همگی دسته‌ها جدایی‌پذیر فطی نباشند:

Choose C_i if

Linear machine

$$g_i(\mathbf{x}) = \max_{j=1}^K g_j(\mathbf{x})$$



در این حالت فضا به k نامیهی ممدب تقسیم می‌شود.

Pairwise Separation

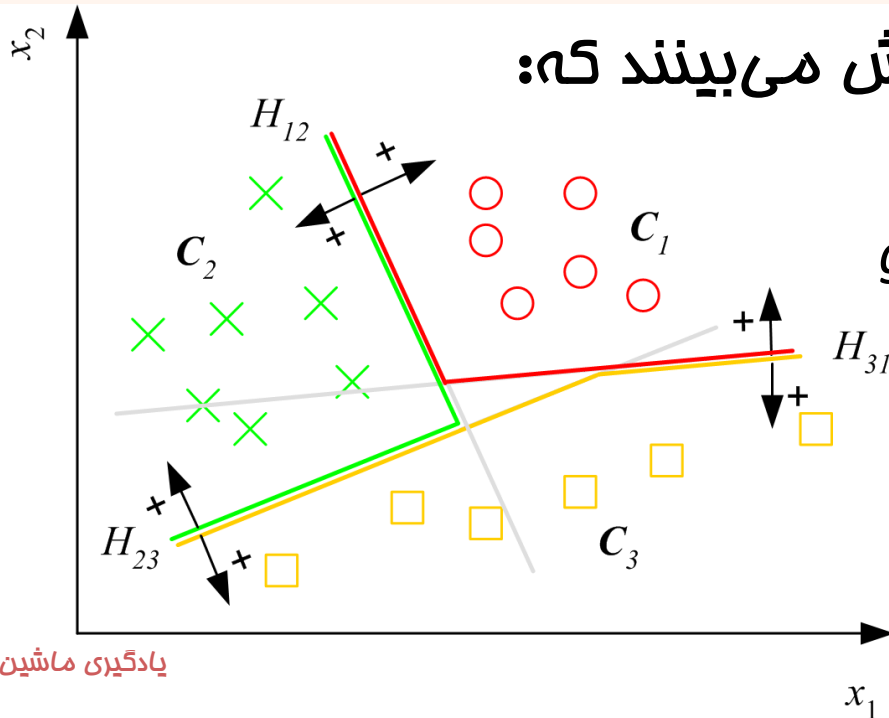
one-versus-one

جداساز دویبه دو

- اگر همه‌ی کلاس‌ها جدایی‌پذیر خطی نباشند، یک رویکرد مناسب تقسیم مسأله به چند جدایی‌ساز خطی است.
- برای هر دو کلاس یک جداساز تعریف شود.
- در این صورت $k(k-1)/2$ جداساز مورد نیاز است.

$$g_{ij}(\mathbf{x} | \mathbf{w}_{ij}, w_{ij0}) = \mathbf{w}_{ij}^T \mathbf{x} + w_{ij0}$$

- پارامترها به گونه‌ای آموزش می‌بینند که:



$$g_{ij}(\mathbf{x}) = \begin{cases} > 0 & \text{if } \mathbf{x} \in C_i \\ \leq 0 & \text{if } \mathbf{x} \in C_j \\ \text{don't care} & \text{otherwise} \end{cases}$$

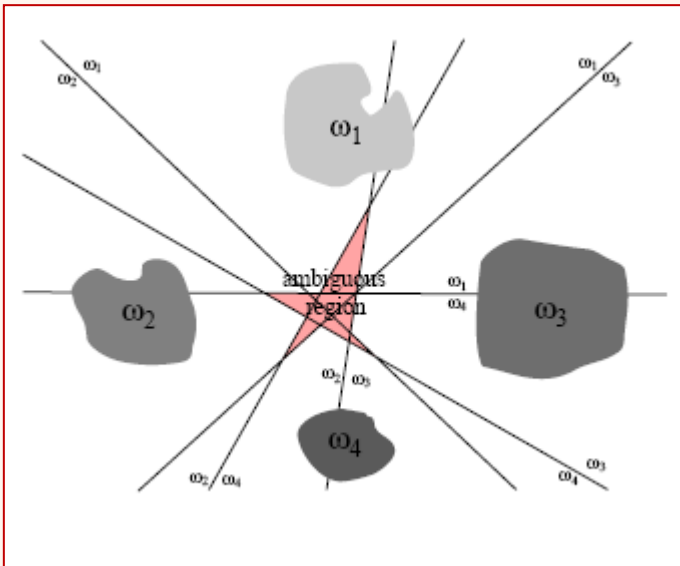
- در زمان آزمایش:
choose C_i if

$$\forall j \neq i, g_{ij}(\mathbf{x}) > 0$$



- در این حالت ممکن است برخی نواحی رابطه‌ی پیش برقرار نباشد.
- در این حالت معیار دیگری را می‌توان جایگزین نمود:

$$g_i(\mathbf{x}) = \sum_{j \neq i} g_{ij}(\mathbf{x})$$



Duda et. al.

