Subject:

Year. Month. Date. ( )

ML_ 4th Session

در این جلسه ابتدا در مورد KNN صحبت کردیم که KNN هم برای regression و هم برای clasification

استفاده میشود، این روش جزء روشهای non-parametric است و ساده‌ترین روش non-P است

تفاوت روشهای parametric و non-parametric :

تمام روشهایی که تا الان بررسی کردیم یک فرم پارامتری بصورت $Y = B^T X + \varepsilon$ داشت یعنی ارتباط بین

متغیرهای X و Y را میتوانستیم براساس تعدادی پارامتر مشخص کنیم اما در روشهای non-Par فرضی روی

فرم تابع نداریم که در این روشها تلاش میکنیم به دنیای واقعی نزدیک بشیم بطوریکه فرم شکل تابع خیلی ridge

نباشد که جزء مزیت‌های این روش است اما عیب اصلی این روشها این است که به تعداد زیادی مشاهده برای

یک تخمین خوب نیاز داریم.

K_nearest neighbors regression:

در این روش یک معیار به داده و پارامتر K را دریافت می‌کند و برای هر مشاهده براساس K همسایه نزدیک به نقطه‌ی

مورد نظر پیش بینی انجام میشه. برای مثال وقتی $K=3$ باشد برای هر x جدید سه تا از همسایه‌های نزدیکش را

در نظر میگیرد و میانگین Y های این سه همسایه میشود $\hat{Y}$ که هدف تخمین آن بوده است.

$$\Rightarrow \hat{f}(x_0) = \frac{1}{K} \sum_{i \in N_0} y[i]$$

Subject:

Year. Month. Date. ( )

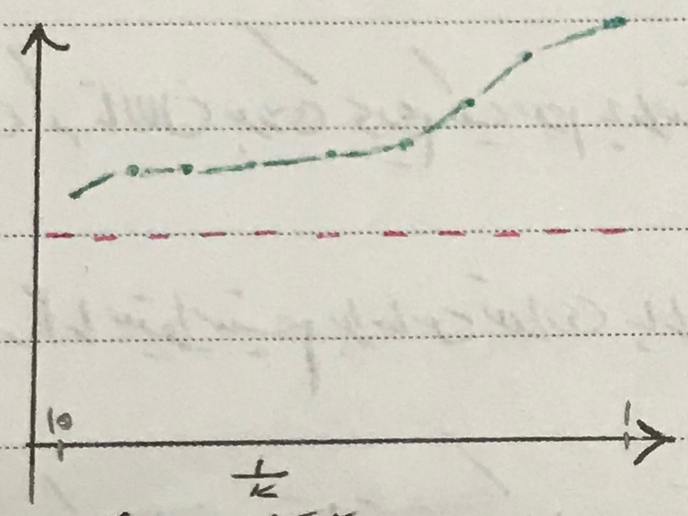
if $k=1 \Rightarrow RSS_{train}=0$ برای هر نقطه تخمین خودش میشه.

bias خیلی کم

varience خیلی زیاد

if $k=10 \Rightarrow$ bias بیشتر میشه و varience کمتر میشه

* در واقع هر چه تعداد همسایه‌ها مورد نظر برای تخمین Y را بیشتر کنیم varience را کاهش میدهیم اما bias



افزایش پیدا میکند.

همانطور که در شکل میبینیم وقتی k را کاهش میدهیم

خطا روی دیتای تست وقتی k را خیلی کاهش میدهیم

خطا افزایش پیدا میکند به دلیل مشکلات overfitting

* اگر ارتباط بین X و Y خطی نباشد آنگاه KNN برای یه سری kهای خاص خطای آن از

خطای least squre کمتر است و اگر عزم polynomial داشته باشد آنگاه KNN همیشه بهتر از

LS است.

مشکلی که KNN دارد این است که وقتی بجای یک متغیر 10 متغیر داشته باشیم آنگاه مشاهدات در فضای

10 بعدی پخش میشوند و وقتی بخواهیم برای یک x_0 نزدیکترین نقاط را پیدا کنیم ممکنه این نزدیکترین نقاط

خیلی دور باشند و به تخمین خوبی نرسیم.

$P\uparrow \Rightarrow MSE_{LS} < MSE_{KNN}$

Sunwood

Subject:

Year. Month. Date. ( )

□□ classification

همانطور که قبلاً گفتیم متغیر response در این مسائل کمی (categorical) است.

