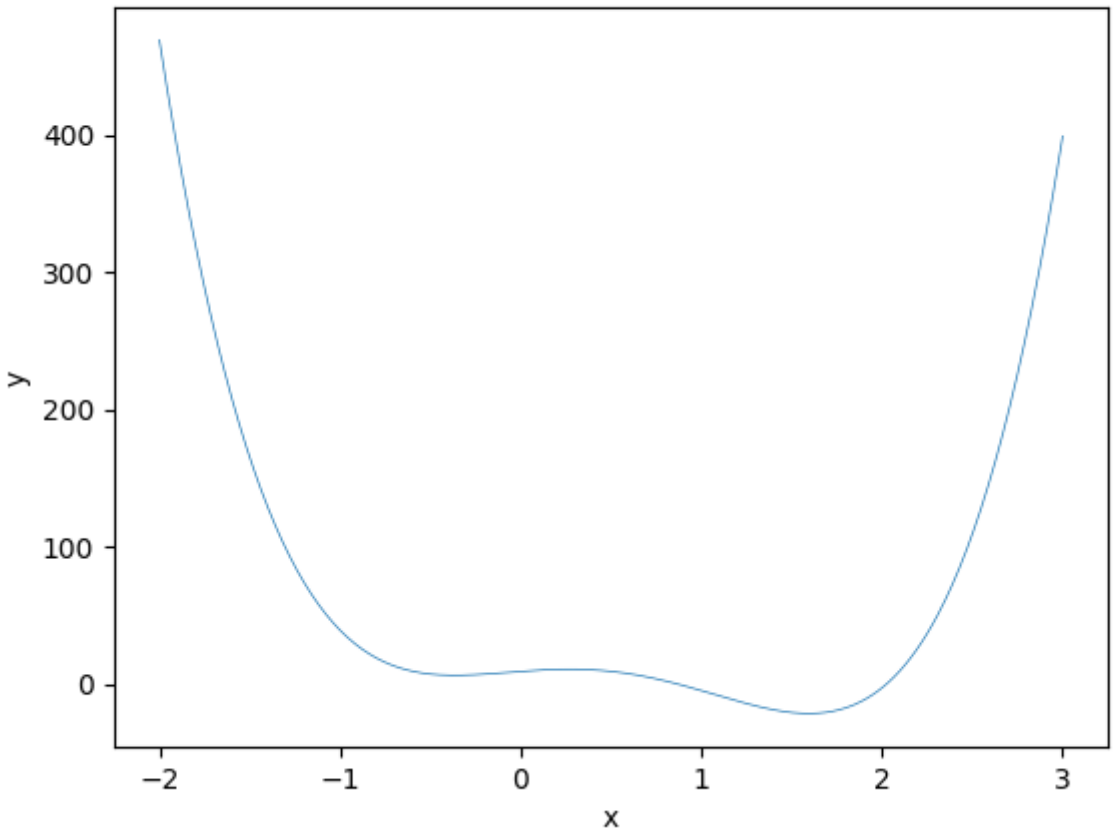
CS6923 Machine Learning

Homework 3

Shang-Hung Tsai

Part 2

1.



(a)
Local minimum = -0.364224
Global minimum = 1.595315

(b) With $x = -1, \eta = 0.001, iteration = 5$
After iteration 1, the x = -0.866000
After iteration 2, the x = -0.776295
After iteration 3, the x = -0.710922
After iteration 4, the x = -0.660782
After iteration 5, the x = -0.620972

With $x = -1, \eta = 0.001, iteration = 1000$
last 5 iterations:
After iteration 996, the x = -0.364224
After iteration 997, the x = -0.364224
After iteration 998, the x = -0.364224
After iteration 999, the x = -0.364224
After iteration 1000, the x = -0.364224

The value of x has converged at -0.364224. The gradient descent found a local minimum.

(c) With $x = 2, \eta = 0.001, iteration = 5$
After iteration 1, the x = 1.894000
After iteration 2, the x = 1.823848
After iteration 3, the x = 1.774085
After iteration 4, the x = 1.737261
After iteration 5, the x = 1.709229

With $x = 2, \eta = 0.001, iteration = 1000$
last 5 iterations:
After iteration 996, the x = 1.595315
After iteration 997, the x = 1.595315
After iteration 998, the x = 1.595315
After iteration 999, the x = 1.595315
After iteration 1000, the x = 1.595315

The value of x has converged at 1.595315. The gradient descent found a global minimum.

(d) With $x = -1, \eta = 0.01, iteration = 1000$
first 5 iterations:
After iteration 1, the x = 0.340000
After iteration 2, the x = 0.380221
After iteration 3, the x = 0.444663
After iteration 4, the x = 0.549356
After iteration 5, the x = 0.720866
last 5 iterations:
After iteration 996, the x = 1.595315
After iteration 997, the x = 1.595315
After iteration 998, the x = 1.595315
After iteration 999, the x = 1.595315
After iteration 1000, the x = 1.595315

