Lab 5

姓名：游灏溢
班级：F1903302
时间：3/11/2021

Exercise1 Linear Regression

Transform the array into the matrix and use the formula to finish the linear regression(Using matrix can provide the inverse matrix X.T to help calculate).

$$\theta = (X \cdot X^T)^{-1} \cdot X^T \cdot y$$
$$\hat{y} = \theta \cdot X$$

```
X1=np.mat(X)
Y1=np.mat(Y)
theta=(X1.T*X1).I*X1.T*Y1
Y2=X1*theta
```

The linear regression can map the X(in 500-dimension space) into 100-dimension space.

Exercise2 Logistic Regression

We choose

```
learning_rate=[0.5,0.1,0.02]
threshold=[0.2,0.5,0.8]
```

Since it is hard to place all the line in one graph, here I draw 9 graphs, you can refer them in the appendice.

Based on the result, we can have

1. When the learning rate increasing, the decreasing speed of loss and increasing speed of precise will be fast, but when it is to high, both of two will be not stable and fluctuate.

2. Learning speed are all similar, but the stochastic is too instable, especially when the learning rate is high
3. The difference between different threshold is no very distinct.
   According to the result, for decision boundary of predictions, when the threshold is large, the predict will have more on 0 than 1.

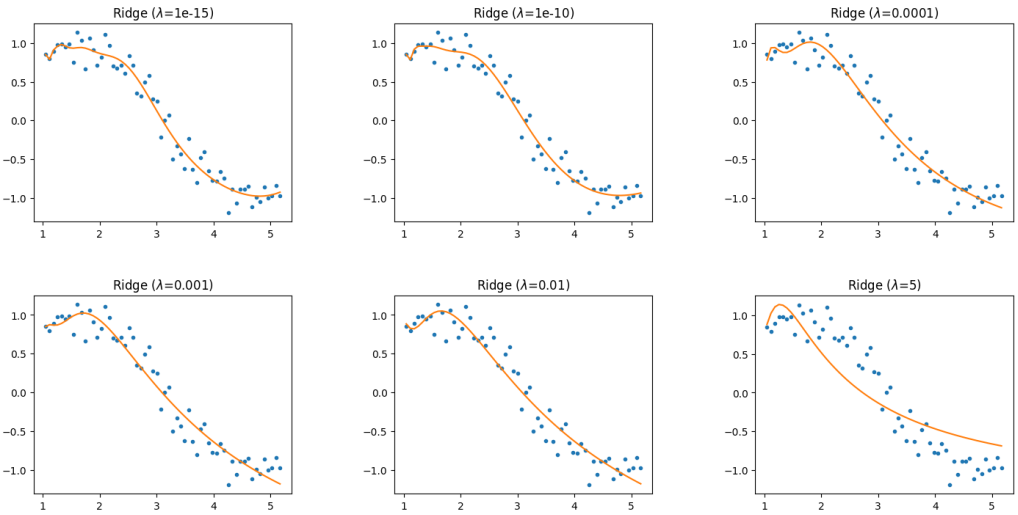
Exercise 3 L1/L2 Regularization

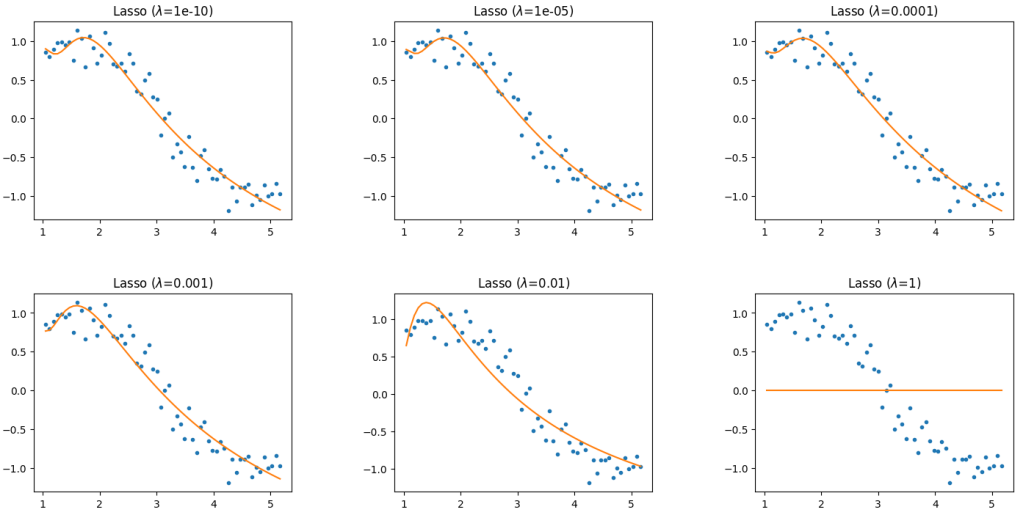
1. I record the time w.r.t 2 methods and compare its distance, Here is

```
Time distance=0.009141
```

   So Ridge cost a little bit more time than Lasso

2. Based on the data, the Lasso create a sparser output.
3. It's a trade-off. More regularization will force to choose larger λ, thus makes more errors between the predict and actual data. When dealing with the complex model, we should take both the regularization and generalizability into consideration, in order to guarantee both precision and avoid overfitting .