← مسائل بیوانفورماتیکی که در این حوزه مطرح میشه مثلاً بیان تعدادی ژن را میدهند و متغیر response

این است که آیا سرطان ریه دارد یا خیر؟

در این مسائل loss Fun را تعبیر میکنیم و یکی از مهمترین آن ها در مسائل classification استفاده از

تابع زیر است:

zero-one loss function : $\frac{1}{n}\sum_{i=1}^{n} I(y[i] \neq \hat{y}[i])$

$$I(y[i] \neq \hat{y}[i]) = \begin{cases} 1 & y[i] \neq \hat{y}[i] \\ 0 & y[i] = \hat{y}[i] \end{cases}$$

تعداد دفعاتی که پیش بینی مثلاً باید کلاس ① میخاست پیش بینی گفته کلاس ⓪ را محاسبه میکند.

$\Rightarrow$ training error : $\frac{1}{n}\sum_{i=1}^{n} I(y[i] \neq \hat{y}[i])$

test error : $E(I(y \neq \hat{y}))$

○ Bayes classifier

کلاسیفایری است که میخوقت نمی توانیم بسازیم و فقط در تئوری بررسی میشه و میگیم به ازای هر دیتایی

کدومی را بهش نسبت بدیم که احتمالش بیشتره، به بیان ریاضی داریم:

$\hat{y} = \underset{c}{\operatorname{argmax}}\, P(y=c \mid x) \xrightarrow{\text{مثلاً}} P(Defult=Yes \mid Balance=1200)=?$

$P(Defult=No \mid Balance=1200)=?$

← هر کدوم بیشتر بود اون میشه $\hat{y}$ مورد نظر

Subject:

Year. Month. Date. ( )

اما قابل محاسبه نیست چون هیچوقت بی نهایت دیتا نداریم.

- Bayes error rate

* کلاسیفایر بیزین کمترین خطای ممکن را برای دیتای تست دارد.

* اما هنوز هم خطا داریم و این خطا معادل همان خطای غیرقابل اجتناب در بحث رگرسیون است و بدلیل ویژگی های است که قابل اندازه گیری نیست.

? 1 − $E(\arg\max_c P(Y=c|X))$ ← تقاضی است.

$P(\text{Default}=\text{yes} \mid \text{Balance}=1200) = ?$

$P(\text{Default}=\text{yes} \mid \text{invest}=400) = ?$

$P(\text{Default}=\text{yes} \mid x_r = 200) = ?$

جمع میانگین

- K_nearest neighbor classifier:

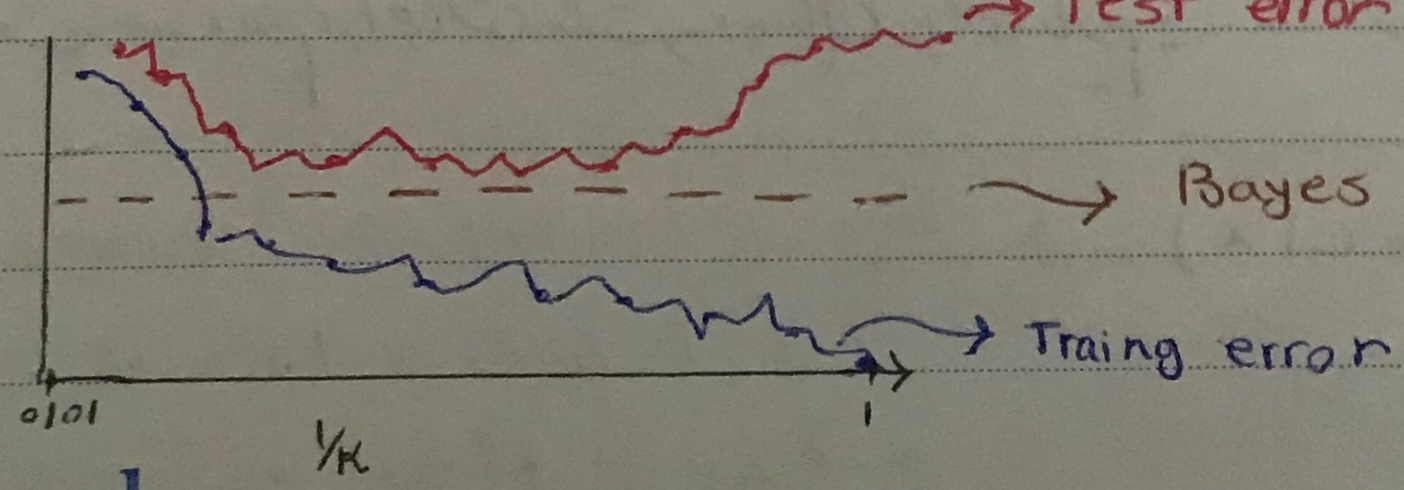
همانطور که گفتیم چون $P(Y|X)$ را برای دیتای واقعی نمیدانیم، بنابراین محاسبه کلاسیفایر بیزی امکان پذیر نیست. حال کلاسیفایرهایی هستند که این مقدار را تخمین میزنند و یکی از آن ها KNN است که:

$$P(Y=c|X) = \frac{1}{K} \sum_{i \in N_0} I(y[i]=c)$$

← K همسایه نزدیک x

- مثلاً $K=10$ و احتمال آبی بودن کلاس را میخواهیم محاسبه کنیم در روش KNN میگوییم از این ده تا همسایه چند تا کلاس آبی است (مثلاً 6 تا) پس 0/6 احتمال این است که کلاس مشاهده جدید آبی باشد.

- Training and Test error rates:



در واقع خطای تست هیچوقت کمتر از Bayes نمیشه و برای بی نهایت K خیلی به آن نزدیک میشه.

Subject:

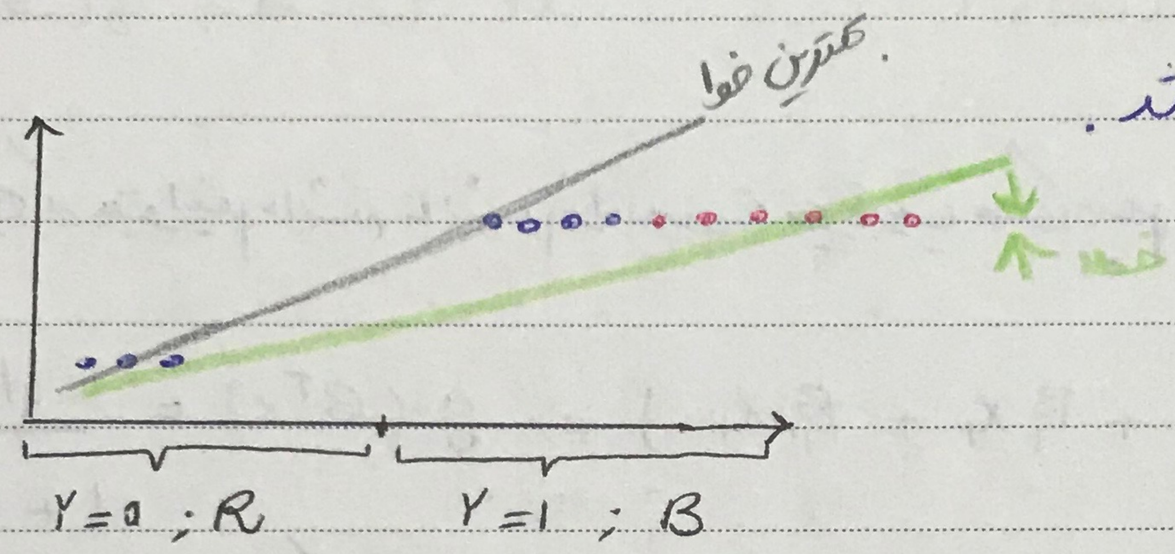
Year. Month. Date. ( )

Regression for classification

یکی از مسائلی که مطرح میشه این است که چرا از خود linear reg برای classification استفاده نکنیم.

ایده این است که ابتدا یک linear reg اعمال کنیم و سپس براساس اینکه $\hat{Y} > 0/5$ یا $\hat{Y} < 0/5$ است

تصمیم بگیریم در کدام کلاس باشد.



مشکلی که این روش دارد این است که برخی تخمین های ما خارج از فضای [0,1] است و همین طور ممکنه خطای آن خیلی زیاد شه.

