Problem 1

(1) set a random variable $X \in [-2, 2]$. $t = 1$

$Pr[X \geq t] = \frac{1}{4} \geq 0 = \frac{E[X]}{t}$.

so Markov's inequality doesn't hold for negative variable.

(2) For chebyshev's Inequality : $Pr[|X - E(X)| \geq t] \leq \frac{Var(X)}{t^2}$, $\forall t > 0$

$\because E(\bar{X}) = E(X)$, $Var(\bar{X}) = E[(\frac{1}{n}\sum_{i=1}^{n} x_i)^2] = \frac{Var(X)}{n}$ for X iid and $E(X)=0$

$\therefore Pr[|\bar{X} - E(X)| \geq t] \leq \frac{Var(X)}{nt^2}$, so large n and small variance imply better concentration.

(3) Hoeffding's inequality has tighter bound because it's decay exponetially $O(e^{-t^2})$

4). Only when all samples are i.i.d, we have $Erout(f) = E_{sup}[Erin(f)]$, which means we can use $Erin(f)$ as unbiased estimator of $Erout(f)$.

5). From $Erout(f) \leq Erin(f) + R(H) + \sqrt{\frac{\log\frac{1}{\delta}}{2n}}$ we know, more samples (bigger n) , tighter bound, smaller error.

6). We know $\hat{R}_s(H) := E_\epsilon [\sup_{f_\theta \in H} \frac{1}{n}\sum_{i=1}^{n} \epsilon_i f_\theta(x_i)]$

when $|H| = 1$, $\hat{R}_s(H) = E_\epsilon[\frac{1}{n}\sum_{i=1}^{n} \epsilon_i f_\theta(X_i)] = \sum_{i=1}^{n} f_\theta(X_i) E(\epsilon_i) = 0$

when $|H| = 2^n$, $\hat{R}_s(H) = E_\epsilon[\sup_{f \in H} \frac{1}{n}\sum_{i=1}^{n} \epsilon_i f(x_i)] = E_\epsilon[\frac{1}{n}\sum_{i=1}^{n} 1] = E[1] = 1$

7) Firstly decide a not so complex hypothesis space H, then choose one optimization algorithm to get $f_{\hat{\theta}} \in H$, which minimize our designed loss function. So we can have small generalization error.

Problem 2. Define $h(S) := \sup_{f \in H} [Erout(f) - Erin(f; S)]$

Let $S = \{z_1, z_2, \dots z_i \dots, z_n\}$, $S' = \{z_1, \dots z_i' \dots z_n\}$

$h(S') - h(S) = \sup_{f \in H} [Erout(f) - Erin(f; S')] - \sup_{f \in H} [Erout(f) - Erin(f; S')]$

$\leq \sup [Erin(f; S) - Erin(f; S')]$

$= \sup_{f \in H} [\frac{e(f(x_i), y_i) - e(f(x_i'), y_i')}{n}]$

$\leq \frac{1}{n}$ since $e(f(x), y) \to [0, 1]$

Apply McDiarmid's inequality to $h(S)$, we have

$Pr[h(S) - E[h(S)] \geq t] \leq e^{-2nt^2}, \forall t > 0$

Let $\delta = e^{-2nt^2}$, we then have with probability at least $1-\delta$

$$h(S) \leq E[h(S)] + \sqrt{\frac{\log\frac{1}{\delta}}{2n}}$$

since $h(S) = \sup_{f\in H} [Er_{out}(f) - Er_{in}(f)] \geq Er_{out}(f) - Er_{in}(f)$ for any $f \in H$

we get $\forall f \in H$, $Er_{out}(f) \leq Er_{in}(f) + E(h(S)) + \sqrt{\frac{\log\frac{1}{\delta}}{2n}}$

