دانشکده‌ی مهندسی کامپیوتر

یادگیری ماشین بهار ۱۴۰۰

پاسخ تمرین سری ششم

مدرس: دکتر محمّدحسین رهبان

پاسخ سوال ۱

آ) اصل *Occam's Razor*:

The simplest model that fits the data is also the most plausible.

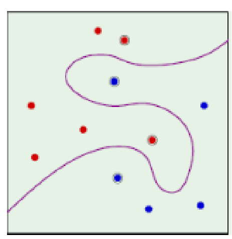
اصل *Sampling Bias*:

If the data is sampled in a biased way , learning will produce a similarly biased outcome.

اصل *Data Snooping*:

If a data set has affected any step in the learning process , its ability to assess the outcome hase been compromised .

ب) خیر. به عنوان مثال شکل زیر را در نظر بگیرید. با اینکه ظاهر مرز جدا کننده دو کلاس پیچیده به نظر می‌رسد ولی تعداد کمی *support vector* دارد.



پ) *Sample bias caused by Snooping*. به عنوان نمونه به مثال اسلاید مراجعه کنید.

پاسخ سوال ۲

آ)

$$h_{\boldsymbol{\theta}}(\mathbf{x}) = \boldsymbol{\theta}^\top \mathbf{x}$$

$$\min_{\boldsymbol{\theta}} \sum_{j=1}^{m} w^j \left[h_{\boldsymbol{\theta}}(\mathbf{x}^j) - y^j\right]^2$$

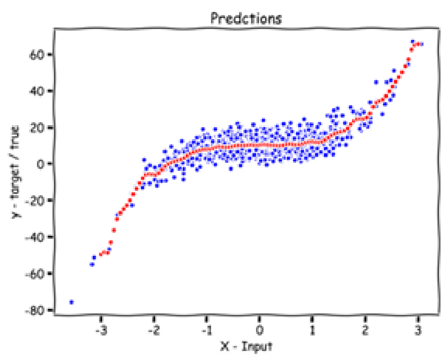
$$\sum_{j=1}^{m} w^j \left[h_{\boldsymbol{\theta}}(\mathbf{x}^j) - y^j\right]^2 = (X\boldsymbol{\theta} - Y)^\top W (X\boldsymbol{\theta} - Y) \qquad W = \begin{bmatrix} w^1 & 0 & 0 \\ 0 & w^2 & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & w^m \end{bmatrix}$$

$$\mathcal{J}(\boldsymbol{\theta}) = \frac{1}{2}\sum_{j=1}^{m} w^j \left[h_{\boldsymbol{\theta}}(\mathbf{x}^j) - y^j\right]^2 = \frac{1}{2}(X\boldsymbol{\theta} - Y)^\top W (X\boldsymbol{\theta} - Y)$$

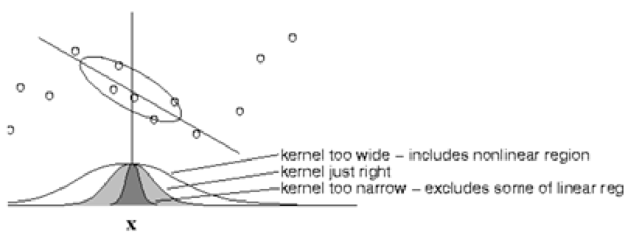
$$\begin{aligned}
\frac{\partial}{\partial \boldsymbol{\theta}}\mathcal{J}(\boldsymbol{\theta}) &= \frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}}(X\boldsymbol{\theta} - Y)^\top W (X\boldsymbol{\theta} - Y) \\
&= \frac{1}{2}\frac{\partial}{\partial \boldsymbol{\theta}}\left(\boldsymbol{\theta}^\top X^\top W X\boldsymbol{\theta} - \boldsymbol{\theta}^\top X^\top W Y - Y^\top W X \boldsymbol{\theta} + Y^\top W Y\right) \qquad (W^\top = W) \\
&= \left(X^\top W X \boldsymbol{\theta} - X^\top W Y\right)
\end{aligned}$$

$$\frac{\partial}{\partial \boldsymbol{\theta}}\mathcal{J}(\boldsymbol{\theta}) = 0 \implies \left(X^\top W X\boldsymbol{\theta} - X^\top W Y\right) = 0 \implies \boldsymbol{\theta} = \left(X^\top W X\right)^{-1} X^\top W Y$$

ب) بدون وزن‌های *w*، رابطه بخش قبل یک *linear regression* است. با اضافه کردن وزن‌ها که تابعی از داده تست *t* هستند(یعنی *w(t)*)، می‌توان نزدیک هر داده آموزش، یک هسته در نظر گرفت و این باعث میشود از حالت *linear* بودن خارج شده و روی منحنی داده‌ها فیت شود؛ مانند تصویر زیر:



همچنین در شکل زیر تاثیر واریانس هسته قابل مشاهده است:



درضمن هیچ منعی برای یکسان بودن واریانس این هسته‌ها وجود ندارد و واریانس هسته‌های مرتبط با زیر مجموعه‌ای از داده‌ها می‌توانند یکسان باشند یا حتی همه هسته‌ها واریانس‌های متفاوت با یکدیگر داشته باشند. بسته به اینکه داده‌ها چه منحنی‌ای می‌سازند(چه مقدار به *locally linear* بودن نزدیک یا دور است)، می‌توان دسته‌های گوناگونی را برای واریانس هسته‌ها در نظر گرفت.

