ESE 546, FALL 2023

HOMEWORK 4

KEQI WU [KEQIWU@SEAS.UPENN.EDU],
COLLABORATORS: NONE

Solution 1 (Time spent: 15 mins)**.** Your solution goes here.

(a)
Proof:
The inner product is non-negative because $f(x)$ is convex, therefore, by Cauchy-Schwartz inequality:

$$\langle \boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y), x - y\rangle = |\langle \boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y), x - y\rangle| \leq ||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||||x - y||$$

So,

$$\begin{aligned}\frac{1}{L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \leq \langle \boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y), x - y\rangle \leq ||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||||x - y|| \\ \frac{1}{L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \leq ||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||||x - y|| \\ ||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)|| \leq L||x - y||\end{aligned}$$

Therefore, coercivity implies Lipschitz continuity.

(b)

Proof:

Because f is convex and differentiable in the domain $\mathbb{R}^d$, and the gradient of f is L-Lipschitz, therefore, f is L-smooth in the domain $\mathbb{R}^d$.

Define $g(x) = f(x) - \langle \boldsymbol{\nabla} f(y), x\rangle$,which is also smooth, in this case g is minimized at y because $\boldsymbol{\nabla} g(y) = \boldsymbol{\nabla} f(y) - \boldsymbol{\nabla} f(y) = 0$. Therefore, $g(y) \leq g(x) \forall x$.

Since g is smooth, by Descent Lemma:

$$\begin{aligned} g(x - \frac{1}{L}\boldsymbol{\nabla} g(x)) &\leq g(x) + \langle \boldsymbol{\nabla} g(x), -\frac{1}{L}\boldsymbol{\nabla} g(x)\rangle + \frac{L}{2}|| - \frac{1}{L}\boldsymbol{\nabla} g(x)||^2 \\ &= g(x) - \frac{1}{L}||\boldsymbol{\nabla} g(x)||^2 + \frac{L}{2}(\frac{1}{L})^2||\boldsymbol{\nabla} g(x)||^2 \\ &= g(x) - \frac{1}{L}||\boldsymbol{\nabla} g(x)||^2 + \frac{1}{2L}||\boldsymbol{\nabla} g(x)||^2 \\ &= g(x) - \frac{1}{2L}||\boldsymbol{\nabla} g(x)||^2 \end{aligned}$$

Because y minimizes g, then $g(y) \leq g(x - \frac{1}{L}\boldsymbol{\nabla} g(x)) \leq g(x) - \frac{1}{2L}||\boldsymbol{\nabla} g(x)||^2$

Substitute the definition of g into the inequality:

$$\begin{aligned} &f(y) - \langle \boldsymbol{\nabla} f(y), y\rangle \leq f(x) - \langle \boldsymbol{\nabla} f(y), x\rangle - \frac{1}{2L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \\ &f(y) - \langle \boldsymbol{\nabla} f(y), y\rangle - f(x) + \langle \boldsymbol{\nabla} f(y), x\rangle \leq -\frac{1}{2L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \\ &f(y) - f(x) + \langle \boldsymbol{\nabla} f(y), x - y\rangle \leq -\frac{1}{2L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \text{ by the property of inner product} \end{aligned}$$

Exchange x and y, we have:

$$\begin{aligned} &f(x) - \langle \boldsymbol{\nabla} f(x), x\rangle - f(y) + \langle \boldsymbol{\nabla} f(x), y\rangle \leq -\frac{1}{2L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \\ &f(x) - f(y) - \langle \boldsymbol{\nabla} f(x), x - y\rangle \leq -\frac{1}{2L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \end{aligned}$$

Combing two results, we have:

$$\begin{aligned} &f(y) - f(x) + \langle \boldsymbol{\nabla} f(y), x - y\rangle + f(x) - f(y) - \langle \boldsymbol{\nabla} f(x), x - y\rangle \leq -\frac{1}{L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \\ &- \langle \boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y), x - y\rangle \leq -\frac{1}{L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \text{ by the property of inner product} \\ &\langle \boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y), x - y\rangle \geq \frac{1}{L}||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||^2 \end{aligned}$$

Therefore, Lipschitz continuity implies coercivity.

(c)

Since f is twice differentiable and the gradient is L-Lipschitz, by mean value theorem, there exist a z between every pair of x and y, such that $\boldsymbol{\nabla}^2 f(z) = \frac{\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)}{x-y}$.

