1. **[15 points] Logistic Regression: Training stability**

In this problem, we will be delving deeper into the workings of logistic regression. The goal of this problem is to help you develop your skills debugging machine learning algorithms (which can be very different from debugging software in general).

We have provided a implementation of logistic regression in `src/p01_lr.py`, and two labeled datasets A and B in `data/ds1_a.txt` and `data/ds1_b.txt`

Please do not modify the code for the logistic regression training algorithm for this problem. First, run the given logistic regression code to train two different models on A and B. You can run the code by simply executing `python p01_lr.py` in the `src` directory.

(a) [2 points] What is the most notable difference in training the logistic regression model on datasets A and B? training dataset B does not converge. (slowly).

(b) [5 points] Investigate why the training procedure behaves unexpectedly on dataset B, but not on A. Provide hard evidence (in the form of math, code, plots, etc.) to corroborate your hypothesis for the misbehavior. Remember, you should address why your explanation does *not* apply to A. B can be perfectly separated by a boundary. Many solutions can be used. to fit training set.
Hint: The issue is not a numerical rounding or over/underflow error.

(c) [5 points] For each of these possible modifications, state whether or not it would lead to the provided training algorithm converging on datasets such as B. Justify your answers.

i. Using a different constant learning rate.
ii. Decreasing the learning rate over time (e.g. scaling the initial learning rate by $1/t^2$, where t is the number of gradient descent iterations thus far).
iii. Linear scaling of the input features.
iv. Adding a regularization term $\|\theta\|_2^2$ to the loss function.
v. Adding zero-mean Gaussian noise to the training data or labels.

(d) [3 points] Are support vector machines, which use the hinge loss, vulnerable to datasets like B? Why or why not? Give an informal justification.

Hint: Recall the distinction between functional margin and geometric margin.

This can cause the ~~lea~~ a very small learning rate which falls in the range of 1e-15

i. the reason for slow convergence is caused by small grad, changing learning rate does not ~~ch~~ fix it.

ii. No.

should be yes.

~~iii. No.~~ ~~can~~ have the same affect scaling weight.

iv. Yes. then loss can't be zero.

v. Yes. then training set would not be perfectly separatable.

(d) No. SVM uses geometric margin. When dataset is perfectly separatable. it would only be one answer.

2. (a). $\ell(\theta) = \prod_{i=1}^{m} p(y^{(i)} \mid x^{(i)}; \theta)$

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

$$\frac{\partial}{\partial \theta_j} \ell(\theta) = \sum_{i=1}^{m} \left(y^{(i)} \frac{1}{h(x^{(i)})} - (1-y^{(i)}) \frac{1}{1-h(x^{(i)})} \right) h(x^{(i)})(1-h(x^{(i)})) x_j$$

$$= \sum_{i=1}^{m} \left(y^{(i)}(1-h(x^{(i)})) - (1-y^{(i)}) h(x^{(i)}) \right) x_j$$

$$= \sum_{i=1}^{m} (y^{(i)} - h(x^{(i)})) x_j$$ We know $x_0 = 1$.

$$\frac{\partial}{\partial \theta_0} = \sum_{i=1}^{m} (y^{(i)} - h(x^{(i)})) x_0 = 0.$$ set it to 0.

$$\sum_{i=1}^{m} y^{(i)} = \sum_{i=1}^{m} h(x^{(i)})$$

because $i \in (0,1) \to 1 \le i \le m$.

(b). ~~No.~~ we can only get the probability of a $P(y^{(i)} = 1 \mid x^{(i)}; \theta)$.

When $P \to 0.5$, ~~the probability to~~ predict mistakenly is pretty large.

The converse does not hold because when a model achieves perfect accuracy,

Then $P(y^{(i)} \mid x^{(i)}; \theta) = 1$.

No, it is saying the sum of

(c). if ~~perfectly~~ accuracy is respect to ~~training~~ set, (infinitely large).

Yes. because the interval I can be infinitely small.) ~~When shrinking size to~~

~~Then every prediction (probability) can be perfect accuracy~~

~~The converse does not hold. because when interval only has one point~~

if ~~set is~~ test set is infinitely large.

