PS 2

Q1) ⓐ Code converges on data A but not on B

ⓑ B is perfectly linearly separable unlike A. The optimization is unbounded & $\theta \to \infty$
Regularization eg L2 can put a limit on θ
SVM might be more appropriate for linearly separable.

(c) i. No - as $\theta \to \infty$ regardless of size of lr
ii. Depends - if $lr \to 0$ in finite time it will converge. If time too short, underfits
iii. No - as only changes values of θ as already linearly separable
iv. Yes - limits $\|\cdot\|_{2}$ of θ to be with a bound so traing stop when bound achieved
v. Depends - if data is very large, ~~#~~ noise cancels & problem persists. Here it may help in some instances due to $\sum$ noise $\neq 0$ but depends if boundary is still linearly separable

(d) No - SVM not vulnerable

it maximizes the distance / margin b/w the two classes (distance of boundary from nearest pt)
it is implicitly regularizing θ as $\min_{\theta} \frac{1}{2}\|\theta\|^2$

Q2 (a) 1722

(b) 0.978

(c) claim, non, prize, tone, might.

(d) 0.1
~~0~~.9695 accuracy

Q3 (a) Yes : $k_1 + k_2 = K = K^T$
$z^T K z = z^T K_1 z + z^T K_2 z \geq 0$
PSD

(c) Yes : $z^T a K z = a(z^T K z) \geq 0 \because a \in \mathbb{R}^+$

(b) No : by c, let $K_2 = 2K_1$ is kernel
$z^T(K_1 - K_2)z$
$= -z^T K_1 z \leq 0$ for certain z
$\therefore$ not PSD

(d) No : by b & c, ~~&~~ $a \in \mathbb{R}^+$

(f) Yes : $f(x) \in \mathbb{R} \therefore \langle f(x); f(z) \rangle = f(x)f(z)$
as scalar ~~as R~~ multiplication

(g) Yes : basic feature map
$k_3 \; p \times p$
$\phi \; d \to p \therefore k_3 (\in \mathbb{R}^p, \in \mathbb{R}^p)$
$\therefore$ kernel

(h) Yes : By a, c, e, f
Eg $p() = ax^3 + cx^2 + d + x$

(e) $K(x,z) = K_1(x,z)\,K_2(x,z)$

$$= \sum_i \phi_1^i(x)\,\phi_1^i(z) \sum_j \phi_2^j(x)\,\phi_2^j(z)$$

$$= \sum_i \sum_j \phi_1^i(x)\,\phi_1^i(z)\,\phi_2^j(x)\,\phi_2^j(z)$$

$\phi_1^j \phi_2^j$ is ϕ_3^{ij} $\therefore \sum_{ij} \theta^{ij}(x)\,\theta^{ij}(z)$

$\therefore \langle \theta(x), \theta(z) \rangle$

$\therefore$ kernel

Q3 (i) $\theta^{(i)} = \sum_{j=1}^{i} \beta_j \phi(x^{(j)})$ $\quad \theta^{(0)} = 0$ when $\beta_j = 0$

β is y essentially & all errors at $t=0$

(ii) $h_\theta^{(i)}(x^{i+1}) = g(\theta^{(i)T} \phi(x^{(i+1)}))$ $\quad g(z) = \text{sign}(z)$

$$= g\left(\left[\sum_{j=1}^{i} \beta_j \phi(x^{(j)}) \right] \phi(x^{(i+1)}) \right)$$

$$= g\left(\sum_{j=1}^{i} \beta_j K(x^{(j)}, x^{(i+1)}) \right)$$

(iii) $$\theta^{(i+1)} = \sum_{j=1}^{i+1} \beta_j \phi(x^{(j)}) = \underbrace{\sum_{j=1}^{i} \beta_j \phi(x^{(j)})}_{\theta^{(i)}} + \beta_{i+1} \phi(x^{i+1})$$

$$\beta_{i+1} = \alpha(y^{(i+1)} - g(-)\,\phi(x^{(i+1)})$$

$$\therefore \theta^{(i+1)} = \theta^{(i)} + \alpha\left(y^{(i+1)} - g(\theta^{(i)T} \phi(x^{(i+1)}))\right) \phi(x^{(i+1)}))$$