Because the gradient is L-Lipschitz, $||\boldsymbol{\nabla} f(y) - \boldsymbol{\nabla} f(x)||_2 \leq L||y-x||_2$, then:

$$\begin{aligned}
\boldsymbol{\nabla}^2 f(x) &= \lim_{\epsilon \to 0} \frac{\boldsymbol{\nabla} f(x+\epsilon) - \boldsymbol{\nabla} f(x)}{\epsilon} \\
||\boldsymbol{\nabla}^2 f(x)||_2 &\leq \lim_{\epsilon \to 0} \frac{||\boldsymbol{\nabla} f(x+\epsilon) - \boldsymbol{\nabla} f(x)||_2}{|\epsilon|} \\
&\leq \lim_{\epsilon \to 0} \frac{L|\epsilon|}{|\epsilon|} \text{ by L-Lipschitz} \\
&= L
\end{aligned}$$

Since f is m-strongly convex, $g(x) = f(x) - \frac{M}{2}||x||_2^2$ is convex. By Monotonicity of the gradient for convex functions, $\langle \boldsymbol{\nabla} g(x) - \boldsymbol{\nabla} g(y), x-y \rangle \geq 0$, then:

$$\begin{aligned}
\langle \boldsymbol{\nabla} g(x) - \boldsymbol{\nabla} g(y), x-y \rangle &\geq 0 \\
(\boldsymbol{\nabla} g(x) - \boldsymbol{\nabla} g(y))^T (x-y) &\geq 0 \\
(\boldsymbol{\nabla} f(x) - Mx - \boldsymbol{\nabla} f(y) + My)^T (x-y) &\geq 0 \\
(\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y))^T (x-y) - M||x-y||_2^2 &\geq 0 \\
(\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y))^T (x-y) &\geq M||x-y||_2^2 \\
\langle \boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y), x-y \rangle &\geq M||x-y||_2^2
\end{aligned}$$

By Cauchy-Schwartz inequality, $||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||_2 ||x-y||_2 \geq \langle \boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y), x-y \rangle \geq M||x-y||_2^2$, therefore $||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||_2 \geq M||x-y||_2$.

By mean value theorem, $||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||_2 \leq ||x-y||_2 ||\boldsymbol{\nabla}^2 f(z)||_2$, then, $M||x-y||_2 \leq ||\boldsymbol{\nabla} f(x) - \boldsymbol{\nabla} f(y)||_2 \leq ||x-y||_2 ||\boldsymbol{\nabla}^2 f(z)||_2$, we get: $M \leq ||\boldsymbol{\nabla}^2 f(z)||_2$

Combining two results, we have $M \leq ||\boldsymbol{\nabla}^2 f(x)||_2 \leq L$

Solution 2 (Time spent: 5 hrs)**.** Your solution goes here.

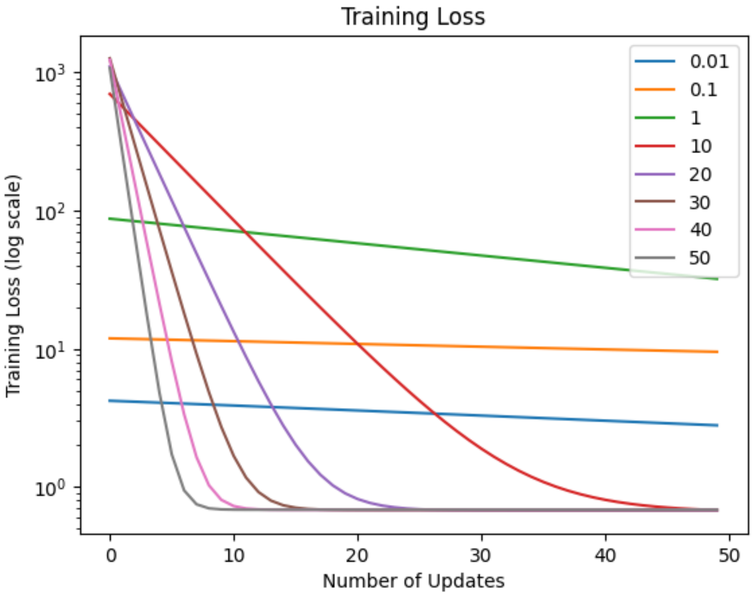
(a)

Check code

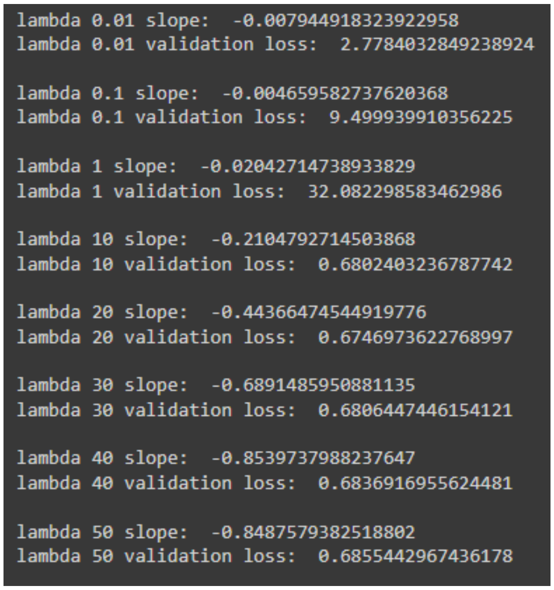
(b)

Check code

Training Loss:



Slope:



```
lambda 0.01 slope:  -0.007944918323922958
lambda 0.01 validation loss:  2.7784032849238924

lambda 0.1 slope:  -0.004659582737620368
lambda 0.1 validation loss:  9.499939910356225

lambda 1 slope:  -0.02042714738933829
lambda 1 validation loss:  32.082298583462986

lambda 10 slope:  -0.2104792714503868
lambda 10 validation loss:  0.6802403236787742

lambda 20 slope:  -0.44366474544919776
lambda 20 validation loss:  0.6746973622768997

lambda 30 slope:  -0.6891485950881135
lambda 30 validation loss:  0.6806447446154121

lambda 40 slope:  -0.8539737988237647
lambda 40 validation loss:  0.6836916955624481

lambda 50 slope:  -0.8487579382518802
lambda 50 validation loss:  0.6855442967436178
```

We will pick $\lambda = 10$, $k = -\frac{1}{slope} = -\frac{1}{-0.21048} = 4.7510$

(c)

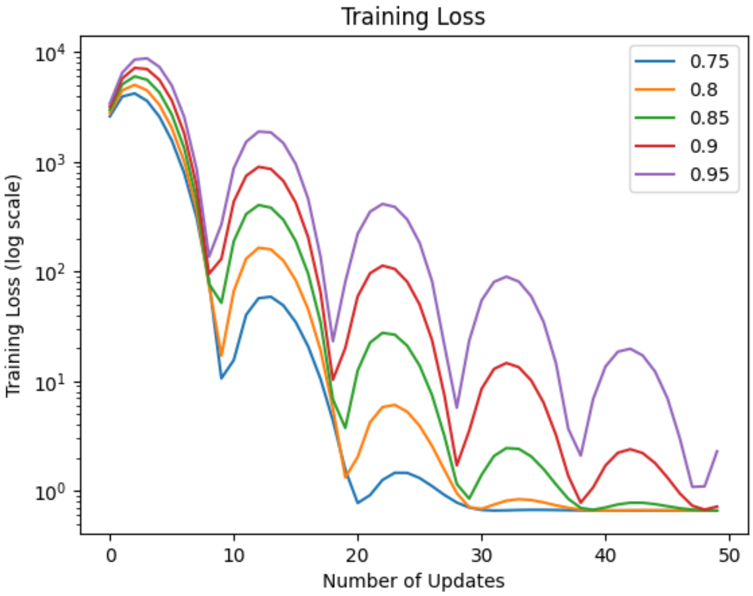
$$
\begin{aligned}
\frac{\partial l}{\partial w} &= \frac{1}{n}\sum_{i=1}^{n}\frac{y_i x_i e^{-y_i(w^T x_i+w_0)}}{1+e^{-y_i(w^T x_i+w_0)}} + \lambda w \\
\frac{\partial^2 l}{\partial w^2} &= \frac{1}{n}\sum_{i=1}^{n}\frac{y_i^2 x_i x_i^T e^{-y_i(w^T x_i+w_0)}}{(1+e^{-y_i(w^T x_i+w_0)})^2} + \lambda \\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{x_i x_i^T e^{-y_i(w^T x_i+w_0)}}{(1+e^{-y_i(w^T x_i+w_0)})^2} + \lambda \\
&= X^T D X + \lambda I \text{ where } D = diag\frac{1}{n}\frac{e^{-y_i(w^T x_i+w_0)}}{(1+e^{-y_i(w^T x_i+w_0)})^2}
\end{aligned}
$$

To check if the equation is strongly convex, we need to check if $(X^T DX + \lambda I) - \mu I$ is positive semidefinite for strong convexity parameter μ. In this case, since the entry in D is not greater than $\frac{1}{4}$, we must have $\mu \leq \frac{1}{4n}\lambda_{min}(XX^T) + \lambda$ where λ_{min} is the smallest eigenvalue of XX^T. In the best case, such smallest eigenvalue is 0, therefore, the best strongly convexity parameter is $lambda$

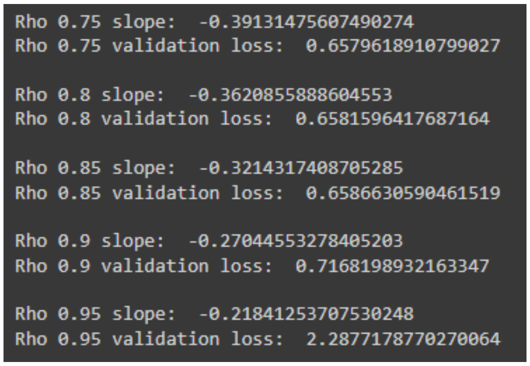
(d)

Check Code

Training Loss:



Slope m:

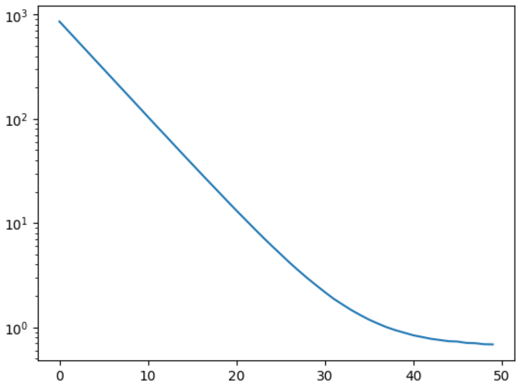


```
Rho 0.75 slope:  -0.39131475607490274
Rho 0.75 validation loss:  0.6579618910799027

Rho 0.8 slope:  -0.3620855888604553
Rho 0.8 validation loss:  0.6581596417687164

Rho 0.85 slope:  -0.3214317408705285
Rho 0.85 validation loss:  0.6586630590461519

Rho 0.9 slope:  -0.27044553278405203
Rho 0.9 validation loss:  0.7168198932163347

Rho 0.95 slope:  -0.21841253707530248
Rho 0.95 validation loss:  2.2877178770270064
```