No. perfectly calibrated $\not\to$ perfectly accuracy. it only implies the average is

(c) [5 points] **Coding problem.** We will now tune the hyperparameter τ. In `src/p05c_tau.py`, find the MSE value of your model on the validation set for each of the values of τ specified in the code. For each τ, plot your model's predictions on the validation set in the format described in part (b). Report the value of τ which achieves the lowest MSE on the `valid` split, and finally report the MSE on the `test` split using this τ-value.

$\boxed{0.017}$

$\boxed{\text{this is wrong}}$ because I did not notice the interval I is open on both sides.

(b) perfectly calibrated. → perfect accuracy.

when shrink the size of interval to 1. then $P(y^{(i)}=1|x^{(i)};\theta) = 1\{y^{(i)}=1\}$.

since rhs is either 1 or 0, the lhs must also be 1 or 0.

Then for $\forall x^{(i)}$, $P(y^{(i)}|x^{(i)};\theta) = 1$ or 0. If it is 1, and $y^{(i)}=1$, the prediction matches the $\boxed{y^{(i)}=1}$. When it is 0, $P(y^{(i)}|x^{(i)};\theta)=0$. and there is no example.

so. It achieves perfect accuracy.

wrong!

perfect accuracy → perfectly calibrated.

↳ for $\forall x^{(i)}$ the prediction matches the label. $h_\theta(x^{(i)}) = y^{(i)}$, then $h_\theta(x^{(i)})$ is 1 or 0.

for ~~many~~ any interval (a,b). suppose there are c $y^{(i)}=1$ labels. and d $y^{(i)}=0$ labels.

where $c, d \geq 0$. $\sum_{i\in(a,b)} P(y^{(i)}=1|x^{(i)};\theta) = \sum_{i\in(a,b)} h_\theta(x^{(i)}) = c\times 1 + d\times 0 = c$.

$\sum_{i\in I(a,b)} 1(y^{(i)}=1) = c\times 1 + d\times 0 = c$. Thus $\sum_{i\in I(a,b)} 1(y^{(i)}=1) = \sum_{i\in I(a,b)} P(y^{(i)}=1|x^{(i)};\theta)$

(c). from $\frac{\partial}{\partial\theta_0} \ell(\theta) = \sum_{i=1}^{m} (y^{(i)} - h(x^{(i)})) + 2\theta_0 = 0$. then $\sum_{i=1}^{m} h(x^{(i)}) = \sum_{i=1}^{m} 1\{y^{(i)}=1\} + 2\theta_0$

it ~~shifts the~~ makes the model not well calibrated.

2.(b). Assume the model achieves perfect accuracy.

And pick the $I = (a, b) = (0.5, 1)$.

$\forall i \in I_{a,b}$ $\quad 0.5 < P(y^{(i)} = 1 | x^{(i)}; \theta) < 1$

Then $\sum_{i \in I_{a,b}} P(y^{(i)} = 1 | x^{(i)}; \theta) < \sum_{i \in I_{a,b}} 1(y^{(i)} = 1)$.

Since all predictions must be 1 to achieve perfect accuracy.

This violates the well-calibrated property

Thus The model is not well-calibrated.

Assume the model is well-calibrated. Then.

$$\frac{\sum P(y^{(i)} = 1 | x^{(i)}; \theta)}{|\{i \in I_{a,b}\}|} = \frac{\sum 1(y^{(i)} = 1)}{|\{i \in I_{a,b}\}|}$$

pick $I = (0.5, 1)$.

Because the model output is $0 < P(y^{(i)} = 1 | x^{(i)}; \theta) < 1$.

The average $0 < \frac{1}{m} \sum P(y^{(i)} = 1 | x^{(i)}; \theta) < 1$.

Because when $0 < P < 1$, the model would predicts 1.

But the total sum of $y^{(i)} = 1$ is not equal to the total num of examples.

So the model is not perfectly accurate.!

