$\boxed{1.1}$ g is a valid kernel

$$h(x,y) = \frac{1}{4}\Big(g(x+y,\, x+y) - g(x-y,\, x-y)\Big) \overset{*}{\underset{**}{=}} \frac{1}{4}\Big(4\, g(x,y)\Big) = g(x,y)$$

$$g(x+y,\, x+y) = \underbrace{g(x,\, x+y)}_{g(x,x)+g(y,x)} + \underbrace{g(y,\, x+y)}_{g(x,y)+g(y,y)} = g(x,x)+g(y,y)+2g(x,y) \quad (*)$$

$$g(x-y,\, x-y) = \underbrace{g(x,\, x-y)}_{g(x,x)-g(y,x)} - \underbrace{g(y,\, x-y)}_{g(x,y)-g(y,y)} = g(x,x)+g(y,y)-2g(x,y) \quad (**)$$

بنابراین نشان دادیم $h(x,y) = g(x,y)$ پس h هم یک کرنل معتبر است

$\boxed{1.2.a}$

یک راه برای اثبات معتبر بودن k_3 آن است که نشان دهیم ϕ_3 ای وجود دارد که $k_3(x_1,x_2) = \langle \phi_3(x_1),\, \phi_3(x_2)\rangle$

$$k_1 \text{ is Valid} \longrightarrow k_1(x_1,x_2) = \langle \phi_1(x_1),\, \phi_1(x_2)\rangle = \begin{bmatrix}\varphi_1(x_1)\\ \vdots \\ \varphi_m(x_1)\end{bmatrix} \cdot \begin{bmatrix}\varphi_1(x_2)\\ \vdots \\ \varphi_m(x_2)\end{bmatrix} = \sum_{i=1}^{m} \varphi_i(x_1)\,\varphi_i(x_2)$$

$$k_2 \text{ is valid} \longrightarrow k_2(x_1,x_2) = \langle \phi_2(x_1),\, \phi_2(x_2)\rangle = \begin{bmatrix}\varphi'_1(x_1)\\ \vdots \\ \varphi'_{m'}(x_1)\end{bmatrix} \cdot \begin{bmatrix}\varphi'_1(x_2)\\ \vdots \\ \varphi'_{m'}(x_2)\end{bmatrix} = \sum_{i=1}^{m'} \varphi'_i(x_1)\,\varphi'_i(x_2)$$

حال ϕ_3 را به صورت زیر تعریف می‌کنیم

$$\phi_3(x) = \begin{bmatrix}\varphi_1(x)\\ \vdots \\ \varphi_m(x)\\ \varphi'_1(x)\\ \vdots \\ \varphi'_{m'}(x)\end{bmatrix} \longrightarrow k_3(x_1,x_2) = \sum_{i=1}^{m} \varphi_i(x_1)\,\varphi_i(x_2) + \sum_{i=1}^{m'} \varphi'_i(x_1)\,\varphi'_i(x_2)$$

$$= k_1(x_1,x_2) + k_2(x_1,x_2)$$

بنابراین k_3 معتبر است

$\boxed{1.2.b}$

مانند سوال قبل نشان می‌دهیم Φ_4 ای وجود دارد که $k_4(x_1,x_2) = \langle \Phi_4(x_1), \Phi_4(x_2) \rangle$

$$k_1 \text{ is valid} \longrightarrow k_1(x_1,x_2) = \cdots = \sum_{i=1}^{m} \varphi_i(x_1)\,\varphi_i(x_2)$$

$$k_2 \text{ is valid} \longrightarrow k_2(x_1,x_2) = \cdots = \sum_{i=1}^{m'} \varphi'_i(x_1)\,\varphi'_i(x_2)$$

$$k_4(x_1,x_2) = k_1(x_1,x_2)\,k_2(x_1,x_2) = \sum_{i=1}^{m}\sum_{j=1}^{m'} \underline{\varphi_i(x_1)}\;\underline{\underline{\varphi_i(x_2)}}\;\underline{\varphi'_j(x_1)}\;\underline{\underline{\varphi'_j(x_2)}}$$