این شیوه نیز نمونه‌ی دیگری از تبدیل یک مسأله‌ی پیچیده به چند مسأله‌ی ساده‌تر است

مروری بر جداسازی پارامتری

تابع خطا

در صورتی که داده‌ها از توزیع گاوسی تبعیت کنند و ماتریس کواریانس **یکسانی** داشته باشند، **جداساز بهینه**، **خطی** خواهد بود:

$$p(\mathbf{x} | C_i) \sim N(\mu_i, \Sigma)$$

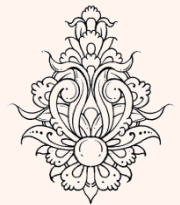
$$g_i(\mathbf{x} | \mathbf{w}_i, w_{i0}) = \mathbf{w}_i^T \mathbf{x} + w_{i0}$$

$$\mathbf{w}_i = \Sigma^{-1} \mu_i \quad w_{i0} = -\frac{1}{2} \mu_i^T \Sigma^{-1} \mu_i + \log P(C_i)$$

- برای یک مجموعه‌ی آموزشی ابتدا مقادیر میانگین و ماتریس کواریانس هر کلاس محاسبه می‌شود.
- برای حالت دو کلاس، خواهیم داشت:

$$y \equiv P(C_1 | \mathbf{x}) \quad \text{and} \quad P(C_2 | \mathbf{x}) = 1 - y$$

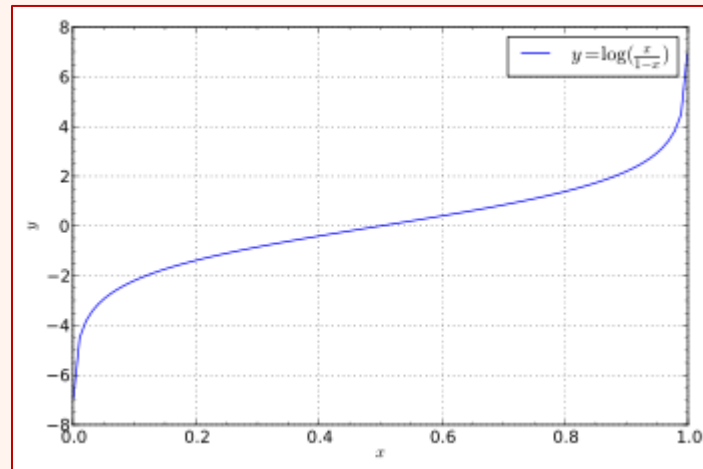
$$\text{choose } C_1 \text{ if } \begin{cases} y > 0.5 \\ y / (1 - y) > 1 \\ \log [y / (1 - y)] > 0 \end{cases} \quad \text{and } C_2 \text{ otherwise}$$



مروری بر جداسازی پارامتری (ادامه...)

- LOGIT به صورت زیر تعریف می‌شود:

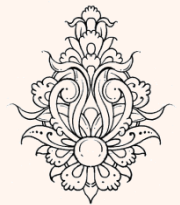
$$\text{logit}(y) = \log \frac{y}{1-y}$$



در نتیجه داده به کلاس یک تعلق دارد اگر و تنها اگر

$$\text{logit}(P(C_1|\mathbf{x})) > 0$$

وگرنه متعلق به کلاس دو خواهد بود.



مروری بر جداسازی پارامتری (ادامه...)

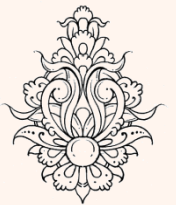
$$\begin{aligned}\text{logit}(P(C_1|\mathbf{x})) &= \log \frac{P(C_1|\mathbf{x})}{1-P(C_1|\mathbf{x})} = \log \frac{P(C_1|\mathbf{x})}{P(C_2|\mathbf{x})} \\ &= \log \frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_2)} + \log \frac{P(C_1)}{P(C_2)} \\ &= \log \frac{(2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left[-(1/2)(\mathbf{x}-\mu_1)^T \Sigma^{-1} (\mathbf{x}-\mu_1)\right]}{(2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left[-(1/2)(\mathbf{x}-\mu_2)^T \Sigma^{-1} (\mathbf{x}-\mu_2)\right]} + \log \frac{P(C_1)}{P(C_2)} \\ &= \mathbf{w}^T \mathbf{x} + w_0\end{aligned}$$

$$\text{where } \mathbf{w} = \Sigma^{-1} (\mu_1 - \mu_2) \quad w_0 = -\frac{1}{2} (\mu_1 + \mu_2)^T \Sigma^{-1} (\mu_1 - \mu_2) + \log \frac{P(C_1)}{P(C_2)}$$