Rewrite

3. (a). $\theta_{MAP} = \arg\max_{\theta} p(\theta \mid X, y)$

$$= \arg\max_{\theta} p(\theta \mid X, y) \cdot P(y \mid X) \quad \text{since } P(y \mid X) \text{ is constant.}$$

$$= \text{''} \quad \frac{P(\theta, X, y)}{P(\cancel{X, y})} \cdot \frac{P(\cancel{y, X})}{P(X)}.$$

$$= \text{''} \quad \frac{P(y \mid \theta, X) \cdot P(\cancel{y}\theta, X)}{P(X)}.$$

$$= \text{''} \quad \frac{P(y \mid \theta, X) \cdot P(\theta \cancel{\theta} \mid X) \cdot P(X)}{P(X)}.$$

$$= \arg\max_{\theta} P(y \mid \theta, X)\, P(\theta).$$

(b). $\theta_{MAP} = \arg\max_{\theta} P(\theta \mid X, y) = \arg\max_{\theta} P(y \mid X, \theta) P(\theta)$

Since $\theta \sim \mathcal{N}(0, \eta^2 I)$

$$= \arg\max_{\theta} P(y \mid X, \theta) \frac{1}{\eta\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\theta - 0}{\eta}\right)^2} \quad \text{since } \frac{1}{\eta\sqrt{2\pi}} \text{ is constant.}$$

$$\propto \arg\max_{\theta} P(y \mid X, \theta)\, e^{-\frac{1}{2}\left(\frac{\theta^T \theta}{\eta^2 I}\right)}$$

$$= \arg\max_{\theta} \log P(y \mid X, \theta) + \left(-\frac{1}{2} \cdot \frac{1}{\eta^2 I} \cdot \|\theta\|_2^2\right).$$

$$= \arg\min_{\theta} -\log P(y \mid X, \theta) + \frac{1}{2\eta^2 I} \|\theta\|_2^2$$

where $\lambda = \frac{1}{2\eta^2 I}$

(c). $$\theta_{MAP} = \arg\min_{\theta} -\log\left(\prod_{i=1}^{m} P(y^{(i)}|x^{(i)},\theta)\right) + \lambda\|\theta\|_2^2$$

$$= \arg\min_{\theta} -\left(\sum_{i=1}^{m} \log\left(\frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right)\right)\right) + \lambda\|\theta\|_2^2$$

$$= \arg\min_{\theta} -\sum_{i=1}^{m} \log\left(\frac{1}{\sqrt{2\pi}\sigma}\right) - \left(\frac{(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right) + \lambda\|\theta\|_2^2$$

$$= \arg\min_{\theta} \sum_{i=1}^{m} -\log\left(\frac{1}{\sqrt{2\pi}\sigma}\right) + \frac{(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}) + \lambda\|\theta\|_2^2$$

since $-\log\left(\frac{1}{\sqrt{2\pi}\sigma}\right)$, σ^2 are constant.

$$= \arg\min_{\theta} \left(\sum_{i=1}^{m} \frac{1}{2}(y^{(i)}-\theta^T x^{(i)})^2\right) + \lambda\|\theta\|_2^2$$

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

$$= \quad '' \quad \log(P(y|x,\theta)) + \sum_{i=1}^{n} \log\left(\frac{1}{2bI}\exp\left(-\frac{|\theta^i|}{bI}\right)\right)$$

$$= \quad '' \quad \log P(y|x,\theta) + \sum_{i=1}^{n}\log\left(\frac{1}{2bI}\right) - \sum_{i=1}^{n}\frac{|\theta|^i}{bI}$$

$$= \quad '' \quad \log\left(\prod_{i=1}^{m} P(y^{(i)}|x^{(i)},\theta)\right) - \frac{n}{bI}\|\theta\|_1 \qquad \gamma = \frac{n}{bI}.$$

$$= \arg\min_{\theta} -\sum_{i=1}^{m} \log\left(\frac{1}{\sqrt{2\pi}\sigma}\exp\left(\frac{-(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right)\right) + \gamma\|\theta\|_1$$

$$= \quad '' \quad \left(\sum_{i=1}^{m} \overbrace{-\log\left(\frac{1}{\sqrt{2\pi}\sigma}\right)}^{\text{constant.}} + \frac{(y^{(i)}-\theta^T x^i)^2}{2\sigma^2}\right) + \gamma\|\theta\|_1$$