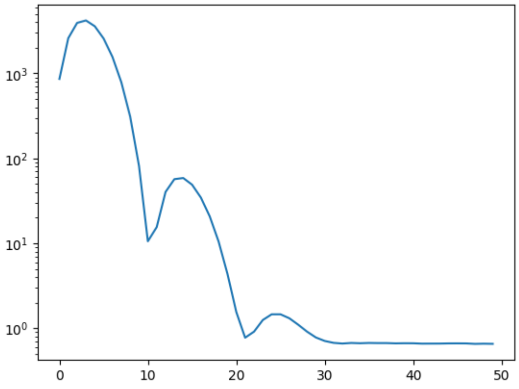
We will pick $\rho = 0.75$ the best slope $m = -0.391315$, therefore, $k = (\frac{1}{-m})^2 = 6.5305$

(e)

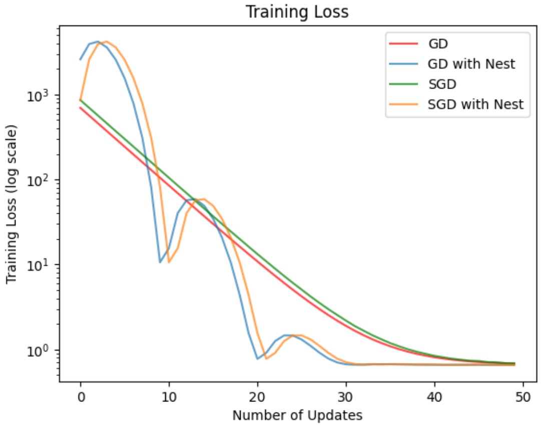
SGD:



SGD with Nest:



Combined:



Nesterov method does accelerate convergence rates of GD and SGD. (iii) is faster than (ii). Convergence of (ii) is slower than convergence of (i). Although the loss of (i) clearly increases for some updates, it has more downward momentum so that the loss drops faster than (i).

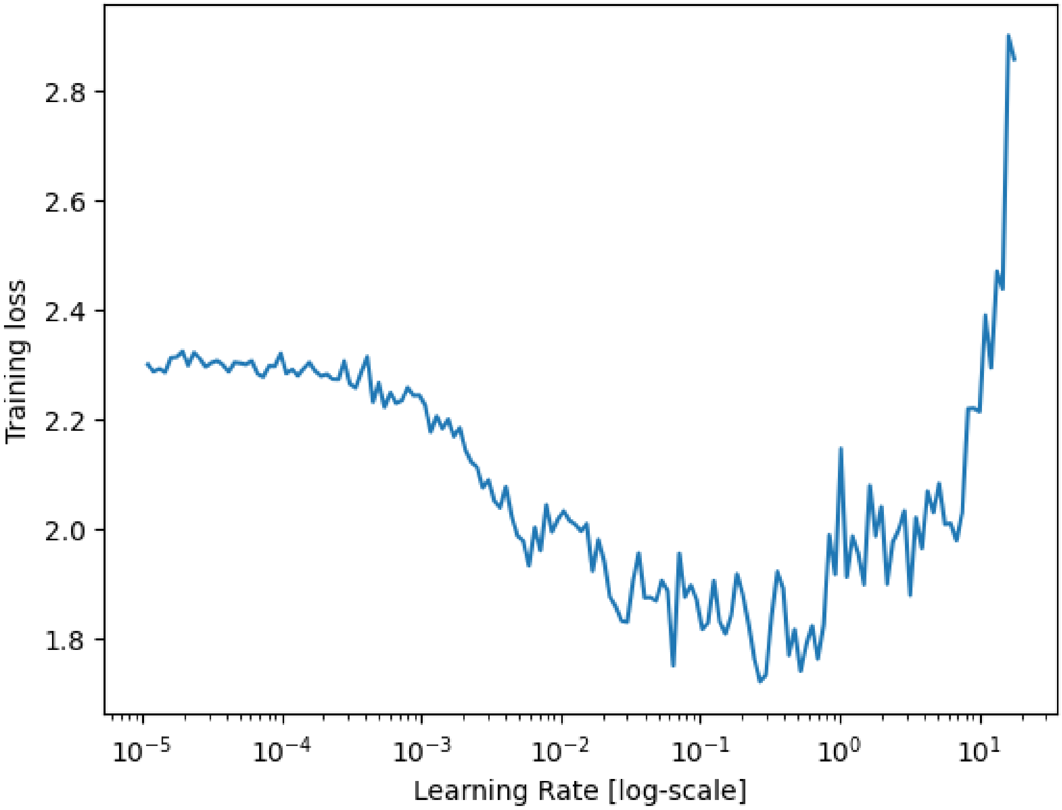
Solution 3 (Time spent: 8 hrs)**.** Your solution goes here.