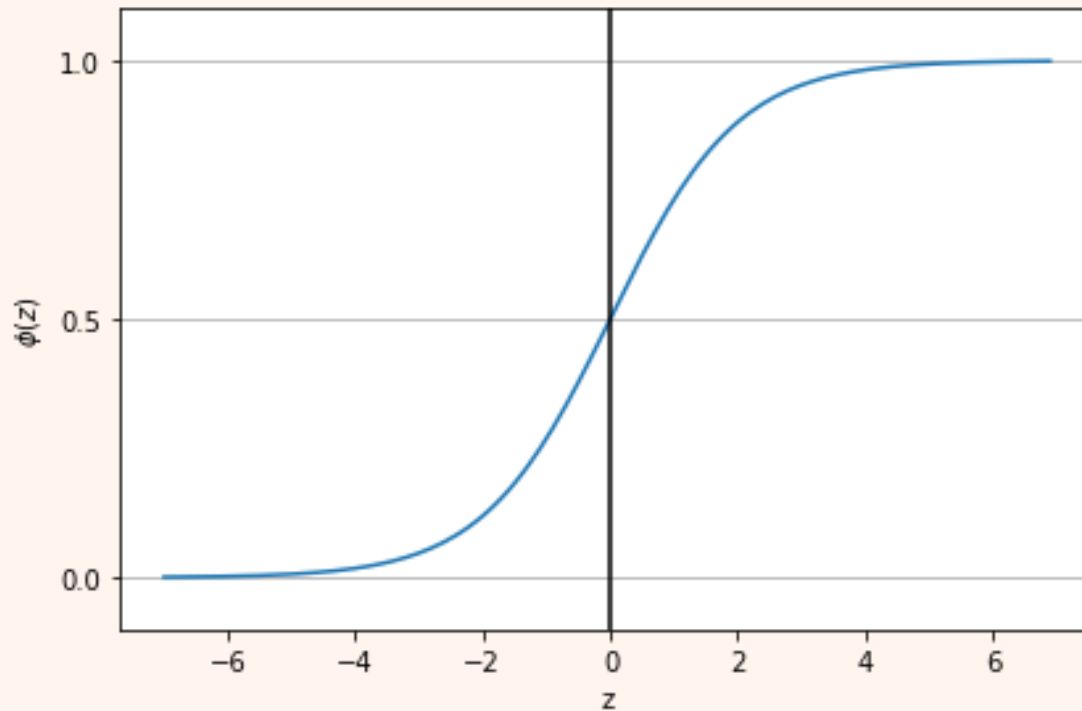
The inverse of logit

$$\log \frac{P(C_1|\mathbf{x})}{1-P(C_1|\mathbf{x})} = \mathbf{w}^T \mathbf{x} + w_0$$

$$P(C_1|\mathbf{x}) = \frac{1}{1 + \exp\left[-(\mathbf{w}^T \mathbf{x} + w_0)\right]} = \text{sigmoid}(\mathbf{w}^T \mathbf{x} + w_0)$$



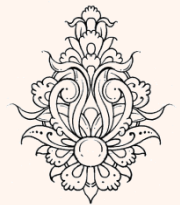
تابع sigmoid



Calculate $g(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$ and choose C_1 if $g(\mathbf{x}) > 0$, or

Calculate $y = \text{sigmoid}(\mathbf{w}^T \mathbf{x} + w_0)$ and choose C_1 if $y > 0.5$

در حالت دوم تابع sigmoid ارتباط مقدار تابع
جداساز و احتمال پسین را نشان می‌دهد.



Logistic Regression

- در دسته‌بندی مبتنی بر درست‌نمایی ابتدا، $P(C_1)$ و $P(x|C_1)$ محاسبه شده، سپس مقدار $P(C_1|x)$ به دست می‌آید.

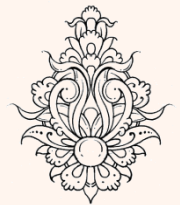
- در logistic discrimination احتمال پسین به صورت مستقیم برآورد می‌شود.

- در صورتی که لگاریتم نسبت درست‌نمایی دو کلاس خطی باشد:

$$\log \frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_2)} = \mathbf{w}^T \mathbf{x} + w_0^o$$

- با توجه به قانون Bayes:

$$\text{logit}(P(C_1|\mathbf{x})) = \log \frac{P(C_1|\mathbf{x})}{1 - P(C_1|\mathbf{x})} = \log \frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_2)} + \log \frac{P(C_1)}{P(C_2)}$$



حالت دوکلاسه

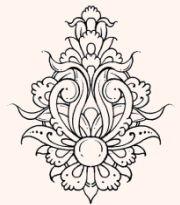
$$\text{logit}(P(C_1|\mathbf{x})) = \log \frac{P(C_1|\mathbf{x})}{1-P(C_1|\mathbf{x})} = \log \frac{p(\mathbf{x}|C_1)}{p(\mathbf{x}|C_2)} + \log \frac{P(C_1)}{P(C_2)}$$
$$= \mathbf{w}^T \mathbf{x} + w_0$$

where $w_0 = w_0^o + \log \frac{P(C_1)}{P(C_2)}$

$$y = \hat{P}(C_1|\mathbf{x}) = \frac{1}{1 + \exp[-(\mathbf{w}^T \mathbf{x} + w_0)]}$$

بدین ترتیب، تقریبی از احتمال پسین به دست می‌آید.

در واقع مسأله یافتن (آموختن) w و w_0 است.