پاسخ سوال ۳

آ) در ادامه مراحل این الگوریتم را برای *Regression* مشاهده می‌کنید:

Input: *training set* $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, *a differentiable loss function* $L(y, F(\mathbf{x}))$, *number of iterations* M.

Algorithm:

1. *Initialize model with a constant value*

$$F_0(\mathbf{x}) = \underset{\gamma}{\boldsymbol{argmin}} \sum_{i=1}^{n} L\left(y_i, \gamma\right)$$

2. *For $m = 1$ to M:*

    1. *Compute so-called pseudo-residuals.*

    $$r_{im} = -\left[\frac{\partial L\left(y_i, F\left(\mathbf{x}_i\right)\right)}{\partial F\left(\mathbf{x}_i\right)}\right]_{F(\mathbf{x})=F_{m-1}(\mathbf{x})} \quad \text{for } i = 1, \ldots, n$$

    2. *Fit a base learner (or weak learner, e.g. tree) $h_m(\mathbf{x})$ to pseudo-residuals, i.e. train it using the training set $\{(\mathbf{x}_i, r_{im})\}_{i=1}^{n}$*

    3. *Compute multiplier γ_m by solving the following one-dimensional optimization problem:*

    $$\gamma_m = \underset{\gamma}{\boldsymbol{argmin}} \sum_{i=1}^{n} L\left(y_i, F_{m-1}\left(\mathbf{x}_i\right) + \gamma h_m\left(\mathbf{x}_i\right)\right)$$

    4. *Update the model:*

    $$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \gamma_m h_m(\mathbf{x})$$

3. ***Output*** $F_M(\mathbf{x})$.

همچنین برای *classification* داریم:

Input: *Data $\{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, and a differentiable Loss Function $L\left(y_i, F(\mathbf{x})\right)$*

Step 1: *Initialize model with a constant value:*

$$F_0(\mathbf{x}) = \boldsymbol{argmin} \sum_{i=1}^{n} L\left(y_i, \gamma\right)$$

Step 2: *for $m = 1$ to M :*

1. *Compute*

$$r_{im} = -\left[\frac{\partial L\left(y_i, F\left(\mathbf{x}_i\right)\right)}{\partial F\left(\mathbf{x}_i\right)}\right]_{F(\mathbf{x})=F_{m-1}(\mathbf{x})} \quad \text{for } i = 1, \ldots, n$$

2. *Fit a regression tree to the r_{im} values and create terminal regions R_{jm}, for $j = 1 \ldots J_m$*

3. *For $j = 1 \ldots J_m$ compute*

$$\gamma_{jm} = \underset{\gamma}{\boldsymbol{argmin}} \sum_{\mathbf{x}_i \in R_{ij}} L\left(y_i, F_{m-1}\left(\mathbf{x}_i\right) + \gamma\right)$$

4. *Update*

$$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \nu \sum_{j=1}^{J_m} \gamma_{jm} \mathbb{1}_{\mathbf{x} \in R_{jm}}$$

Step 3: ***Output*** $F_M(\mathbf{x})$

تفاوت عمده این دو روش در تعریف *loss function* و محاسبه ضریب لاندا برای بروزرسانی مدل است.

ب) مراحل حل را بصورت زیر می‌توان دنبال کرد:

$$\text{هدف}: \sum_{i=1}^{n} \frac{\partial}{\partial r} L(y_i, \underbrace{F_{m-1}(\mathbf{x}_i) + \gamma h_m(\mathbf{x}_i)}_{g_{mi}(\gamma)}) = 0 \qquad (I)$$

$$p_i \triangleq \sigma(g_{mi}(\gamma)); \quad \sigma(x) = \frac{1}{1+e^{-x}} = \textit{sigmoid function}$$

$$L(y_i, g_{mi}(\gamma)) = -[y_i \log p_i + (1-y_i)\log(1-p_i)]$$

$$\frac{\partial}{\partial \gamma} L(y_i, g_{mi}(\gamma)) = -\left[\frac{y_i}{p_i} - \frac{1-y_i}{1-p_i}\right]\frac{\partial p_i}{\partial \gamma}, \quad \frac{\partial p_i}{\partial \gamma} = p_i(1-p_i)h_m(\mathbf{x}_i)$$

$$\Longrightarrow \frac{\partial}{\partial \gamma} L(y_i, g_{mi}(\gamma)) = -(y_i - p_i)h_m(\mathbf{x}_i) \qquad (II)$$

$$(I),(II): \sum_{i=1}^{n} -\left(y_i - \frac{1}{1+e^{-g_{mi}(\gamma)}}\right) h_m(\mathbf{x}_i) = 0$$