(a)

This is because we are using logits with softmax and there are 10 classes in total in the dataset, therfore, we have starting corss entropy loss $-ln(\frac{1}{10}) \approx 2.302$

(b)

check code

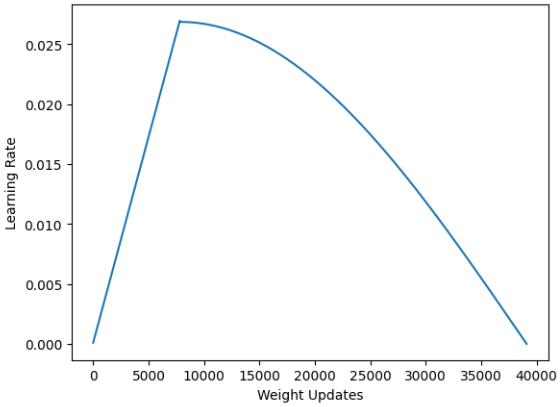


Training loss minimum achieved at: 0.268545149150829, therefore, $\eta^* = 0.268545149150829$

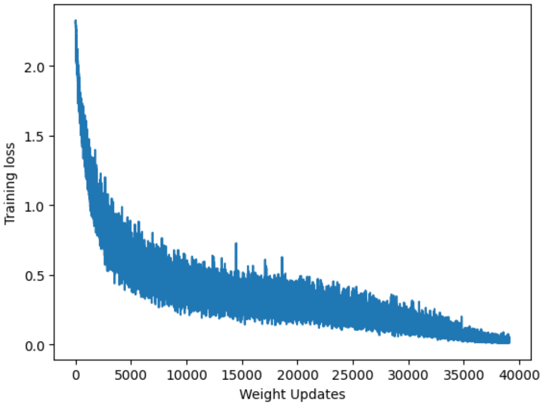
(c)

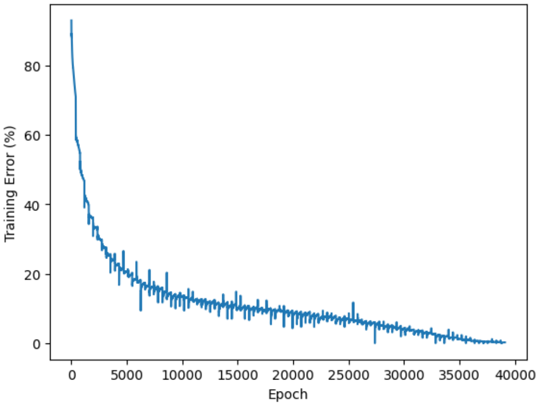
check code

Some intermediate results:
Epoch 1: Validation loss = 1.9685, validation accuracy = 28.0542%
Epoch 20: Validation loss = 0.7958, validation accuracy = 73.1435%
Epoch 50: Validation loss = 0.5074, validation accuracy = 83.7745%
Epoch 80: Validation loss = 0.4535, validation accuracy = 85.5229%
Epoch 100: Validation loss = 0.3613, validation accuracy = 89.0428%

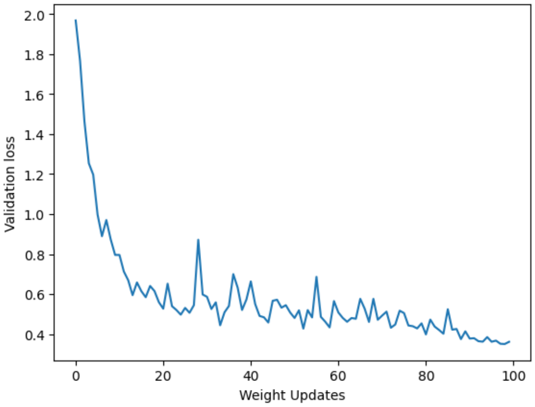


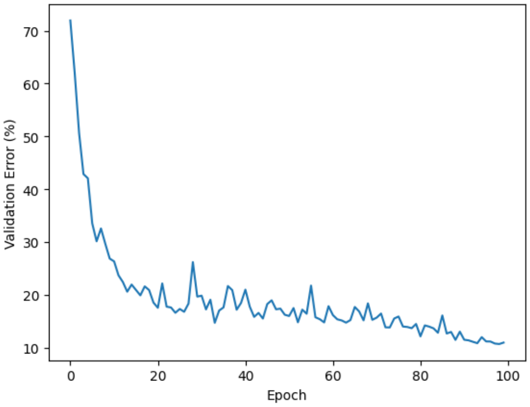
Training Part:





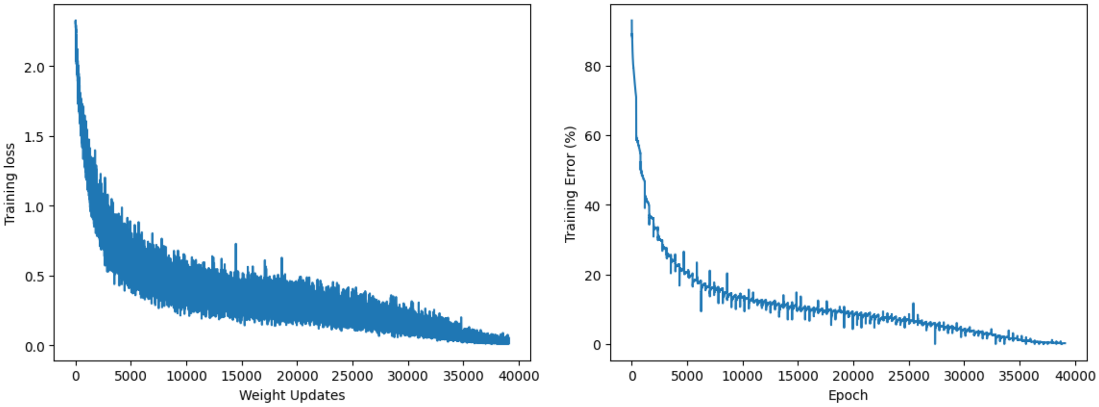
Validation Part:


Validation loss
2.0
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0
20
40
60
80
100
Weight Updates

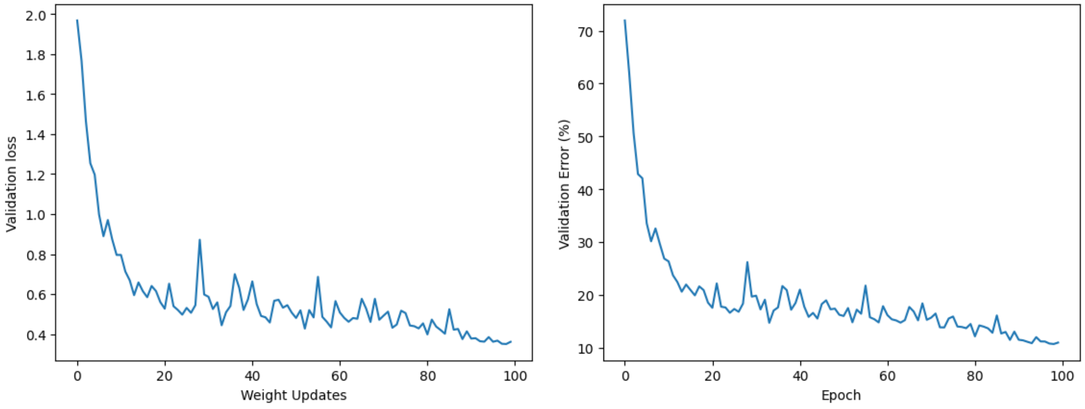

Validation Error (%)
70
60
50
40
30
20
10
0
20
40
60
80
100
Epoch

(d)

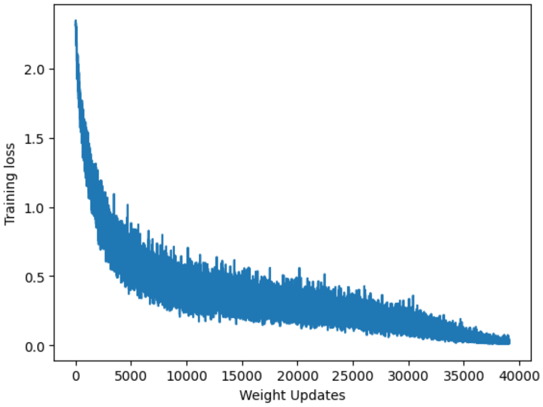
We will continue to use 100 epochs
(i) η_{max} and $\rho = 0.9$, plots inherited from (c)
Training Part:

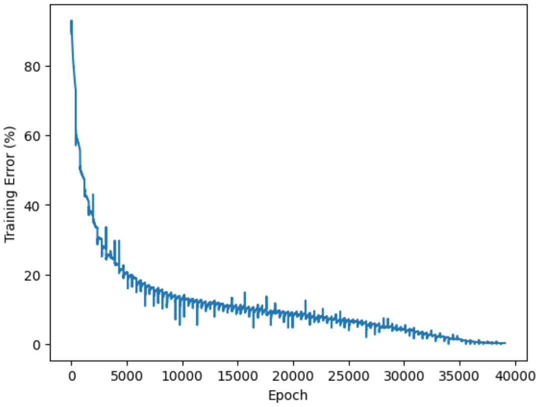


Validation Part:

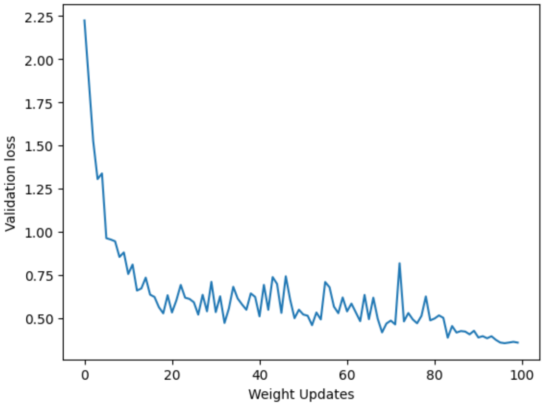


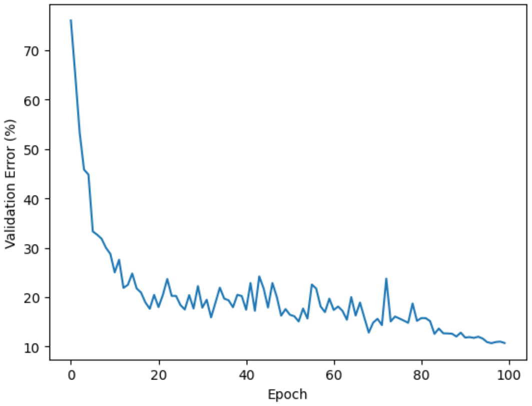
(ii) $\eta_{max} \leftarrow 5 * \eta_{max}$ and $\rho = 0.5$ Training Part:





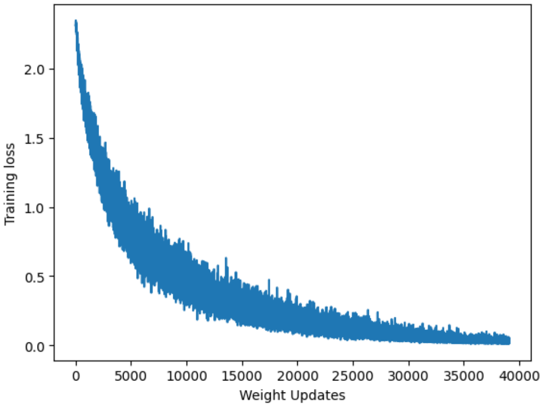
Validation Part:

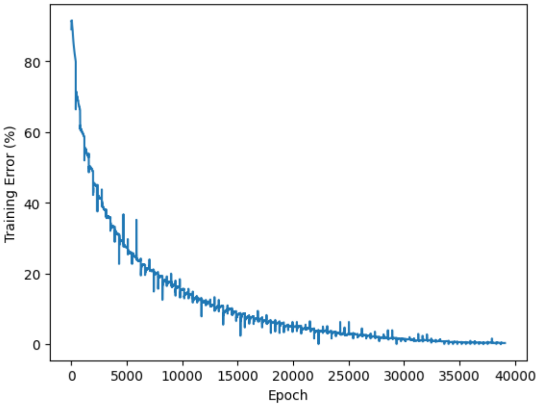




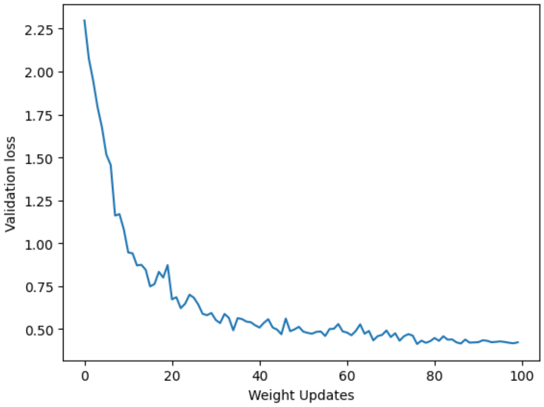
(iii) η_{max} and $\rho = 0.5$

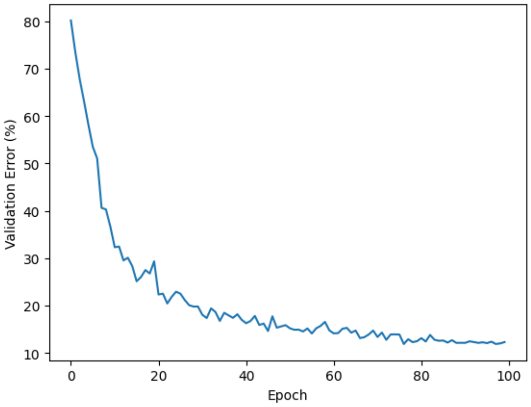
Training Part:





Validation Part:


Validation loss
2.25
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0
20
40
60
80
100
Weight Updates


Validation Error (%)
80
70
60
50
40
30
20
10
0
20
40
60
80
100
Epoch

Solution 4 (Time spent: 2 hrs)**.** Your solution goes here.

(a)

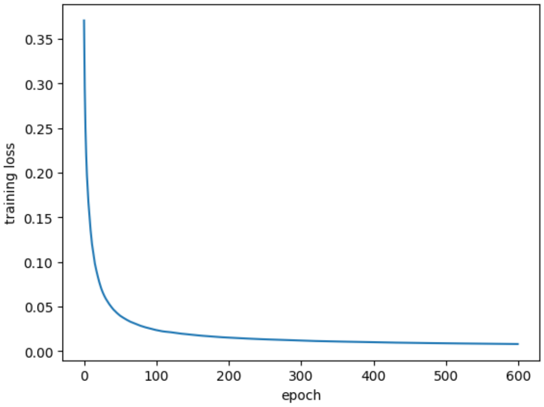
check code

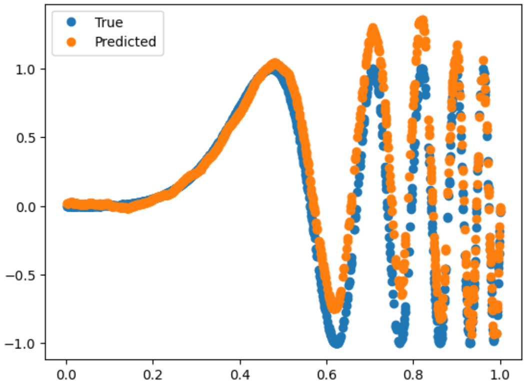
(b)