آموزش (حالت دو کلاسه)

$$y = \hat{P}(C_1 | \mathbf{x}) = \frac{1}{1 + \exp[-(\mathbf{w}^T \mathbf{x} + w_0)]}$$

$$\mathcal{X} = \{\mathbf{x}^t, r^t\}_t \quad r^t | \mathbf{x}^t \sim \text{Bernoulli}(y^t)$$

در این جا به صورت مستقیم، مقدار y تخمین زده می‌شود. درست‌نمایی \mathcal{X} به ازای پارامترهای مورد نظر محاسبه شده:

$$l(\mathbf{w}, w_0 | \mathcal{X}) = \prod_t (y^t)^{r^t} (1 - y^t)^{(1-r^t)}$$

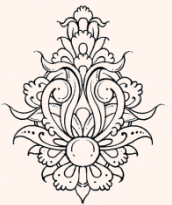
بر اساس آن تابع خطا محاسبه می‌شود:

$$E = -\log l$$

Binary cross entropy

$$E(\mathbf{w}, w_0 | \mathcal{X}) = -\sum_t r^t \log y^t + (1 - r^t) \log (1 - y^t)$$

هدف کاهش میزان خطا است! هیچ راه تحلیلی برای حل این مسأله وجود ندارد.

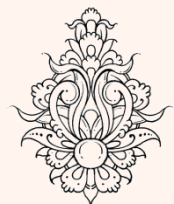


• در دسته‌بندی مبتنی بر جداساز، پارامترها به گونه‌ای بهینه‌سازی می‌شوند که فضای دسته‌بندی حداقل شود:

$$w^* = \arg \min_w E(w | X)$$

- در بیشتر مواقع، راه حل تحلیلی برای یافتن پارامترها وجود ندارد و چاره‌ای جز استفاده از یک روش بهینه‌سازی تکرارشونده نخواهد بود.
- استفاده از نزول گرادیان یکی از پرکاربردترین راه‌کارهاست.
- بردار گرادیان به صورت زیر تعریف می‌شود:

$$\nabla_w E = \left[\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \dots, \frac{\partial E}{\partial w_d} \right]^T$$

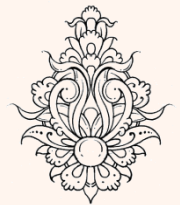
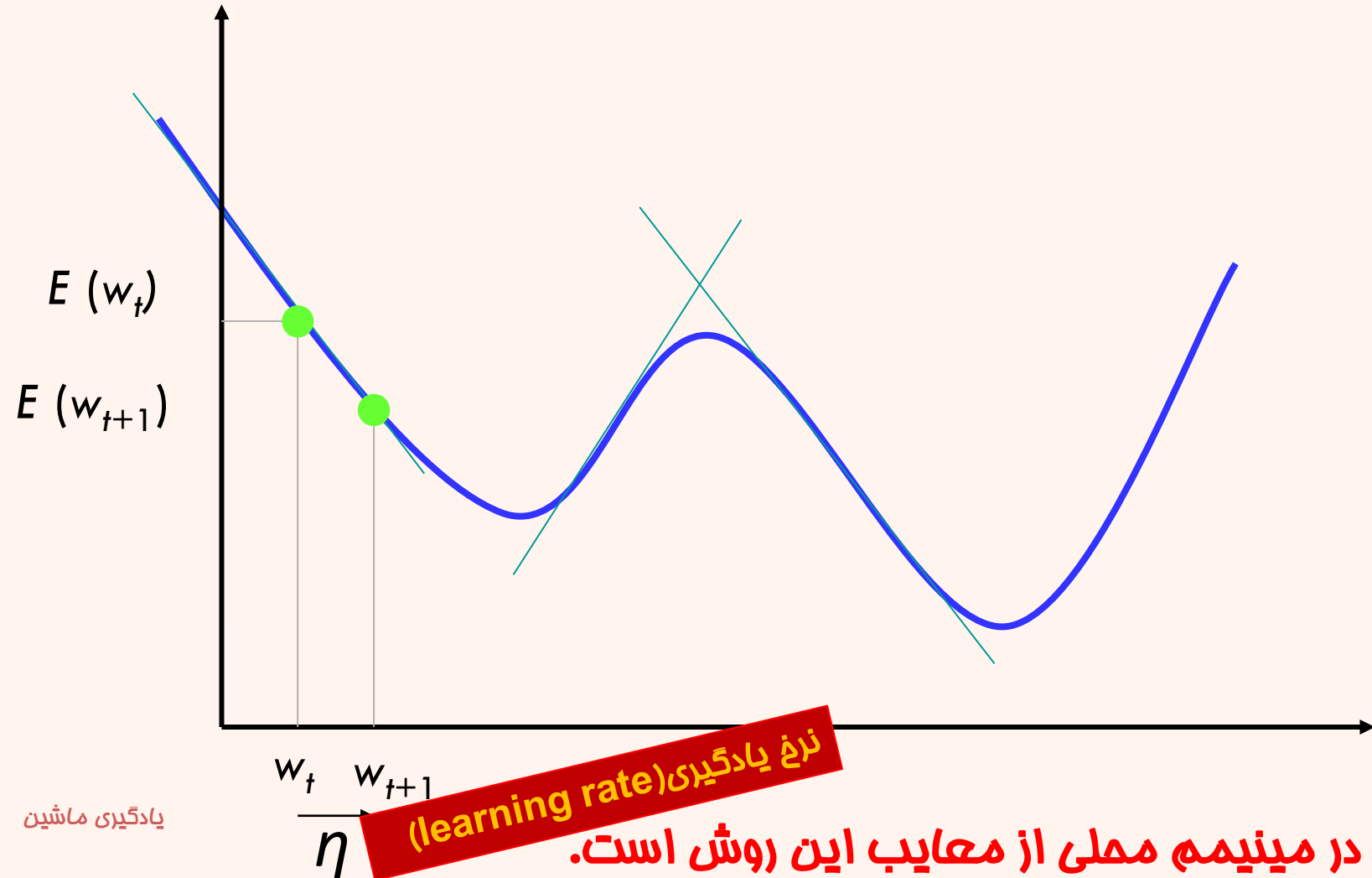