(c) linear kernel $(a \cdot b)$ poor fit as dat not linear or not approx linear

RBF can build circles to separate data

6. (a) $\theta_{MAP} = \arg\max_{\theta} p(\theta | x, y)$

$$= \arg\max_{\theta} \frac{p(y|x,\theta)\, p(\theta|x)}{p(y|x)}$$

$$= \arg\max_{\theta} p(y|x,\theta)\, p(\theta|x)$$

const wrt θ

$$= \arg\max_{\theta} p(y|x,\theta)\, p(\theta)$$

(b) Take $-\log$

$$\therefore \theta_{MAP} = \arg\min_{\theta} -\log p(y|x,\theta) - \log p(\theta)$$

$$\because \theta \sim N(0, \eta^2 I)$$

$$\therefore -\log p(\theta) = -\log(\text{const}) + \frac{1}{2}\theta^T\left(\frac{1}{\eta^2} I\right)\theta$$

$$\therefore \arg\min_{\theta} -\log p(y|x,\theta) + \underbrace{\frac{1}{2\eta^2}}_{\lambda}\|\theta\|_2^2$$

(c) $\because -\log p(y|x,\theta) = \frac{1}{2\sigma^2}\|\bar{y} - X\theta\|_2^2$ ignore const

$$\therefore J(\theta) = \frac{1}{2\sigma^2}\|\bar{y} - X\theta\|_2^2 + \frac{1}{2\eta^2}\|\theta\|_2^2$$

$$\nabla_\theta J(\theta) = 0 = \frac{1}{\sigma^2}X^T(X\theta - \bar{y}) + \frac{1}{\eta^2}\theta$$

$$\therefore \hat{\theta}_{MAP} = \left(X^TX + \frac{\sigma^2}{\eta^2} I\right)^{-1} X^T y$$

(d) $f_L(\theta_i | \bar{0}, b) = \frac{1}{2b} e^{-\frac{\theta_i}{b}}$

$\therefore -\log f — = -\text{const} + \frac{\|\theta\|_1}{b}$

using c, ignore const

$J(\theta) = \frac{1}{2\sigma^2}\|X\theta - y\|_2^2 + \frac{1}{b}\|\theta\|_1$

$\hat{\theta}_{map}$ arg min $J(\theta)$

Rescale to $J(\theta)$ in form $\|X\theta - y\|_2^2 + \frac{2\sigma^2}{b}\|\theta\|_1$

$\gamma = \frac{2\sigma^2}{b}$

Q7 (a) $l(\theta) = \sum_{i=1}^{m} y^{(i)} \log h(x^{(i)}) + (1-y^{(i)}) \log(1-h(x^{(i)}))$

$\frac{\partial l(\theta)}{\partial \theta_j} = \sum_{i=1}^{m} (y^{(i)} - h(x^{(i)})) x_j^{(i)} = 0$ — (1)

bias $= 1$ $\therefore$ set $j = 0$

$\sum_{i=1}^{m} P(y=1|x;\theta) = \therefore \sum_{i=1}^{m} h(x^{(i)}) = \sum_i y = \sum_{i=1}^{m} \mathbb{I}\{y^{(i)} = 1\}$ — (2)

change to $(0,1) = (a,b)$ as $i = 1$ to m

$\therefore |\{i \in I_{a,b}\}| = m$

$\therefore \frac{\sum_{i \in I_{a,b}} P \ldots}{|\{i \ldots\}|} = \frac{\sum_{i \in A,B} \mathbb{I}\{y^{(i)} = 1\}}{|\ldots|}$

(b) No, no for converse

- model can misclassif based on threshold set so even through prob perfect, not 100% accuracy
- model can get correct accuracy with non-optimal probabilities & thus is not correct due to threshold. $P(y=1|x) = 0.9$ when truth is 0.8

(c) may improve by reducing overfitting & overconfidence. too strong L2 can underfit.

$\nabla J(\theta) = (1) + \lambda\theta_2 \quad \therefore (2) \sum h + \lambda\theta$ $\therefore$ not calibrated well