حال Φ_4 را به صورت مقابل تعریف می‌کنیم

$$\Phi_4(x) = \begin{bmatrix} \varphi''_{11}(x) \\ \vdots \\ \varphi''_{mm'}(x) \end{bmatrix} \qquad \varphi''_{ij}(x) = \varphi_i(x)\,\varphi'_j(x)$$

$$\longrightarrow k_4(x_1,x_2) = \sum_{i=1}^{m}\sum_{j=1}^{m'} \varphi''_{ij}(x_1)\,\varphi''_{ij}(x_2) = \langle \Phi_4(x_1), \Phi_4(x_2) \rangle$$

بنابراین k_4 معتبر است

$\boxed{1.2.c}$

اگر بسط تیلور $e^{k_1(x_1,x_2)}$ را در نقطه 0 بنویسیم داریم

$$e^{k_1(x_1,x_2)} = \lim_{N\to\infty} \sum_{i=0}^{N} \frac{k_1(x_1,x_2)^i}{i!} = \sum_{i=0}^{\infty} \frac{k_1(x_1,x_2)^i}{i!}$$

⊛ طبق بخش $\boxed{1.2.b}$ می‌دانیم که ضرب 2 کرنل معتبر همچنان معتبر است پس این کار را می‌توان به تعداد دلخواه انجام داد، یعنی گام استقرا آن است که

$k^n \times k$ خود حاصل ضرب 2 کرنل معتبر است پس $k_1(x_1,x_2)^i$ کرنل معتبر است

⊛ همچنین ضرب عدد مثبت در کرنل معتبر نیز یک کرنل معتبر است زیرا :

$$k \text{ is valid} \longrightarrow k(x_1,x_2) = \sum_{i=1}^{m} \varphi_i(x_1)\,\varphi_i(x_2) \longrightarrow \forall \lambda > 0 \quad \lambda k(x_1,x_2) = \sum_{i=1}^{m} \underbrace{\sqrt{\lambda}\,\varphi_i(x_1)}_{\varphi'_i(x_1)}\;\underbrace{\sqrt{\lambda}\,\varphi_i(x_2)}_{\varphi'_i(x_2)}$$

⊛ طبق بخش $\boxed{1.2.a}$ جمع کرنل های معتبر نیز معتبر است

$$\longrightarrow e^{k_1(x_1,x_2)} \text{ is valid}$$

$\boxed{1.2.d}$

مشابه سوال قبل بسط تیلور $\frac{1}{1-x_1^T x_2}$ را در نقطه 0 می‌نویسیم

$$\frac{1}{1-x_1^T x_2} = \lim_{N\to\infty} \sum_{i=0}^{N} (x_1^T x_2)^i = \lim_{N\to\infty} \frac{1-(x_1^T x_2)^{N+1}}{1-x_1^T x_2}$$

یعنی اگر بخواهیم به $\frac{1}{1-x_1^T x_2}$ برسیم باید شرط تصاعد هندسی را داشته باشیم.

$$\text{if } |x_1^T x_2| < 1 \longrightarrow \frac{1}{1-x_1^T x_2} = \sum_{i=0}^{\infty} (x_1^T x_2)^i \longrightarrow k_6 \text{ is valid}$$

اگر این شرط را نداشته باشیم لزوماً k_6 معتبر نیست. به عنوان مثال، مثال نقض زیر را در نظر بگیرید.

2 samples in $\mathbb{R}^3$ $\quad x_1 = \begin{bmatrix} 1 \\ 3 \\ 5 \end{bmatrix}, \quad x_2 = \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix}$

$$\longrightarrow \text{kernel matrix} = \begin{bmatrix} \frac{1}{1-35} & \frac{1}{1-44} \\ \frac{1}{1-44} & \frac{1}{1-56} \end{bmatrix} \longrightarrow \exists x = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad [1 \;\; 1] \begin{bmatrix} \frac{-1}{34} & \frac{-1}{43} \\ \frac{-1}{43} & \frac{-1}{56} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = -\frac{1}{34} - \frac{2}{43} - \frac{1}{56} < 0$$