Gradient-Descent

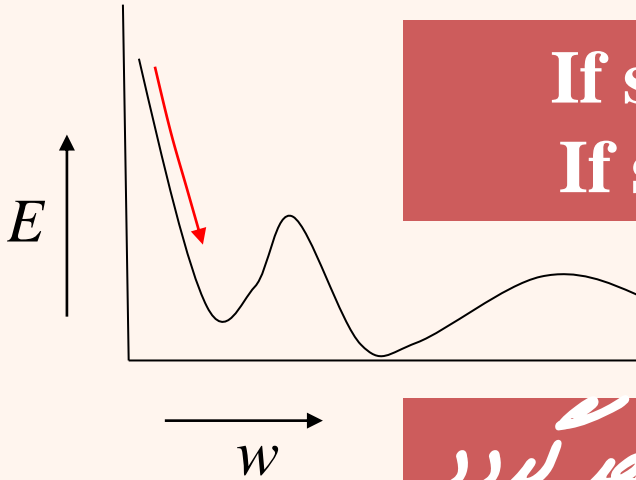
$$\Delta w_t = -\eta \frac{\partial E}{\partial w_t}, \forall i$$

$$w_{t+1} = w_t + \Delta w_t$$

- پارامترها با یک مقدار تصادفی مقداردهی می‌شوند.
- بر خلاف جهت گرادیان مقدار پارامترها را به روز می‌شوند.

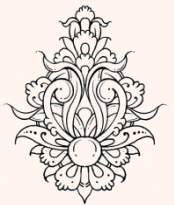
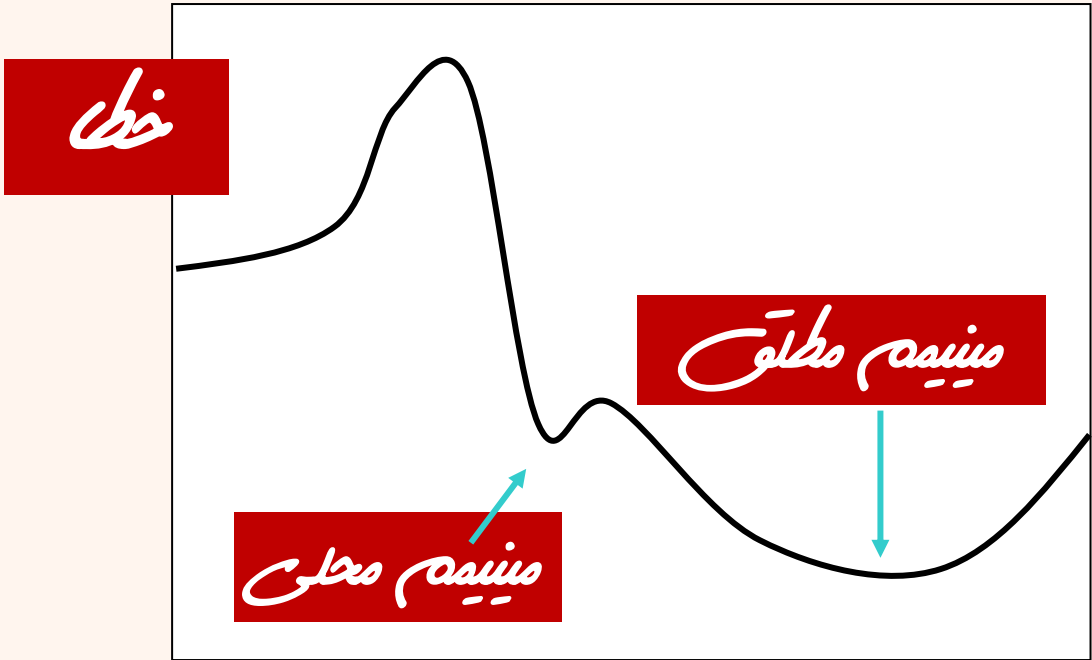