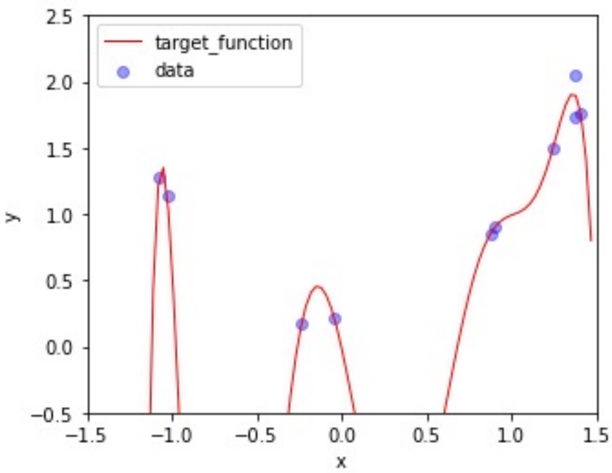
$\Longrightarrow$ so $Er_{out}(f) \leq Er_{in}(f) + R(H) + \sqrt{\frac{\log\frac{1}{\delta}}{2n}}$

$E[h(S)] \leq R(H)$

Problem3

(a1): y is an $[10,1]$ matrix, while X is an $[10,9]$ vandermonde matrix.

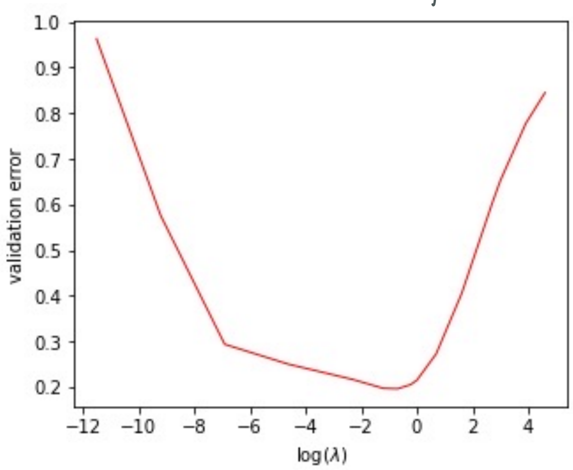
(a2):



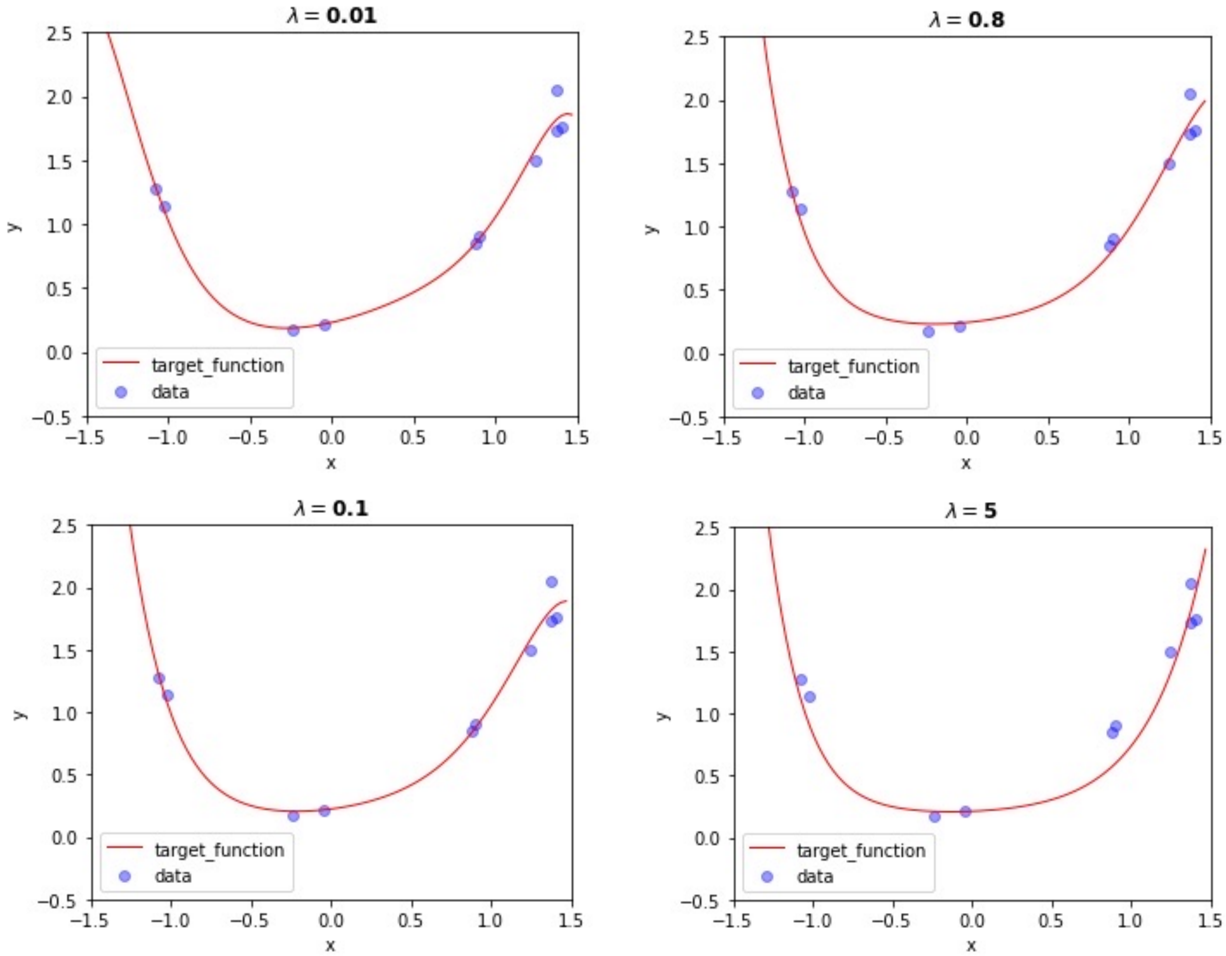
the overfitting is very obvious.