k_6 is not valid $\longleftarrow$ kernel matrix is not positive semi definite

---

$\boxed{1.3}$

$$k \text{ is valid} \longrightarrow k(x, x') = \phi(x) \cdot \phi(x')$$

$$\hat{k}(x^{(i)}, y^{(i)}) = \sum_{x^{(i)} \in A} \sum_{y^{(i)} \in B} \phi(x^{(i)}) \cdot \phi(y^{(i)}) = \sum_{x^{(i)} \in A} \sum_{y^{(i)} \in B} \sum_{j=1}^{d} \varphi(x_j^{(i)}) \varphi(y_j^{(i)}) = ⊛$$

$$\underbrace{\sum_{x^{(i)} \in A} \phi(x^{(i)})}_{\hat{\phi}(x^{(i)})} \cdot \underbrace{\sum_{y^{(i)} \in B} \phi(y^{(i)})}_{\hat{\phi}(y^{(i)})} = \begin{bmatrix} \sum_{x^{(i)} \in A} \varphi(x_1^{(i)}) \\ \vdots \\ \sum_{x^{(i)} \in A} \varphi(x_d^{(i)}) \end{bmatrix} \cdot \begin{bmatrix} \sum_{y^{(i)} \in B} \varphi(y_1^{(i)}) \\ \vdots \\ \sum_{y^{(i)} \in B} \varphi(y_d^{(i)}) \end{bmatrix} = ⊛$$

بنابراین نشان دادیم $\hat{k}(x^{(i)}, y^{(i)}) = \hat{\phi}(x^{(i)}) \cdot \hat{\phi}(y^{(i)})$ پس $\hat{k}$ کرنل معتبر است

2.1

پس از به دست آمدن α بهینه توسط QP برای به دست آوردن b داشتیم :

به ازای هر $x^{(n)}$ که SV باشد $\quad y^{(n)}\left(w^T x^{(n)} + b\right) = 1$

حال اگر طرفین را در α_i ضرب کرده و sum بگیریم رابطه زیر برقرار است زیرا طبق KKT condition می دانیم که $x^{(i)}$ هایی که SV نباشند، α_i متناظرشان 0 است

$$\underbrace{\sum_{i=1}^{N} \alpha_i y^{(i)} \left(w^T x^{(i)} + b\right)}_{w^T \underbrace{\sum_{i=1}^{N} \alpha_i y^{(i)} x^{(i)}}_{w} \; + \; b \underbrace{\sum_{i=1}^{N} \alpha_i y^{(i)}}_{0}} = \sum_{i=1}^{N} \alpha_i \longrightarrow w^T w = \|w\|^2 = \sum_{i=1}^{N} \alpha_i$$

$$\boxed{\text{margin} = \frac{1}{\|w\|} = \frac{1}{\sqrt{\sum_{i=1}^{N} \alpha_i}}}$$

2.2

می دانیم که اگر بعد جدیدی اضافه کنیم فاصله نقاط تنها می تواند افزایش یابد. با توجه به اینکه صورت سوال فرض کرده که margin افزایش پیدا نمی کند پس تغییری در margin نداریم. پس ابرصفحه جدا کننده همان ابرصفحه قبلی می ماند و فقط یک بعد به آن اضافه می شود

اگر داده ها از قبل linearly separable باشند بعد از افزایش بعد نیز قابل جدا کردن اند و همین ابرصفحه این کار را انجام می دهد

پس در فاز training ایرادی پیش نمی آید و مدل robust است اما برای generalization مشکلاتی نظیر spurious relationship ایجاد می شود

2.3

دو داده با لیبل متفاوت linearly separable است و بنابراین طبق فرمولی که برای hard margin SVM داشتیم، ماتریس زیر را به QP ورودی می دهیم :

$$Q = \begin{bmatrix} y^{(1)} y^{(1)} x^{(1)T} x^{(1)} & y^{(1)} y^{(2)} x^{(1)T} x^{(2)} \\ y^{(2)} y^{(1)} x^{(2)T} x^{(1)} & y^{(2)} y^{(2)} x^{(2)T} x^{(2)} \end{bmatrix}$$