Gradient Descent



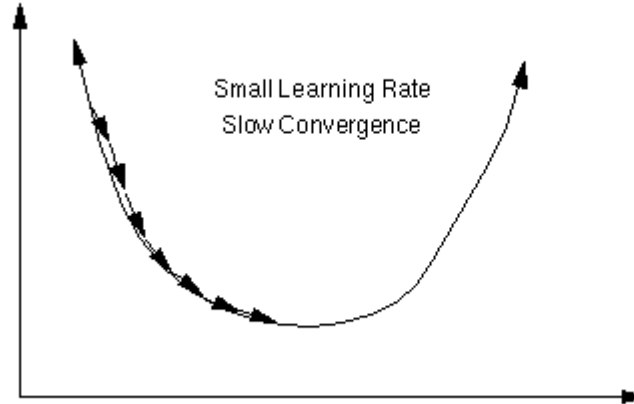
If slope is negative \rightarrow increase w
If slope is positive \rightarrow decrease w

مینیمم محلی جایی است که مشتق صفر گردد

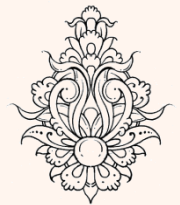
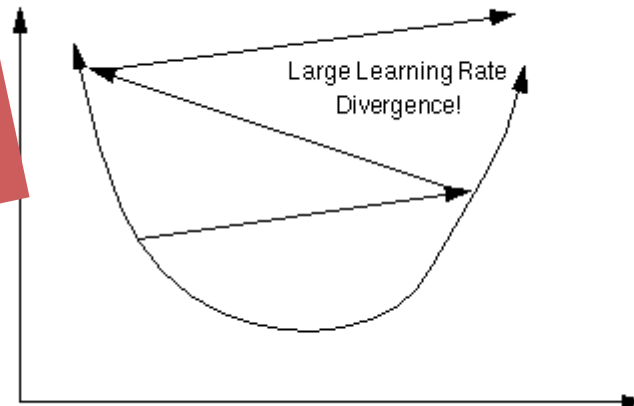


تنظیم نرخ یادگیری

همه را این کند است.

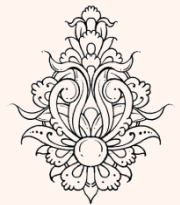
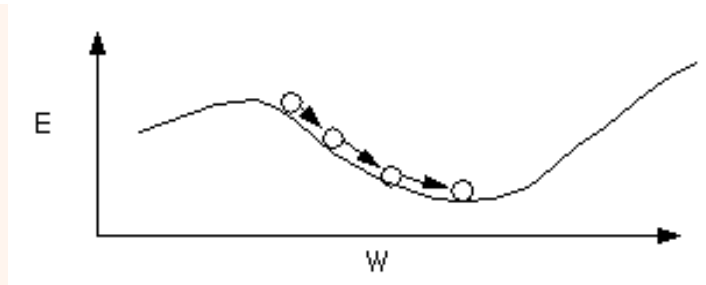
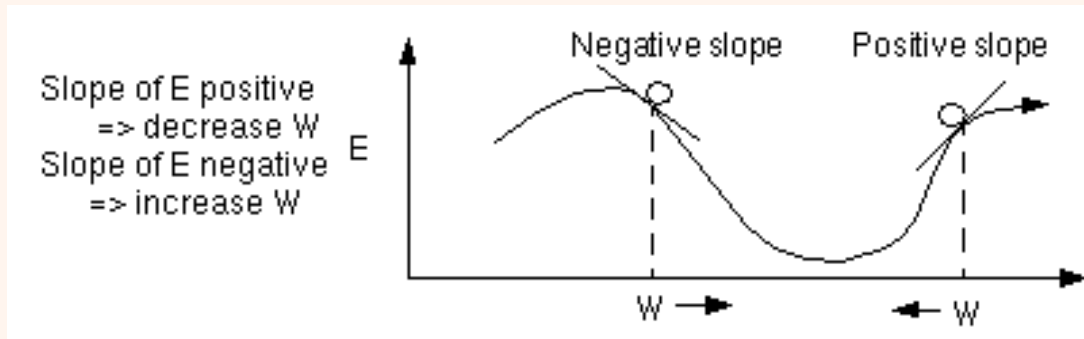
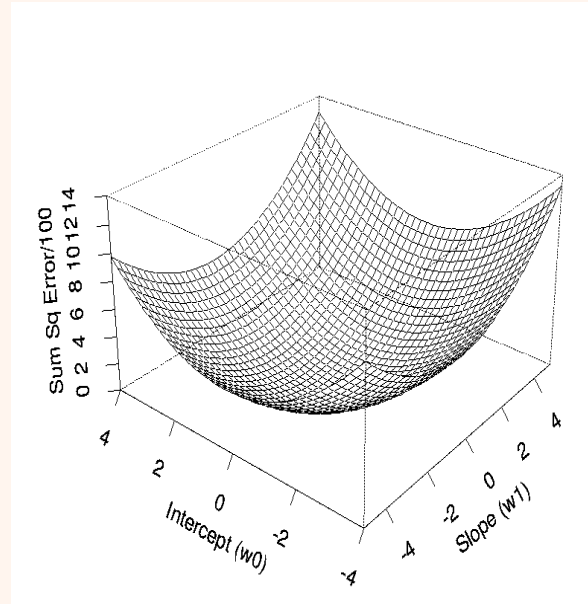
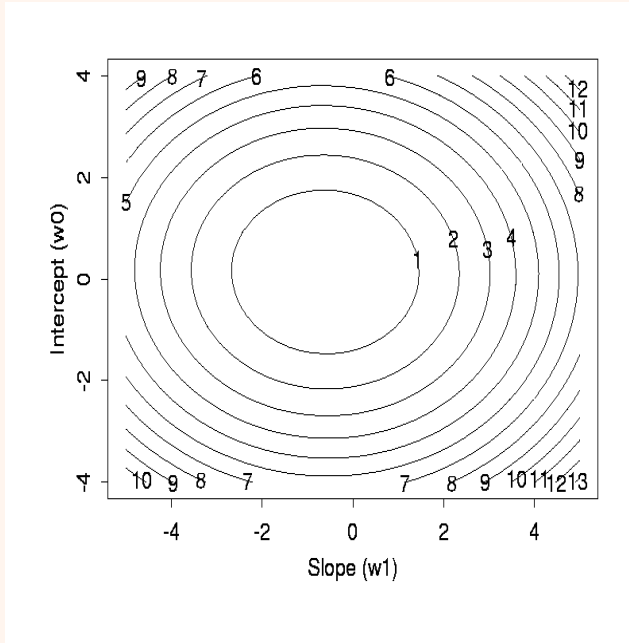