روشی که مبتنی بر رگرسیون برای classification وجود دارد اسمش logestic Reg است که برای binary استفاده میشه و ایده آن این است که متغیر response را مجبور میکنه در محدوده [0,1] باشد.

فرض کنیم مدل خطی زیر را داریم که براساس شیوه هایی تصمیم بگیریم فرد بعد از دو سال سرطان دارد یا خیر؟

$$Y = \beta_0 + \underbrace{\beta_1 X_1}_{\text{سن}} + \underbrace{\beta_2 X_2}_{\text{جنسیت}} + \underbrace{\beta_3 X_3}_{\text{میان تن}}$$

خروجی مدل به ازای X_1، X_2 و X_3 را میگیریم و تابعی روش اعمال میکنیم که خروجی مدل را به بازه صفر و یک ببره

Subject:

Year. Month. Date. ( )

تعریف ریاضی تابعی که برای logestic Reg استفاده میشه به صورت زیر است و به کل تابع sigmoid یا logestic می‌گوییم:

$$g(z) = \frac{1}{1+e^{-z}}$$

if $z=0 \Rightarrow g(0) = \frac{1}{2}$

if $z=+\infty \Rightarrow g(+\infty) = 1$ if $z=-\infty \Rightarrow g(-\infty) = 0$

⇐ فرم پارامتری که میتوانیم داشته باشیم را بصورت زیر تعریف می‌کنیم:

$$f(x) = g(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3) \Rightarrow g(\beta^T x) = \frac{1}{1+e^{-\beta^T x}}$$

یک مزیت این روش این است که قابل تفسیر است برای مثال وقتی βها را تخمین بزنیم و $\beta_1 > 0$ باشد یعنی هرچه سن فرد بیشتر میشه احتمال اینکه فرد بعد از دو سال سرطان گرفته بیشتر میشود. و مزیت دیگر هم این است که وقتی $p \ll n$ است مشکلات overfiting نداریم.

حال سوال این است که چطور β را تخمین بزنیم که با دیتایی که داریم همخوانی بیشتری داشته باشد.

$$D = \{(x[1], y[1]), (x[2], y[2]), \dots, (x[n], y[n])\}$$

$x[n] \in \mathbb{R}^p$ ، $y[n] \to (0,1)$

$$Y \sim \text{Bernouli}(g(\beta^T x))$$

توزیع برنولی : $P(Y=H) = \theta$; $P(Y=T) = (1-\theta)$

$$g(\beta^T x) = \frac{1}{1+e^{-(\beta_0 + \beta_1 x_1[i] + \dots + \beta_p x_p[i])}} \quad ; i = 1, \dots, n = \alpha[i]$$

Sunwood

Subject:

Year. Month. Date. ( )

$$MLE: \hat{\beta}_{MLE} = \underset{\beta}{\operatorname{argmax}}\ L(\beta)$$

$$L(\beta) = P(y[1], \dots, y[n] \mid x[1], \dots, x[n], \beta)$$

iid

$$\prod_{i=1}^{n} P(y[i] \mid x[i], \beta) \xrightarrow{\text{توزیع برنولی}} \begin{cases} P(y[i]=1 \mid x[i], \beta) = g(\beta^T x) \\ P(y[i]=0 \mid x[i], \beta) = 1 - g(\beta^T x) \end{cases}$$

notation := $\alpha[i] = g(\beta^T x[i])$

$$\Rightarrow L(\beta) = \prod_{i=1}^{n} (\alpha[i])^{y[i]} (1-\alpha[i])^{1-y[i]}$$

$$\Rightarrow \ell(\beta) = \sum_{i=1}^{n} \underbrace{\boxed{y[i] \log \alpha[i] + (1-y[i]) \log(1-\alpha[i])}}_{E[i]}$$

حال برای پیدا کردن βها باید مشتق بگیریم و مساوی صفر قرار دهیم اما ثابت شده که به صورت تئوری جوابی برای βها پیدا نمی‌کنیم و باید از روش‌های مبتنی بر gradient محاسبه شود بنابراین از gradient ascent استفاده می‌کنیم.