شرط لازم برای اینکه بهینه سازی convex شود و جواب را بتوانیم از QP بگیریم آن است که این ماتریس PSD باشد

$$\xrightarrow{\text{لیبل متفاوت}} \begin{bmatrix} x^{(1)T} x^{(1)} & -x^{(1)T} x^{(2)} \\ -x^{(2)T} x^{(1)} & x^{(2)T} x^{(2)} \end{bmatrix}$$

برای ماتریس های 2x2 می دانیم که ماتریس PSD است اگر و تنها اگر شرایط زیر برقرار باشند :

- Q is symmetric $\longrightarrow$ مشاهده می شود که برقرار است ✓
- $trace(Q) > 0 \longrightarrow$ ✓ $trace(Q) = \|x^{(1)}\|^2 + \|x^{(2)}\|^2 > 0$
- $\det(Q) \geq 0 \longrightarrow$ ✓ $\det(Q) = \|x^{(1)}\|^2 \|x^{(2)}\|^2 - \langle x^{(1)}, x^{(2)} \rangle^2 \geq 0$

طبق کوشی شوارتز

بنابراین QP جواب $(\vec{\alpha})$ را به دست می آورد و طبق فرمول های درس w و b را به دست می آوریم و معادله ابرصفحه را به صورت کامل داریم و بنابراین فاصله آن تا مبدأ را که برابر $\frac{|b|}{\|w\|}$ است را نیز داریم. اگر لیبل ها متفاوت نبود ابرصفحه بین آنها نمی افتاد و margin مقدار محدود نمی داشت

3.1

متغیر ξ_i^* را میزان تخطی مربوط به کمتر بودن $f(x^{(i)})$ از $y^{(i)}$ در نظر می‌گیریم پس می‌توان نوشت

$$\xi_i^* = \max\left(y^{(i)} - f(x^{(i)}) - \varepsilon \;,\; 0\right)$$

متغیر ξ_i را میزان تخطی مربوط به بیشتر بودن $f(x^{(i)})$ از $y^{(i)}$ در نظر می‌گیریم پس می‌توان نوشت

$$\xi_i = \max\left(f(x^{(i)}) - y^{(i)} - \varepsilon \;,\; 0\right)$$

⊛ متغیر ε را برای پارامتری کردن جریمه تخطی به فرمول اضافه کردیم

حال واضح است که در آن واحد تنها یکی از جریمه ها (ξ_i یا ξ_i^*) اعمال خواهد شد زیرا شرط مثبت بودن آنها اشتراک ندارد

$$\xi_i^* \geq 0 \rightarrow y^{(i)} \geq f(x^{(i)}) + \varepsilon$$

⊛ اشتراک ندارند

$$\xi_i \geq 0 \rightarrow y^{(i)} \leq f(x^{(i)}) - \varepsilon$$

$$L_\varepsilon(x^{(i)}, y^{(i)}, f) = \begin{cases} \overbrace{\max\left(y^{(i)} - f(x^{(i)}) - \varepsilon \;,\; 0\right)}^{\xi_i^*} & y^{(i)} \geq f(x^{(i)}) \\ \underbrace{\max\left(f(x^{(i)}) - y^{(i)} - \varepsilon \;,\; 0\right)}_{\xi_i} & y^{(i)} < f(x^{(i)}) \end{cases} \overset{⊛}{=} \xi_i + \xi_i^*$$

بنابراین صورت primal مسئله به صورت گفته شده در می‌آید و شروط روبه رو را داریم

① $\xi_i^* \geq 0$
② $\xi_i^* \geq y^{(i)} - f(x^{(i)}) - \varepsilon$
③ $\xi_i \geq 0$
④ $\xi_i \geq f(x^{(i)}) - y^{(i)} - \varepsilon$