بیستم ناپایدار است.



Steepest descent (کمینه کردن خطا (ادامه...))

rule



انواع شیوه‌های آموزش

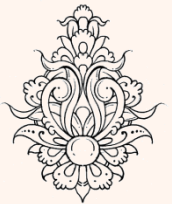
- آموزش به دو صورت قابل انجام است:

- **دسته‌ای:**

در این شیوه تابع خطایی که بناست مورد استفاده قرار گیرد، تابع خطا به ازای اعمال همگی ورودی‌هاست.

- **ترتیبی:**

در این شیوه ورودی‌های تک‌تک اعمال شده، پس از محاسبه‌ی خطا، پارامترها اصلاح شده و به همین ترتیب ورودی بعدی اعمال می‌شود.



آموزش (حالت دو کلاسه)، نزول گرادیان

$$y = \hat{P}(C_1|\mathbf{x}) = \frac{1}{1 + \exp[-(\mathbf{w}^T \mathbf{x} + w_0)]}$$

$$E(\mathbf{w}, w_0|\mathcal{X}) = -\sum_t r^t \log y^t + (1 - r^t) \log (1 - y^t)$$

$g(x)$

If $y = \text{sigmoid}(a)$ $\frac{dy}{da} = y(1 - y)$

$$\Delta w_j = -\eta \frac{\partial E}{\partial w_j}$$

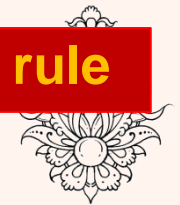
$$\frac{\partial E}{\partial w_j} = \frac{\partial E}{\partial y^t} \times \frac{\partial y^t}{\partial g} \times \frac{\partial g}{\partial w_j}$$

$$= \eta \sum_t \left(\frac{r^t}{y^t} - \frac{1 - r^t}{1 - y^t} \right) y^t (1 - y^t) x_j^t$$

$$= \eta \sum_t (r^t - y^t) x_j^t, j = 1, \dots, d$$

$$\Delta w_0 = -\eta \frac{\partial E}{\partial w_0} = \eta \sum_t (r^t - y^t)$$

Chain rule



الگوریتم

For $j = 0, \dots, d$
 $w_j \leftarrow \text{rand}(-0.01, 0.01)$

Repeat

For $j = 0, \dots, d$

$\Delta w_j \leftarrow 0$

For $t = 1, \dots, N$

$o \leftarrow 0$

For $j = 0, \dots, d$

$o \leftarrow o + w_j x_j^t$

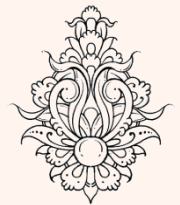
$y \leftarrow \text{sigmoid}(o)$

$\Delta w_j \leftarrow \Delta w_j + (r^t - y)x_j^t$

For $j = 0, \dots, d$

$w_j \leftarrow w_j + \eta \Delta w_j$

Until convergence



$$E(\mathbf{w}, w_0 | \mathcal{X}) = -\sum_t r^t \log y^t + (1 - r^t) \log (1 - y^t) + \frac{\lambda}{2} \|W\|^2$$

استفاده از منظم سازی به ویژه در ابعاد بالا به مدل کمک می کند که تر از نویز تاثیر بگیرد.