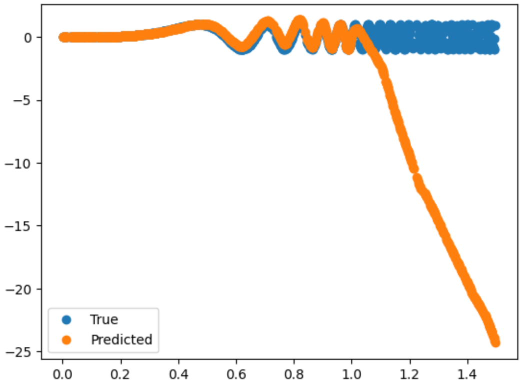
Model Architecture (from top to bottom):
Input size: 1
Hidden layer1 size: 256
ReLU()
Batch Norm(256)
Hidden layer2 size: 512
ReLU()
Batch Norm(512)
Hidden layer3 size: 256
ReLU()
Batch Norm(256)
Output size: 1

Hyperparameter:
N samples: 1000
Batch Size: 200
Learning Rate: 1e-4
Weight Decay: 1e-5
Epoch: 600

Final Loss Achieved: 0.008087085321720224

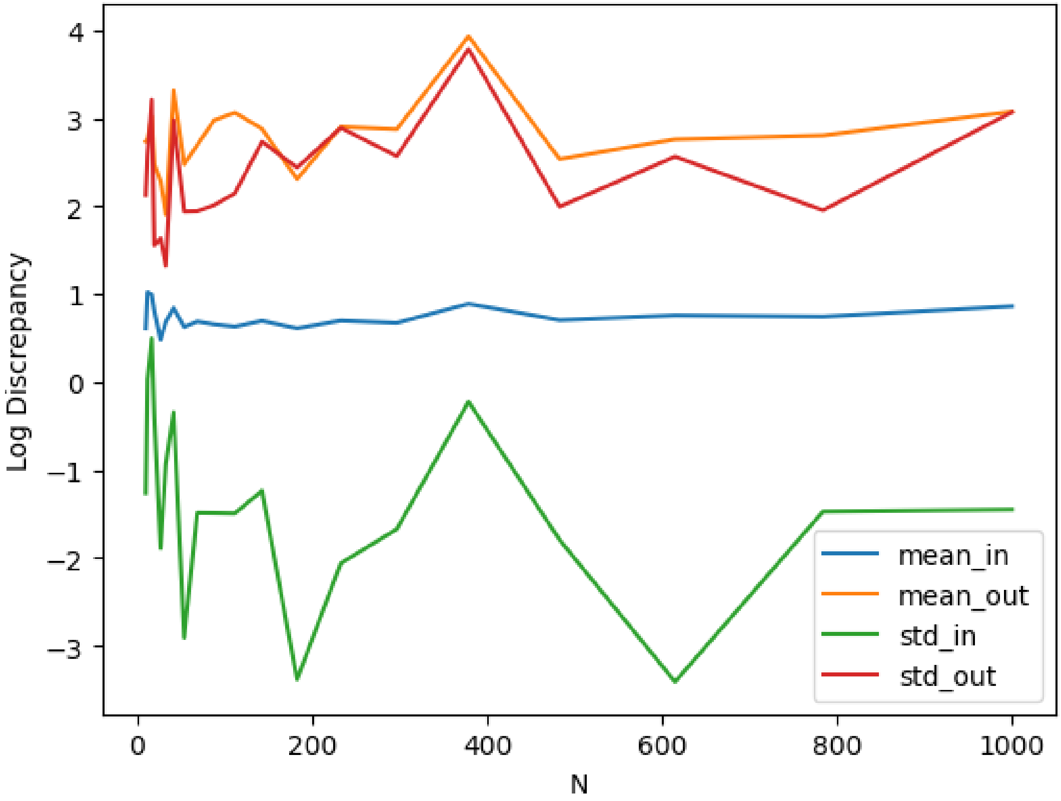

training loss
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
100
200
300
400
500
600
epoch


True
Predicted
1.0
0.5
0.0
−0.5
−1.0
0.0
0.2
0.4
0.6
0.8
1.0


0
−5
−10
−15
−20
−25
True
Predicted
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4

(c)

check code



From this plot, we see that the means and standard deviations of samples in support fall in a really small range, with log numbers below 1, and the mean values are stable. The max discrepancy of the model with respect to the true value is small. However, for out support samples, there are large variations of mean and standard deviations among different models, and numbers are generally large. The means and standard deviations are really close for different models. Therefore, our model training does not provide a good generalization for new datasets, especially for data out of range (never trained before).