3.2 primal

$$\min_{w, b, \xi_i, \xi_i^*} \; \max_{\substack{\alpha_i, \alpha_i^*, \\ \beta_i, \beta_i^*}} \; \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) - \overset{①}{\sum_{i=1}^{N}\alpha_i^*\xi_i^*} - \overset{②}{\sum_{i=1}^{N}\beta_i^*\left(\xi_i^* - y^{(i)} + \overbrace{f(x^{(i)})}^{w^T x^{(i)} + b} + \varepsilon\right)}$$
$$- \overset{③}{\sum_{i=1}^{N}\alpha_i\xi_i} - \overset{④}{\sum_{i=1}^{N}\beta_i\left(\xi_i - \underbrace{f(x^{(i)})}_{w^T x^{(i)} + b} + y^{(i)} + \varepsilon\right)}$$

مسئله dual جای max و min را عوض می‌کند. با توجه به اینکه برای مسئله ما primal = dual پس ابتدا نسبت به w و b و ξ_i و ξ_i^* کمینه می‌کنیم

$$\nabla_w L = w - \sum_{i=1}^{N}\beta_i^* x^{(i)} + \sum_{i=1}^{N}\beta_i x^{(i)} = 0 \rightarrow w = \sum_{i=1}^{N}(\beta_i^* - \beta_i)x^{(i)}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N} -\beta_i^* + \sum_{i=1}^{N}\beta_i = 0 \rightarrow \sum_{i=1}^{N}(\beta_i - \beta_i^*) = 0$$

$$\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \beta_i = 0 \rightarrow \alpha_i = C - \beta_i$$

$$\frac{\partial L}{\partial \xi_i^*} = C - \alpha_i^* - \beta_i^* = 0 \rightarrow \alpha_i^* = C - \beta_i^*$$

KKT conditions

$$\alpha_i\xi_i = (C - \beta_i)\xi_i = 0$$
$$\alpha_i^*\xi_i^* = (C - \beta_i^*)\xi_i^* = 0$$
$$\beta_i\left(\xi_i - f(x^{(i)}) + y^{(i)} + \varepsilon\right) = 0$$
$$\beta_i^*\left(\xi_i^* - y^{(i)} + f(x^{(i)}) + \varepsilon\right) = 0$$

3.2 ادامه

$$\longrightarrow L = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\beta_i^*-\beta_i)(\beta_j^*-\beta_j)\, x^{(i)T}x^{(j)} + \cancel{C\sum_{i=1}^{N}(\xi_i+\xi_i^*)}$$

$$-\sum_{i=1}^{N}(\cancel{C}-\beta_i^*)\xi_i^* - \sum_{i=1}^{N}\beta_i^*\left(\xi_i^* - y^{(i)} + \sum_{j=1}^{N}\left((\beta_j^*-\beta_j)\,x^{(j)T}x^{(i)}\right) + \cancel{b} + \varepsilon\right)$$

$$-\sum_{i=1}^{N}(\cancel{C}-\beta_i)\xi_i - \sum_{i=1}^{N}\beta_i\left(\xi_i - \sum_{j=1}^{N}\left((\beta_j^*-\beta_j)\,x^{(j)T}x^{(i)}\right) + \cancel{b} + y^{(i)} + \varepsilon\right)$$

$$= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\beta_i^*-\beta_i)(\beta_j^*-\beta_j)\,x^{(i)T}x^{(j)} + \cancel{\sum_{i=1}^{N}\beta_i^*\xi_i^*} + \cancel{\sum_{i=1}^{N}\beta_i\xi_i}$$

$$-\varepsilon\sum_{i=1}^{N}(\beta_i+\beta_i^*) + \sum_{i=1}^{N}y^{(i)}(\beta_i^*-\beta_i) - \cancel{\sum_{i=1}^{N}\beta_i^*\xi_i^*} - \cancel{\sum_{i=1}^{N}\beta_i\xi_i}$$

$$-\sum_{i=1}^{N}\sum_{j=1}^{N}(\beta_i^*-\beta_i)(\beta_j^*-\beta_j)\,x^{(j)T}x^{(i)}$$

$$= \boxed{-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\beta_i^*-\beta_i)(\beta_j^*-\beta_j)\,x^{(i)T}x^{(j)} - \varepsilon\sum_{i=1}^{N}(\beta_i+\beta_i^*) + \sum_{i=1}^{N}y^{(i)}(\beta_i^*-\beta_i)}$$

$$\left.\begin{array}{l}\beta_i \geq 0\\ \alpha_i \geq 0 \longrightarrow C-\beta_i \geq 0 \longrightarrow \beta_i \leq C\end{array}\right\} \longrightarrow 0 \leq \beta_i \leq C$$