"Gradient ascent"

$$\beta^{new} := \beta + \lambda \nabla_{\beta} \ell(\beta)$$

learning rate

$$\frac{\partial \ell(\beta)}{\partial \beta_j} = \sum_{i=1}^{n} \frac{\partial E[i]}{\partial \beta_j} \xrightarrow{\text{قاعده زنجیری}} \sum_{i=1}^{n} \frac{\partial E[i]}{\partial \alpha[i]} \times \frac{\partial \alpha[i]}{\partial z[i]} \times \frac{\partial z[i]}{\partial \beta_j}$$

$$\alpha[i] = \frac{1}{1 + e^{\underbrace{-\beta^T x[i]}_{z[i]}}}$$

Sunwood

Subject:

Year. Month. Date. ( )

$$\frac{\partial z[i]}{\partial B_j} = \frac{\partial (B_1 x_1[i] + B_2 x_2[i] + \cdots + B_p x_p[i])}{\partial B_j} = x_j[i]$$

$$\frac{\partial \alpha[i]}{\partial z[i]} = \frac{e^{z[i]}}{(1+e^{z[i]})^2} = \alpha[i](1-\alpha[i])$$

$$\frac{\partial E[i]}{\partial \alpha[i]} = \frac{y[i]}{\alpha[i]} - \frac{(1-y[i])}{1-\alpha[i]}$$

$$\Rightarrow \frac{\partial \ell(B)}{\partial B_j} \left(\frac{\overbrace{y[i](1-\alpha[i]) - (1-y[i])\alpha[i]}^{y[i] - \cancel{\alpha[i]y[i]} - \alpha[i] + \cancel{\alpha[i]y[i]}}}{\cancel{\alpha[i](1-\alpha[i])}} \right) \times \cancel{\alpha[i](1-\alpha[i])} \times x_j[i]$$

$$= \sum_{i=1}^{n} (y[i] - \alpha[i])\, x_j[i] \quad \blacksquare$$

حال عزم vector را مینویسیم :

$$\alpha = \begin{bmatrix} \alpha[1] \\ \vdots \\ \alpha[n] \end{bmatrix} \qquad y = \begin{bmatrix} y[1] \\ \vdots \\ y[n] \end{bmatrix}$$

$$X = \begin{bmatrix} x_1[1] & \cdots & x_p[1] \\ & & \\ x_1[n] & \cdots & x_p[n] \end{bmatrix} \qquad \Rightarrow \nabla \ell(B) = X^T (y - \alpha)$$

gradient ascent algo :

$$B^{new} := B + \overset{\text{learning rate}}{\lambda} (X^T (y - \alpha))$$

$$H = \frac{\nabla^2 \ell(B)}{\partial B^2} = - X^T \overset{\text{ماتریس قطری}}{w} X \qquad \begin{bmatrix} w_{11} & & 0 \\ & w_{22} & \\ 0 & & \ddots \; w_{nn} \end{bmatrix} \qquad w_{ii} = \alpha[i] \times (1-\alpha[i])$$

Subject:

Year. Month. Date. ( )

همین ما یک ماتریس نیمه معین منفی است (یعنی دنبال ماکسیمم هستیم)

Newton Method :

← یک روش مرتبه دوم است برای پیدا کردن صفرهای یک تابع.

ابتدا این متد را معرفی می‌کنیم :

فرض کنیم تابع f روی متغیر x را بصورت $f(x): \mathbb{R} \to \mathbb{R}$ داریم و هدف پیدا کردن صفر تابع است. یعنی می‌خواهیم $f(x) = 0$ را بیابیم و این روش هم مانند روش‌های gradient کاری است و بدنبال بهروزرسانی آن به شکل زیر است :

$$x^{new} := x - \frac{f(x)}{f'(x)}$$

Example. 1 :

ریشهٔ تابع زیر را با استفاده از روش نیوتن پیدا کنید. $(x^0 = 4)$

$$f(x) = x^3 - 8$$

$$x^1 := x^0 - \frac{x_0^3 - 8}{3x_0^2} \Rightarrow 4 - \frac{56}{3(4)^2} = 2.83$$

$$x^2 := x^1 - \frac{x_1^3 - 8}{3x_0^2} = 2.22 \quad ; x^3 = 2.02 \quad ; x^4 = 2.00223$$

$$x^5 = 2$$

سرعت همگرایی این روش بیشتر از نیوتن است.