$$= \arg\min_{\theta} \sum_{i=1}^{m} \frac{(y^{(i)}-\theta^T x^i)^2}{2\sigma^2} + \gamma\|\theta\|_1$$

$$= \arg\min_{\theta} \|X\theta - \vec{y}\|_2^2 + \gamma\|\theta\|_1$$

Rewrite 3C

$$P(\vec{y}\,|\,X;\theta) = \prod_{i=1}^{m} P(y^{(i)}\,|\,x^{(i)};\theta)$$

$$= \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(\frac{-(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}\right)$$

$$\log(P(\vec{y}\,|\,X,\theta)) = \sum_{i=1}^{m} \log\left(\frac{1}{\sqrt{2\pi}\,\sigma}\right) + \sum_{i=1}^{m} \frac{-(y^{(i)}-\theta^T x^{(i)})^2}{2\sigma^2}$$

$$= -m\log(\sqrt{2\pi}\,\sigma) - \frac{1}{2\sigma^2}\sum_{i=1}^{m}(y^{(i)}-\theta^T x^{(i)})^2$$

$$= -m\left(\frac{1}{2}\log(2\pi) + \log(\sigma)\right) - \frac{1}{2\sigma^2}\|\vec{y} - X\theta\|_2^2$$

$$= \underbrace{-\frac{1}{2}m\log(2\pi)}_{\text{constant}} \underbrace{- m\log(\sigma)}_{\text{constant.}} - \frac{1}{2\sigma^2}\|\vec{y} - X\theta\|_2^2$$

$$\theta_{MAP} = \arg\min_{\theta} -\log P(y\,|\,X,\theta) + \frac{1}{2\eta^2 I}\|\theta\|_2^2$$

$$= \arg\min_{\theta} \frac{1}{2\sigma^2}\|\vec{y} - X\theta\|_2^2 + \frac{1}{2\eta^2 I}\|\theta\|_2^2$$

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

$$\frac{1}{\sigma^2}X^TX\theta - \frac{1}{\sigma^2}X^T\vec{y} + \frac{1}{\eta^2 I}\theta = 0.$$

$$\left(\frac{1}{\sigma^2}X^TX + \frac{1}{\eta^2 I}\right)\theta = \frac{1}{\sigma^2}X^T\vec{y}$$

$$\left(X^TX + \frac{\sigma^2}{\eta^2 I}\right)\theta = X^T\vec{y}$$

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

$$\nabla_\theta \|\vec{y} - X\theta\|_2^2$$

$$= \nabla_\theta (\vec{y} - X\theta)^T(\vec{y} - X\theta)$$

$$= \nabla_\theta (\vec{y}^T - \theta^T X^T)(\vec{y} - X\theta)$$

$$= \nabla_\theta\ \vec{y}^T\vec{y} \underbrace{- y^TX\theta - \theta^TX^Ty}_{2y^TX\theta} + (\theta^TX^T)(X\theta)$$

$$= -2X^Ty + 2X^TX\theta$$

Rewrites 3 (d). $\theta \sim \mathcal{L}(0, bI)$.

$$P(\theta) = \prod_{i=1}^{n} \frac{1}{2b} \exp\left(-\frac{|\theta_i - 0|}{b}\right).$$

$$= \frac{1}{(2b)^n} \exp\left(-\frac{|\theta|}{b}\right)$$

$$\log P(\theta) = \log\left(\frac{1}{(2b)^n}\right) - \frac{\|\theta\|_1}{b}$$

$$= -n\log(2b) - \frac{\|\theta\|_1}{b}$$ since $-n\log(2b)$ is constant.

$$\theta_{MAP} = \arg \max_{\theta} \log P(y|X,\theta) - \frac{\|\theta\|}{b} - n\log(2b)$$

$$= \arg\min_{\theta} -\log P(y|X,\theta) + \frac{\|\theta\|_1}{b}$$

$$= \arg\min_{\theta} \frac{1}{2\theta^2}\|\vec{y} - X\theta\|_2^2 + \frac{1}{b}\|\theta\|_1$$

$$= \arg\min_{\theta} \|\vec{y} - X\theta\|_2^2 + \frac{2\sigma^2}{b}\|\theta\|_1$$

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

(a). where x is the ith entry and z is the jth entry.