حالت چنددسته‌ای

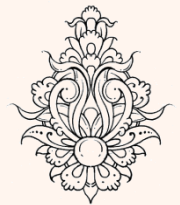
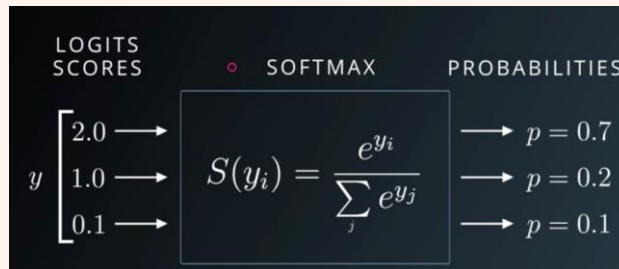
- در این حالت می‌توانیم سراغ شیوه o.v.a برویم.
- فاصله هر نمونه از مرز متناسب با احتمال است.
- برای تبدیل این فاصله به احتمال می‌توان به صورت زیر عمل کرد:

$$y_i = \hat{P}(C_i | \mathbf{x}) = \frac{\exp[\mathbf{w}_i^T \mathbf{x} + w_{i0}]}{\sum_{j=1}^K \exp[\mathbf{w}_j^T \mathbf{x} + w_{j0}]}, i = 1, \dots, K$$

softmax

محاسبه‌ی ماکزیمم است، با این تفاوت که مشتق‌پذیر است.

Normalization
Exponential Scaling
differentiable



$$\mathcal{X} = \{\mathbf{x}^t, \mathbf{r}^t\}_t \quad r^t | \mathbf{x}^t \sim \text{Mult}_K(1, \mathbf{y}^t)$$

$$y_i = \frac{\exp[\mathbf{w}_i^T \mathbf{x} + w_{i0}]}{\sum_{j=1}^K \exp[\mathbf{w}_j^T \mathbf{x} + w_{j0}]}$$

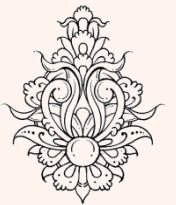
$$l(\{\mathbf{w}_i, w_{i0}\}_i | \mathcal{X}) = \prod_t \prod_i (y_i^t)^{r_i^t}$$

cross entropy

$$E(\{\mathbf{w}_i, w_{i0}\}_i | \mathcal{X}) = -\sum_t \sum_i r_i^t \log y_i^t$$

هدف کاهش میزان خطا است. هیچ راه تملیلی برای حل این مسأله وجود ندارد.

$$\Delta \mathbf{w}_j = \eta \sum_t (r_j^t - y_j^t) \mathbf{x}^t \quad \Delta w_{j0} = \eta \sum_t (r_j^t - y_j^t)$$



الگوریتم

For $i = 1, \dots, K$, For $j = 0, \dots, d$, $w_{ij} \leftarrow \text{rand}(-0.01, 0.01)$

Repeat

For $i = 1, \dots, K$, For $j = 0, \dots, d$, $\Delta w_{ij} \leftarrow 0$

For $t = 1, \dots, N$

For $i = 1, \dots, K$

$o_i \leftarrow 0$

For $j = 0, \dots, d$

$o_i \leftarrow o_i + w_{ij}x_j^t$

For $i = 1, \dots, K$

$y_i \leftarrow \exp(o_i) / \sum_k \exp(o_k)$

For $i = 1, \dots, K$

For $j = 0, \dots, d$

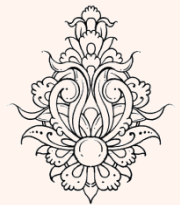
$\Delta w_{ij} \leftarrow \Delta w_{ij} + (r_i^t - y_i)x_j^t$

For $i = 1, \dots, K$

For $j = 0, \dots, d$

$w_{ij} \leftarrow w_{ij} + \eta \Delta w_{ij}$

Until convergence



یادگیری رتبه‌بندی

- رتبه‌بندی مسأله‌ای متفاوت از دسته‌بندی و رگرسیون است.

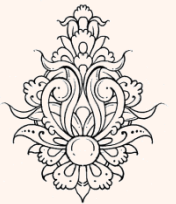
- به عنوان مثال در موتورهای جستجو

- به عنوان مثال اگر x^u و x^v دو نمونه باشند، مثلاً دو فیلم u به v ترجیح داده شود، در این حالت یک تابع برای امتیازدهی ($g(x/\theta)$) آموزش داده می‌شود به نحوی که

- $g(x^u) > g(x^v)$

- و در نهایت

- Choose u if $g(x^u/\theta) = \max_t g(x^t/\theta)$



یادگیری رتبه‌بندی (ادامه...)

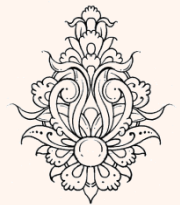
- در این حالت تابع خطا به صورت زیر به دست می‌آید:

$$E(\mathbf{w} | \{r^u, r^v\}) = \sum_{r^u < r^v} [g(\mathbf{x}^v | \theta) - g(\mathbf{x}^u | \theta)]_+$$

where a_+ is equal to a if $a \geq 0$ and 0 otherwise.

- در صورتی که مدل خطی در نظر گرفته شود:
- $g(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$
- تعبیر هندسی آن یعنی یافتن جهتی که در صورتی که داده‌ها در راستای آن نگاهت شوند، رتبه‌ی مورد نظر به دست آید.

$$\Delta w_j = -\eta \frac{\partial E}{\partial w_j} = -\eta (x_j^v - x_j^u), j = 1, \dots, d$$



مثال