(a3) the test error is = 177.2312089244576

(b1) the plot of validation error versus the value of λ is as follow:



(b2) the fitted curve using the four choice of λ is:



(b3) the test error for $\lambda = 0.01, 0.1, 0.8$, and 5 is $(1.21, 3.69, 4.11, 3.49)$

Problem4

(a) the loss function can be define as:

$$l(\theta) := \sum_{n=1}^{N} \left[\theta_{y_n}^{T} X_n - \log \sum_{c=1}^{Class} \exp(\theta_c^{T} X_n) \right]$$

for the first term:

$$\frac{\partial \theta_{y_n}^{T} X_n}{\partial \theta_{y_n}^{(j)}} \theta_{y_n}^{T} X_n := [y_n = y'] X_n^{(j)})$$

for the second term:

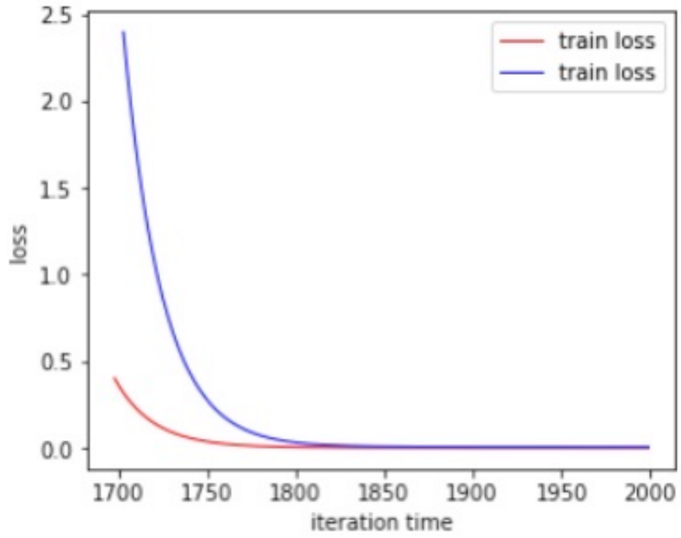
$$\frac{\partial}{\partial \theta_{y'}^{(j)}} \log \sum_{c=1}^{Class} \exp(\theta_c^{T} X_n) := \frac{\sum_{c=1}^{C} \exp(\theta_c^{T} X_n) \times [y = y'] X_n^{(j)}}{\sum_{c=1}^{C} \exp(\theta_c^{T} X_n)}$$

$$= \sum_{c=1}^{C} \left[\frac{\exp(\theta_c^{T} X_n)}{\sum_{c=1}^{C} \exp(\theta_c^{T} X_n)} \times [y_n = y'] X_n^{(j)} \right]$$

$$= \sum_{c=1}^{C} \left[p(y = c \mid X_n) \times [y_n = y'] X_n^{(j)} \right]$$

$$= p(y = y' \mid X_n) X_n^{(j)}$$

Applying AGD we can get test loss and training loss decreasing as follow:



Then, we can further calculate the training error is 56.31%, and test accuracy is 43.58%. Compared with random probility, which is $\frac{1}{4} = 25\%$, multinomial logistic regression obviously has higher odds ratio.