$K_{ij} = K(x,z) = K_1(x,z) + K_2(x,z)$. since K_1 K_2 are kernels.

$K_{ji} = K(z,x) = K_1(z,x) + K_2(z,x)$.

$K_{1\,ij} = K_{1\,ji} \Rightarrow K_1(x^{(i)}, x^{(j)}) = K_1(x^{(j)}, x^{(i)})$

same for K_2.

Since R side, $K_{ij} = K_{ji}$.

Thus the kernel matrix is symmetric.

let z be any vector $z \in \mathbb{R}^n$.

$$z^T K z = \sum_i \sum_j z_i K_{ij} z_j = \sum_i \sum_j z_i \left(K_1(x^{(i)}, x^{(j)}) + K_2(x^{(i)}, x^{(j)}) \right) z_j$$

$$= \sum_i \sum_j z_i K_1(x^{(i)}, x^{(j)}) z_j + z_i K_2(x^{(i)}, x^{(j)}) z_j$$

$$= \sum_i \sum_j z_i K_{1(ij)} z_j + \sum_i \sum_j z_i K_{2\,ij} z_j$$

$$= z^T K_1 z + z^T K_2 z$$

since K_1, K_2 are semi-definite, then both terms ≥ 0.

Thus. K is semi-definite. Thus, K is a valid kernel.

(b). $K_{ij} = K(x^{(i)}, x^{(j)}) = K_1(x^{(i)}, x^{(j)}) - K_2(x^{(i)}, x^{(j)})$

$K_{ji} = K(x^{(j)}, x^{(i)}) = K_1(x^{(j)}, x^{(i)}) - K_2(x^{(j)}, x^{(i)})$

since $K_1(x^{(i)}, x^{(j)}) = K_1(x^{(j)}, x^{(i)})$

K_2 ... $= K_2$ ...

Thus $K_{ij} = K_{ji}$. symmetry.

$$z^T K z = \sum_i \sum_j z_i K_{ij} z_j = \sum_i \sum_j z_i K_{1\,ij} z_j - \sum_i \sum_j z_i K_{2\,ij} z_j$$

$$= z^T K_1 z - z^T K_2 z$$

When for $z \in \mathbb{R}^n$, when $z^T K_1 z < z^T K_2 z$, $K(x,z)$ is not a valid kernel.

(c). $K_{ij} = K(x^{(i)}, x^{(j)}) = a k_1(x^{(i)}, x^{(j)})$ since $K_1(x^{(i)}, x^{(j)}) = k_1(x^{(j)}, x^{(i)})$

$K_{ji} = K(x^{(j)}, x^{(i)}) = a K_1(x^{(j)}, ax^{(i)})$ $K_{ij} = K_{ji}$ thus K is symmetric.

$z^T K z = \sum_i \sum_j z^T K_{ij} z = \sum_i \sum_j z^T a k_{ij} z$

$= a \sum \sum z^T k_{ij} z$ since K_1 is kernel.

$\sum \sum z^T K_{1ij} z \geq 0$. then $z^T K z \geq 0$.

$\boxed{\text{(d). Wrong}}$ $K_{ij} = -a k_1(x^{(i)}, x^{(j)})$ since $K_1(x^{(i)}, x^{(j)}) = K_1(x^{(j)}, x^{(i)})$

$K_{ji} = -a K_1(x^{(j)}, x^{(i)})$ $K_{ij} = k_{ji}$. thus K is symmetric.

$z^T K z = \sum \sum z^T K_{ij} z = -a \sum \sum z^T K_{ij} z$ since K_1 is kernel.

$\sum \sum z^T k_{ij} z \geq 0$. thus $z^T K z \geq 0$.

$$\boxed{\begin{aligned} z^T K z &= z^T(-aK_1)z \\ &= -a z^T K_1 z \\ &\leq 0. \end{aligned}}$$

(e). $K_{ij} = K_1(x^{(i)}, x^{(j)}) K_2(x^{(i)}, x^{(j)})$ $K_{ji} = K_1(x^{(j)}, x^{(i)}) K_2(x^{(j)}, x^{(i)})$

$= K_1(x^{(i)}, x^{(j)}) K_2(x^{(i)}, x^{(j)})$ $K_1 K_2$ are kernels

$= K_{ij}$.