• کاری که این الگوریتم میکند به این صورت است که در هر قطعه تابع را با یک خط تقریب میزند و در این خط تقریب

Sunwood

Subject:

Year. Month. Date. ( )

زده شده نگاه کنید که کجا و در چه نقطه‌ای $y=0$ شده و آن نقطه میشود نقطه‌ی مرحله بعدی در روش نیوتن

همانطور که میدانیم فرمول مشتق تابع در یک نقطه به صورت زیر است:

$$f'(x_0) = \lim_{x \to x_0} \frac{f(x) - f(x_0)}{x - x_0}$$

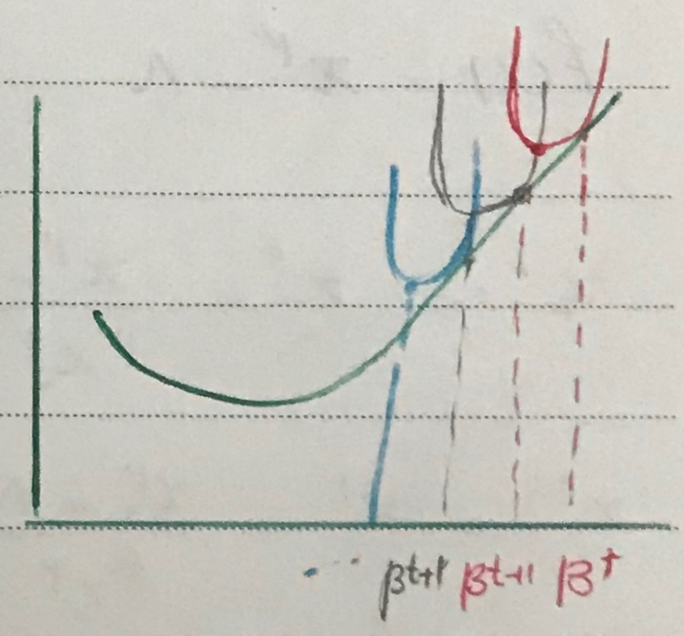
یعنی وقتی x خیلی خیلی نزدیک x_0 باشد تقریب مشتق است؛ حال داریم:

$$f'(x_0) \approx \frac{\cancel{f(x)}^{\,0} - f(x_0)}{x - x_0} \Rightarrow (x - x_0) = -\frac{f(x_0)}{f'(x)}$$

$$\Rightarrow x = x_0 - \frac{f(x_0)}{f'(x_0)}$$ تعریف تابع نیوتن است.

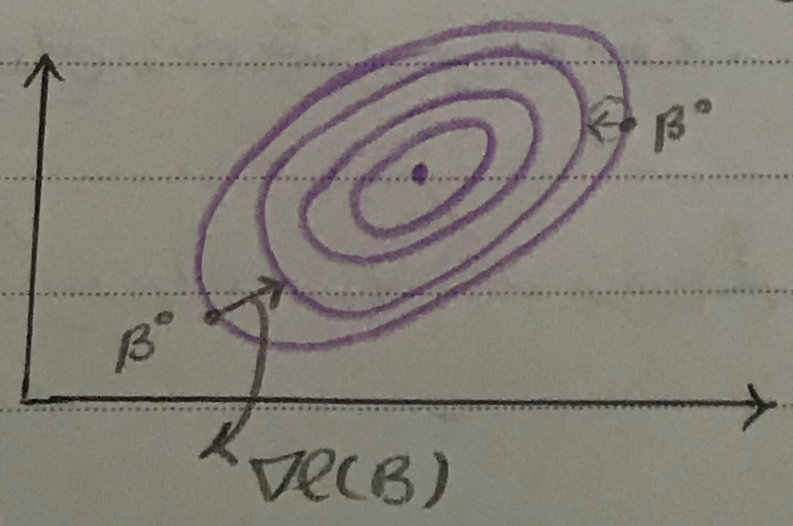
حال در مورد تابع likelihood هدف این است که $l(B)$ را با استفاده از متد نیوتن ماکسیمم کنیم یعنی در univariate case هدف پیدا کردن نقطه‌ای است که $l'(B)=0$ است

فرمول نیوتن $\Rightarrow$ $$B^{new} := B - \frac{l'(B)}{l''(B)}$$



در واقع در هر نقطه چون بدنبال مینیمم نقطه هستیم یک تابع درجه دوم تقریب میزنیم و نقطه مینیمم آن تابع میشود نقطه شروع بعدی در روش نیوتن

در فضای دو بعدی:



Sunwood

Subject:

Year. Month. Date. ( )

idea: make a second order approximation and minimize it.