حال باید عبارت به دست آمده max شود، با شرط روبه رو

نام گذاری من با صورت سوال فرق دارد و به جای α، β گذاشته ام

3.3 مشابه مسئله دسته بندی به مقداری برحسب یک متغیر لاگرانژ رسیدیم و می توان آن را توسط QP حل کرد. کافی است فرم ماتریسی عبارت به دست آمده را بنویسیم و سپس ماتریس ضرایب را به دست آوریم. البته چون عبارت به دست آمده با مسئله دسته بندی متفاوت است، روند متفاوتی باید برای به دست آوردن ماتریس ضرایب و ورودی دادن آن به QP انجام شود

3.4 مشابه قبل داده هایی که β_i یا β_i^* شان مثبت باشند SV خواهند بود

$$\left.\begin{array}{l}\beta_i > 0 \xrightarrow{KKT} \xi_i = f(x^{(i)}) - y^{(i)} - \varepsilon\\ \text{طبق تعریف } \xi_i = \max\left(f(x^{(i)}) - y^{(i)} - \varepsilon, 0\right)\end{array}\right\} \longrightarrow y^{(i)} \leq f(x^{(i)}) - \varepsilon$$

$$\left.\begin{array}{l}\beta_i^* > 0 \xrightarrow{KKT} \xi_i^* = y^{(i)} - f(x^{(i)}) - \varepsilon\\ \text{طبق تعریف } \xi_i^* = \max\left(y^{(i)} - f(x^{(i)}) - \varepsilon, 0\right)\end{array}\right\} \longrightarrow y^{(i)} \geq f(x^{(i)}) + \varepsilon$$

⊛ بنابراین داده هایی که خارج از رنج $f \pm \varepsilon$ هستند SV هستند

$\boxed{3.5}$

$$W = \sum_{i=1}^{N} (\beta_i^* - \beta_i)\, x^{(i)} \longrightarrow f(x^{(k)}) = \sum_{i=1}^{N} \left((\beta_i^* - \beta_i)\, x^{(i)T} x^{(k)} \right) + b$$

از رابطه روبه رو می توان برای پیش بینی استفاده کرد

همچنین می توان از کرنل هم استفاده کرد و در تمامی فرمول ها تنها به جواب ضرب داخلی $\phi(x^{(i)})$ ها نیاز است و نمی خواهد فضا را ملاقات کنیم

$$W = \sum_{i=1}^{N} (\beta_i^* - \beta_i)\, \phi(x^{(i)}) \longrightarrow f(x^{(k)}) = \sum_{i=1}^{N} \left((\beta_i^* - \beta_i)\, \underbrace{\phi(x^{(i)})^T \phi(x^{(k)})}_{k(x^{(i)},\, x^{(k)})} \right) + b$$

$\boxed{3.6}$

اگر $f(x)$ را معادل خط / ابرصفحه به دست آمده بدانیم ، بازه ی $f(x) \pm \varepsilon$ بازه ای است که خطای به آن اندازه tolerate می شود سمپل هایی که داخل این بازه قرار دارند ارور صفر دارند. بنابراین افزایش ε باعث کاهش پیچیدگی مدل و آسان گرفتن می شود و کاهش ε نیز باعث افزایش پیچیدگی مدل و سخت گیری می شود ($\frac{1}{\varepsilon}$ شبیه ضریب regularization عمل می کند)

C بزرگ یعنی ξ_i و ξ_i^* باید کوچک باشند و اجازه violation کمی می دهیم و به همین صورت C کوچک یعنی اجازه violation بیشتری می دهیم مانند قبل افزایش C به نوعی باعث کاهش پیچیدگی مدل و کاهش C باعث افزایش پیچیدگی مدل می شود ($\frac{1}{C}$ شبیه ضریب regularization عمل می کند)