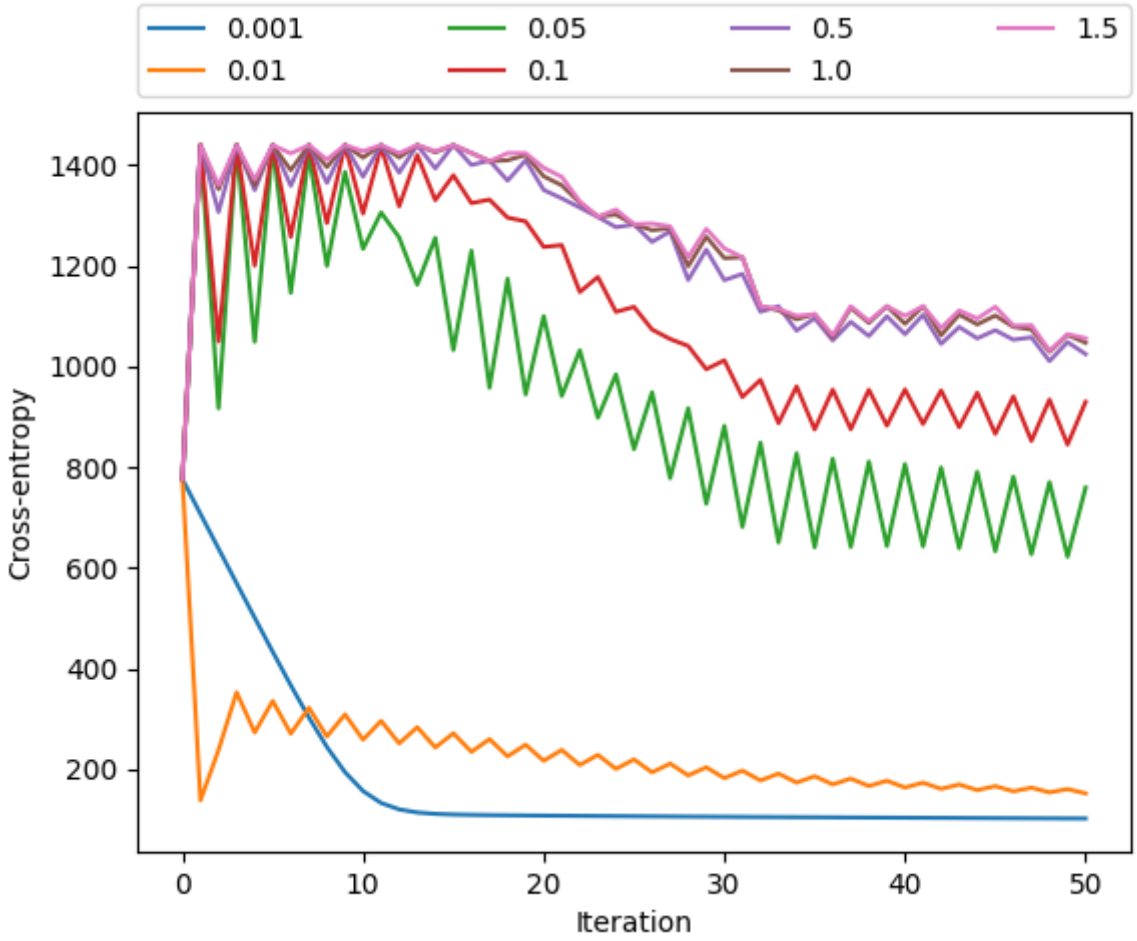
The learning rate was higher than the learning rate in the first experienment. The value of x jumped over to the other valley and found the global minimum.

(e) The learning rate is too high. The value of x bounced around significantly and eventually went overflow.

2.

(a)

i.


0.001
0.01
0.05
0.1
0.5
1.0
1.5
Cross-entropy
1400
1200
1000
800
600
400
200
0
10
20
30
40
50
Iteration

ii & iii.

η	**0.001**	**0.01**	**0.05**	**0.1**	**0.5**	**1.0**	**1.5**
Cross-entropy error (training)	102.695001	152.485400	759.863202	930.220967	1024.835574	1047.440783	1056.344492
Classification error (training)	0.283333	0.361111	0.361111	0.366667	0.366667	0.366667	0.366667
$\|\|w\|\|_2$	2.615655	6.596876	32.759741	65.007503	324.261856	649.271428	973.892726

(b)

i && ii.

λ	**0**	**0.05**	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**
Cross-entropy error (training)	102.695	102.884414	103.072320	103.443651	103.809079	104.168684	104.522549
Classification error (training)	0.283333	0.283333	0.283333	0.283333	0.277778	0.277778	0.277778
Classification error (cross-validation)	0.333333	0.333333	0.333333	0.327778	0.327778	0.327778	0.327778
$\|\|w\|\|_2$	2.615655	2.609749	2.603857	2.592117	2.580435	2.568810	2.557242

(c) As we increase the value of λ, the cross-entropy increases. However, classification error on the training set and cross-validation decrease. The value of $||w||_2$ gradually decreases. This is as expected, because the regularized term will penalize high weight values. Therefore, we see a drop in the $||w||_2$ values. Even though cross-entropy error increases, it actually prevents the classifier from overfitting and helps to generalize better.

3.

The possible problem with this approach is overfitting. If we perform cross-validation on the whole training set for different values of λ, and then choose the λ than yielded the smallest classification error, the λ we choose will fit too closely to the training set.

A better solution is to divide the dataset into three parts: training set, validation set, and test set. We use training set to train several classifiers with different λ values, and use validation set to pick the λ than yielded the smallest classification error. Finally, we compute classification error on the test set.