Finally, comparing the corresponding parameters' norm, we find the most important feature is "RAM".

P.S: for more details, please see "p4. ipynb" in my folder.

1b) The learning problem can be formulated as:

$$\min_{\theta, Z} L(\Theta, Z) := -\frac{1}{n} \cdot \mathbb{1}_{1\times n} \cdot \{ P \odot \log(h(X \cdot \Theta \cdot \mathbb{1}_{1\times K} - \mathbb{1}_{n\times 1} \cdot Z^T_{1\times K}) \cdot M_{KK}) \} \cdot \mathbb{1}_{K\times 1}$$

$$\theta : R^{d\times 1}, \quad Z : R^{K\times 1}$$

$$P : R^{n\times k} \quad P_{ij} := \begin{cases} 1, & \text{if } f_i = j \\ 0, & \text{others} \end{cases}$$

$$M : R^{K\times K} := \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

then, we get the grdient as:

$$\frac{\partial L(\theta, Z)}{\partial \theta} := -\frac{1}{n} \cdot (\mathbb{1}_{1\times K} \otimes \mathbb{1}_{1\times n}) \cdot \mathrm{diag}(\mathrm{vec}(P)) \cdot Q_a \cdot Q_{\log} \cdot (M^T \otimes \mathbb{I}_{n\times n}) \cdot Q_h \cdot (\mathbb{1}_{K\times 1} \otimes X)$$

$$\frac{\partial L(\theta, Z)}{\partial Z} := \frac{1}{n} \cdot (\mathbb{1}_{1\times K} \otimes \mathbb{1}_{1\times n}) \cdot \mathrm{diag}(\mathrm{vec}(P)) \cdot Q_a \cdot Q_{\log} \cdot (M^T \otimes \mathbb{I}_{n\times n}) \cdot Q_h \cdot (\mathbb{1}_{K\times 1} \otimes \mathbb{I}_{K\times K})$$

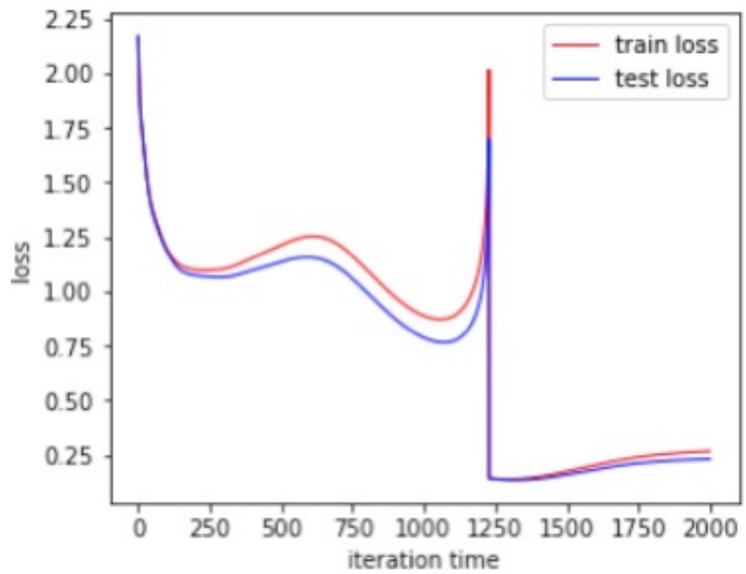
where $H : h(X \cdot \Theta \cdot \mathbb{1}_{1\times K} - \mathbb{1}_{n\times 1} \cdot Z.T) \quad R^{n\times d}$

$$Q_a := diag(vec(P)) \qquad R^{nK \times nK}$$

$$Q_{log} := diag(vec(H \otimes M)) \qquad R^{nK \times nK}$$

$$Q_h := diag(vec(H')) \qquad R^{nK \times nK}$$

By applying AGD, we can get loss decreasing as follow:



And the accarary for train data is 76.66%, while the accarary for train data is 73%. Both of them get improved greatly by ordinal logistic regression. So we can say, for ordinal label, it's better to use ordinal logistic regression than multinomial logestic regression.

As before, comparing the corresponding parameters norm, we find the most important feature is "RAM".