$\boxed{4.1}$

اگر سمپلی که حذف می‌شود جزء SV ها نباشد هیچ تغییری رخ نمی‌دهد زیرا فقط SV ها روی margin تأثیر می‌گذارند

اگر سمپل از SV ها باشد اما به جز آن حداقل 2 تا SV دیگر با همان لیبل وجود داشته باشد باز تغییری ایجاد نمی‌شود
زیرا 2 نقطه برای نگه داشتن خط کافی است (ما برای خط حرف زدیم اما به طور کلی برای ابرصفحه n بُعدی حداقل n-1 سمپل SV نیاز است تا مرز جابجا نشود)

در غیر اینصورت مرز تصمیم به سمت سمپل حذف شده می‌رود تا margin بزرگتری به دست آید

در logistic regression مرز تصمیم تغییر می‌کند زیرا تابع هزینه‌ای داریم که به ازای هر سمپل error آن را حساب کرده و بر اساس آن مرز تصمیم را جابجا می‌کند
پس حتی اگر batch هم داشته باشیم، هر سمپل در loss تأثیر گذاشته و حذف سمپل قطعاً مرز تصمیم را جابجا می‌کند

$\boxed{4.2}$

می‌دانیم که $\varepsilon_i > 1$ معادل اشتباه لیبل خوردن سمپل i ام است. یعنی به طور کلی داریم

$\varepsilon_i = 0 \longrightarrow$ correctly classified

$0 < \varepsilon_i \leq 1 \longrightarrow$ correctly classified but inside margin

$\varepsilon_i > 1 \longrightarrow$ misclassified

پس واضح است که $\sum_{i=1}^{N} \varepsilon_i$ یک کران بالا برای تعداد سمپل هایی است که اشتباه لیبل زده شده اند

$\boxed{4.3}$ minimize $\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \varepsilon_i$

$C \longrightarrow \infty$

یعنی باید ε_i ها کوچک باشند و اجازه violation نمی‌دهیم (به حالت hard margin برگشتیم)

$C \longrightarrow 0$

یعنی ε_i ها می‌توانند بسیار بزرگ باشند و اجازه violation زیادی می‌دهیم. با توجه به اینکه در این حالت وزن بیشتری برای $\|w\|^2$ است پس $\|w\|$ بیشتر کمینه می‌شود
و بنابراین margin بزرگتری داریم

$\frac{1}{C}$ را می‌توان شبیه ضریب regularization دید زیرا بین پیچیدگی مدل, generalization یک tradeoff ایجاد می‌کند
اگر C را بزرگ قرار دهیم اجازه پیچیدگی بیشتری به مدل می‌دهیم و اگر C را کم انتخاب کنیم جلوی پیچیدگی را می‌گیریم

4.4

مدل logistic regression به دلیل داشتن تابع هزینه بین ابرصفحه های جدا کننده که به درستی دیتای آموزش را جدا کرده اند نیز تفاوت قائل می شود و به طور کلی عملکرد خوبی در پیدا کردن مرزی که بهتر generalize کند دارد. اما SVM ابرصفحه با margin بیشینه را انتخاب می کند و از این نظر از حیث generalization از LR بهتر است

بنابراین اگر دیتا جدایی پذیر خطی باشد هر 2 مدل جواب را پیدا می کنند و روی داده آموزش به ارور صفر می رسند. تفاوت آن است که SVM در زمان تست بهتر generalize می کند (هم از نظر ابرصفحه ای که margin بزرگتری دارد و هم از نظر robust بودن نسبت به outlier ها در داده آموزش)

4.5

در این حالت عملکرد 2 مدل شبیه هم می شود (به جز robust بودن SVM نسبت به outlier ها) و هر 2 مدل جوابی sub optimal برمی گردانند که تا حدی توانسته داده را دسته بندی کند.

ضمناً هر دو مدل می توانند از تبدیل غیرخطی ϕ استفاده کنند اما SVM از این نظر بهتر است چون نیازی ندارد که فضای ϕ را ملاقات کند و بنابراین سریع تر است. حتی می تواند داده ها را به فضای ∞ بعدی ببرد که LR نمی تواند و از این نظر قابلیت دسته بندی بالاتری دارد