This equation is hard to solve, so we use taylor expansion for its approximation:

$$L(x, y+\gamma) \simeq L(x,y) + \frac{d}{dy}L(x,y)\gamma + \frac{d^2}{dy^2}\frac{L(x,y)}{2}\gamma^2 \Longrightarrow$$

$$\sum_{i=1}^{n} L(y_i, g_{mi}(\gamma)) \approx \sum_{i=1}^{n}\left[L(y_i, g_{mi}(0)) + \frac{\partial}{\partial g_{mi}(\mathbf{x}_i)} L(y_i, g_{mi}(0))\, h_m(\mathbf{x}_i)\,\gamma + \frac{1}{2}\frac{\partial^2}{\partial g_{mi}^2(\mathbf{x}_i)} L(y_i, g_{mi}(0))\, h_m^2(\mathbf{x}_i)\,\gamma^2\right] \Longrightarrow$$

$$\gamma^* = -\frac{\sum_{i=1}^{n} \frac{\partial}{\partial g_{mi}(\mathbf{x}_i)} L(y_i, g_{mi}(0))\, h_m(\mathbf{x}_i)}{\sum_{i=1}^{n} \frac{1}{2}\frac{\partial^2}{\partial g_{mi}^2(\mathbf{x}_i)} L(y_i, g_{mi}(0))\, h_m^2(\mathbf{x}_i)} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} (\mathrm{y}_i - p_i)\, h_m(\mathbf{x}_i)}{\sum_{\mathrm{i}=1}^{\mathrm{n}} p_i(1-p_i)\, h_m^2(\mathbf{x}_i)} \qquad (*)$$

برای مطالعه بیشتر به این آدرس مراجعه کنید.

پ) با در نظر گرفتن روند الگوریتم *classification* که در بخش آ آورده شده است، در این مساله مراحل زیر را دنبال می‌کنیم:

$$\textbf{\textit{Step1}}: F_0(\mathbf{x}) = \underset{\gamma}{\textbf{\textit{argmin}}} \sum_{i=1}^{n} L(y_i, \gamma)$$

$$(II): g_{mi}(\gamma) \triangleq \gamma, \quad \sum_{i=1}^{n} -(y_i - \sigma(\gamma)) = 0 \Longrightarrow \sigma(\gamma) = \frac{\sum_{i=1}^{n} y_i}{n} = \sigma(F_0)$$

$$\left.\begin{array}{l} No \equiv 0 \\ Yes \equiv 1 \end{array}\right\} \Longrightarrow \sigma(F_0) = \frac{4}{6} \simeq 0.7$$

Step2 :

1.*Residual computation:*

$$(II): \left.\begin{array}{l} \textbf{\textit{Res}}_{\textbf{\textit{im}}} = y_i - \sigma(F_{m-1}) \\ m = 1 \Longrightarrow \sigma(F_0) \simeq 0.7 \end{array}\right\} \Longrightarrow \textit{compute it for eatch data:}$$

i	1	2	3	4	5	6
Res_im	0.3	0.3	−0.7	−0.7	0.3	0.3

2.*Quantize Residual with a threshold:*

$$th = 0 : \begin{cases} if \quad \boldsymbol{Res_{im}} \geqslant th \quad then \quad label = 1 \\ if \quad \boldsymbol{Res_{im}} < th \quad then \quad label = 0 \end{cases} \Longrightarrow \text{compute it for eatch data:}$$

i	1	2	3	4	5	6
label	1	1	0	0	1	1

now with this new labels, Train SVM: 1,2,3,5,6 assign to class of 1 and 4 assign to class of 0

3.*Computation of* γ_0 *for class 0 (i.e. data: 1,2,3,5,6) and* γ_1 *for class of 1 (i.e. data: 4) with* $(*)$ *formula:*

$$\gamma_1 = \frac{0.3 + 0.3 - 0.7 - 0.7 + 0.3 + 0.3}{0.7 \times (1 - 0.7) \times 5} = \frac{0.5}{1.05} = 0.48$$

$$\gamma_0 = \frac{-0.7}{0.7 \times (1 - 0.7)} = -3.33$$

4.*Update the model:*

ν *(learning rate* $= 0.1), m = 1$

$$F_m(\mathbf{x}) = \underbrace{F_{m-1}(\mathbf{x})}_{\sigma^{-1}(0.7) \simeq 0.7} + \underbrace{\nu}_{0.1} \sum_{j=0}^{1} \gamma_{jm} \mathbb{1}_{\mathbf{x} \in R_{jm}}$$

در آخرین عبارت، یعنی وقتی داده‌ای وارد شد، اگر مدل آن را در دسته ۰ قرار داد از γ_0 و در غیر این حالت از γ_1 استفاده می‌شود.

ت) برای مشاهده حل یک نمونه دیگر می‌توانید به این آدرس مراجعه کنید. اگر پس از مشاهده نمونه ارایه شده، پرسشی باقی ماند، در این بخش از دستیار محترم بپرسید.

پاینده باشید