Thus matrix K is symmetric.

For an arbitrary $z \in \mathbb{R}^n$

$zKz = \sum_i \sum_j z_i K_{ij} z_j = \sum_i \sum_j (z_i k_{1ij} z_j)(K_{2ij}) = \sum_i \sum_j \sum_a \sum_b (z_i k_{1ij} z_j)(1 \cdot K_{2ab} \cdot 1)$ when $a=i$ $b=j$

or $(0 \cdot K_{2ab} \cdot 0)$ when $a \neq i$ or $b \neq j$

(1). we know $K_{1ij} = k_{1ji}$ and $k_{2ij} = K_{2ji}$

Thus $z_j K_{1ji} z_i K_{2ji} = z_i k_{1ij} z_j k_{2ij}$

Thus $c^T K_2 d = d^T K_2 c$

$= \sum_i \sum_j (z_i k_{1ij} z_j)(\underset{c^T}{} K_2 \underset{d}{})$ while c is $\mathbb{R}^n$ vector with ith term being 1 and others being 0.

d is $\mathbb{R}^n$ vector with jth term being 1 and others veing 0.

Then $\sum_i \sum_j (z_i k_{1ij} z_j)(c^T K_2 d) = \sum_i \sum_j (z_i k_{1ij} z_j)\left(\underbrace{\sum_i \sum_j (c^T K_2 d)^2}_{\text{when } i>j} + \underbrace{\sum_i \sum_j c^T K_2 d}_{\text{when } i=j}\right)$

since k_2 is kernel $c^T K_2 d \geq 0$ for $c = d$

also, $(c^T k_2 d)^2 \geq 0$. also $\sum_i \sum_j z_i k_{ij} z_j \geq 0$.

Thus $z^T K z \geq 0$. Thus K is positive semidefinite (f). $K_{ij} = K(x^{(i)}, x^{(j)}) = f(x^{(i)}) f(x^{(j)}) = f(x^{(j)}) f(x^{(i)}) = K(x^{(j)}, x^{(i)}) = K_{ji}$ K is symmetric.

$$Z^T K Z = \sum_i \sum_j z_i K_{ij} z_j = \sum_i \sum_j z_i f(x^{(i)}) f(x^{(j)}) z_j$$

since $z_i f(x^{(i)}) f(x^{(j)}) z_j = z_j f(x^{(j)}) f(x^{(i)}) z_i$ $= \sum_i \sum_{j>i} \left(z_i f(x^{(i)}) f(x^{(j)}) z_j\right)^2 + \sum_i \sum_{j=i} z_i f(x^{(i)}) f(x^{(i)}) z_i$

Thus $Z^T K Z \geq 0$. Thus K is positive semidefinite.

(g). $K_{ij} = K_3(\phi(x^{(i)}), \phi(x^{(j)})) = K_3(\phi(x^{(j)}), \phi(x^{(i)})) = K_{ji}$ K is symmetric.

$Z^T K Z = \sum_i \sum_j z_i k_{ij} z_j = \sum_i \sum_j z_i K_{3ij} z_j$ since K_3 is a kernel then $\sum_i \sum_j z_i k_{3ij} z_j \geq 0$.

Thus $Z^T K Z \geq 0$. K is positive semidefinite.

(h). $K_{ij} = p(k_1(x^{(i)}, x^{(j)})) = p(k_1(x^{(j)}, x^{(i)})) = K_{ji}$ K is symmetric.

$$Z^T K Z = \sum_i \sum_j z_i \, p(K_1(x^{(i)}, x^{(j)})) z_j = \sum_i \sum_j z_i \left(a_1 k_1(x^{(i)}, x^{(j)}) + a_2 k_1(x^{(i)}, x^{(j)}) + \cdots + a_p k_1(x^{(i)}, x^{(j)})\right) z_j$$

$$= \sum_i \sum_j \sum_p z_i \, a_p k_1(x^{(i)}, x^{(j)}) z_j = \sum_p a_p \sum_i \sum_j z_i k_1(x^{(i)}, x^{(j)}) z_j$$

Since k_1 is kernel $\sum_i \sum_j z_i k_1(x^{(i)}, x^{(j)})^K z_j \geq 0$.