Taylor's theorem:

$$f(x) \approx f(a) + (x-a)\nabla f(x) + \frac{1}{2}(x-a)^T H (x-a)$$

نقطه‌ای که می‌خواهیم ازش تقریب بزنیم

همسایگی از نقطه a است.

بنابراین اگر گرادیان و هسین را داشته باشیم می‌دانیم اطراف a شکل تابع به چه صورتی است. و شرط مورد نیاز هم این است که تابع تا حدی که لازم داریم هموار باشد و بتوانیم مشتق اول و دوم آن را بدست آوریم.

اگر سطح تیلور را ساده کنیم به فرم زیر می‌رسیم:

$$q(x) = \frac{1}{2} x^T H x + b^T x + c$$

$$b = \nabla f - Ha$$

$$\nabla q(x) = 0 \Rightarrow x = a - H^{-1}\nabla f$$

یعنی: $\beta^{new} := \beta - H^{-1}\nabla \ell(\beta)$

Example:

ماتریس هسین روش logestic Reg:

$$H = -(X^T W X) \longrightarrow W = \begin{bmatrix} w_{11} & & 0 \\ & w_{22} & \\ 0 & & \ddots \; w_{nn} \end{bmatrix} \quad , w_{ii} = \alpha[i](1-\alpha[i])$$

$$\nabla L(\beta) = X^T(y-\alpha)$$

روش نیوتن: $\beta^{new} := \beta + (X^T W X)^{-1} X^T (y-\alpha)$

Subject:

Year. Month. Date. ( )

iteratively reweighted least square:

فرمول که داریم را به تغییراتی میدهیم:

$$\underbrace{\overbrace{(X^T W X)^{-1}(X^T W X)}^{F}}_{I}\beta + \overbrace{(X^T W X)^{-1} X^T \underbrace{(W W^{-1})}_{I}}^{F}(y-\alpha)$$

$$= (X^T W X)^{-1} X^T W \overbrace{(X\beta + W^{-1}(y-\alpha))}^{Z}$$

$$= (X^T W X)^{-1} X^T W Z \rightarrow \text{adjusted response}$$

در واقع آنلاد همان مسئله LS است ولی با X و Z جدید

$$\beta^{new} = \operatorname{argmin} (Z - X\beta)^T W (Z - X\beta)$$

Regularized logestic Regression:

$$\hat{\beta} = \operatorname{argmin} \left(\sum_{i=1}^{n} y[i] \log \alpha[i] + (1-y[i]) \log(1-\alpha[i]) + \lambda \sum_{i=1}^{p} |\beta_i| \right.$$

میتوانیم بصورت تکراری یک لاسو را حل کنیم.

Subject:

Year. Month. Date. ( )

Exponential Family:

یک مجموعه مدل داریم که مدل‌های reg , logestic reg هم جزء GLM هستند.

تعریف: یک Exponential Family یک مجموعه توزیع است که چهار مولفه دارد و به فرم زیر است:

$$P_\theta(x) = \frac{1}{Z(\theta)} \underbrace{h(x)} \exp\left(t(\theta)^T s(x)\right)$$

sufficient statistic

partition funtion

for supporting and scaling

Example: $X \sim Exp(\lambda)$

$$\Rightarrow P_\lambda(x) = \lambda \exp(-\lambda x)\, I(x \geq 0)$$

$$Z(\lambda) = \frac{1}{\lambda} \quad ; \quad h(x) = I(x \geq 0)$$

$$t(\lambda) = -\lambda \quad ; \quad s(x) = x$$

Example: Gaussian distribution

$\sigma^2 = 1$

$$p(x) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu)^2\right)$$

$$p(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(x^2 - 2x\mu + \mu^2)\right) = \underbrace{\frac{1}{\sqrt{2\pi}} \exp\left(\frac{-1}{2}\mu^2\right)}_{Z(\mu)} \times$$

$$\underbrace{\exp(+x\mu)}_{\substack{t(\mu)=\mu \\ s(x)=x}} \times \underbrace{\exp\left(-\frac{x^2}{2}\right)}_{h(x)}$$