Since $a_p \geq 0$, the whole term ≥ 0. $Z^T K Z \geq 0$. K is positive semidefinite.

$$\boxed{\text{There is a missing power in } K_1(x^{(i)}, x^{(j)})^K}$$

5. (a)(i). use a map to represent infinite dimensional vector. index : value.

where index represent the position of the entry, non-recorded entries are 0.

$\theta^{(0)}$ is the empty map.

(ii). $h_{\theta^{(i)}}(x^{(i+1)}) = \sum_{\substack{key \in \theta^{(i)} \text{ map and} \\ \phi(x^{(i+1)}) \text{ map}}} \theta^{(i)}[key] \cdot \phi(x^{(i+1)})[key]$.

(iii). $\theta^{(i+1)} := \theta^{(i)} + \alpha\left(y^{(i+1)} - \text{sign}\left(\sum_{key} \theta^{(i)}[key]\, \phi(x^{(i+1)})[key] \right)\right) \phi(x^{(i+1)})$

(a): (i). use a zero vector whose dimension is same as x^i to represent $\theta^{(0)}$

$\theta^{(i)}$ is represented in the same manner s.t. $\theta^{(i)} = \phi(\theta^{(i)}_{true})$.

(ii). since $\theta^{(i)T} \phi(x^{(i+1)})$ is an inner product. we can represent it as kernel trick.

Using $K(\theta^{(i)}_{true}, x^{(i+1)})$ to represent it.

(iii). $\theta^{(i+1)}_{true} := \theta^{(i)}_{true} + \alpha\left(y^{(i+1)} - K(\theta^{(i)}_{true}, x^{(i+1)})\right) x^{(i+1)}$.

~~(a). (i). we know that $\theta^{(i)}$ is a linear combination of $\phi(x_0), \phi(x_1) \dots \phi(x_i)$~~

~~we can record the coefficients of $\phi(x)$ and number i~~

~~the x_k entry can be got by i % n where n is total number of x.~~

~~the coefficents can be got as vector C[ ]. also α learning rate.~~

~~$\theta^{(i)} = \alpha \sum_j^i b^{(j)} \phi(x^{(j)})$ since ϕ is a known mapping. recording $x^{(1)}$ to $x^{(i)}$ will get~~

~~sufficient to compute $\phi(x^{(1)})$ to $\phi(x^{(i)})$.~~

~~$\theta^{(i)} =$~~

5.(a).(i). $\theta^{(i)}$ can be represented as a linear combination of $\phi(x_1), \phi(x_2), \ldots, \phi(x_i)$.

$\theta^{(i)} = \sum_{k}^{i} b^{(k)} \phi(x^{(k)})$. ~~Than~~ Since we know the ϕ mapping will not change.

$\phi(x^k)$ can be represented as x^k. Assume there are n ~~inputs~~ inputs.

we can record K and use it as $K \% n$ to represent index of x.

We also need to record $b^{(k)}$ as an array of coefficents.

Knowing $\vec{b}$, i are enough to represent $\theta^{(i)}$.

$\theta^{(0)}$ can be represented as $i=0$ $\vec{b}=0$.

(ii). $h_{\theta^{(i)}}(x^{(i+1)}) = g(\theta^{(i)T} \phi(x^{(i+1)})) = g\left(\sum_{k}^{i} b^{(k)} \phi(x^{(k)}) \cdot \phi(x^{(i+1)})\right)$

$$= g\left(\sum_{k}^{i} b^{(k)} \left[\phi(x^{(k)}) \cdot \phi(x^{(i+1)})\right]\right).$$

$$= g\left(\sum_{k}^{i} b^{(k)} K(x^{(k)}, x^{(i+1)})\right).$$

(iii). $b^{i+1} = y^{(i+1)} - h_{\theta^{(i)}}(x^{(i+1)})$.

put b^{i+1} to the ~~e~~end of $\vec{b}$, and $i := i+1$

(c). the dot product kernel performs poorly because the data is not linearly separatedble in two-dim. rbf raises it to a higher dimension which makes it separatable.