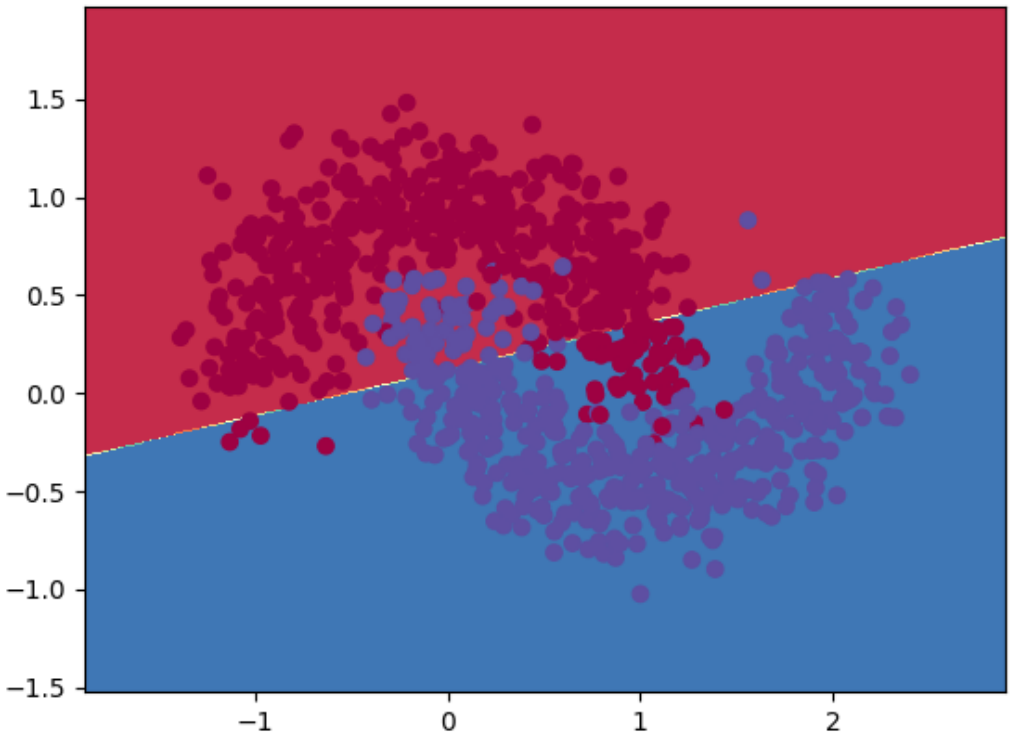




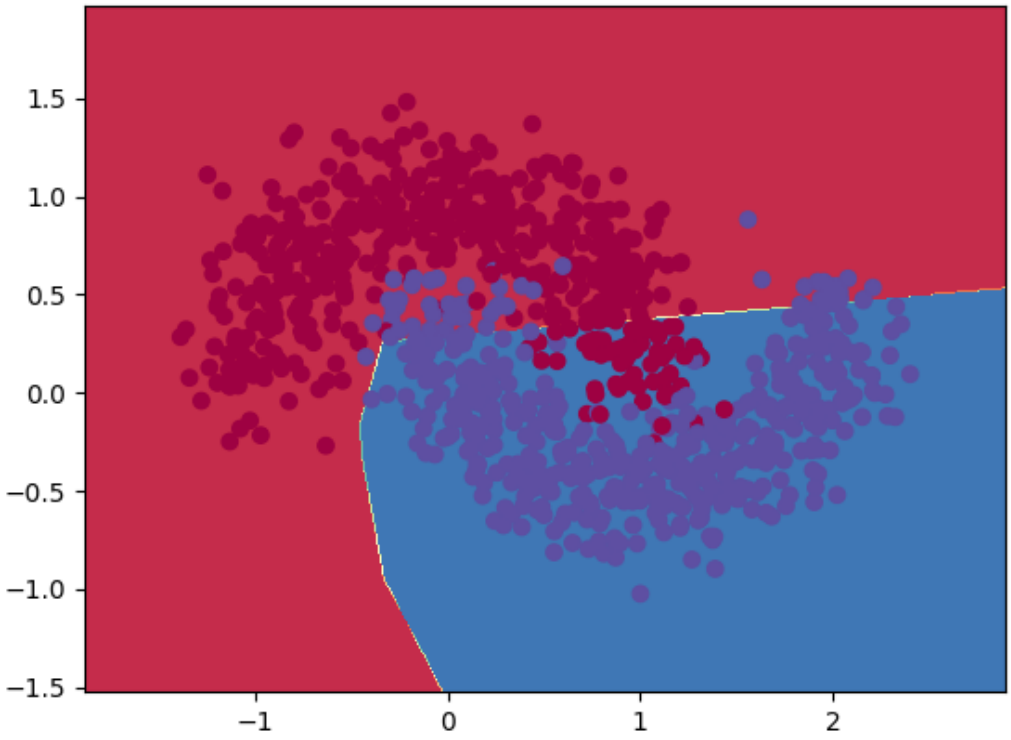
Exercise 4: Two-layer Perceptron Network

Here is the result

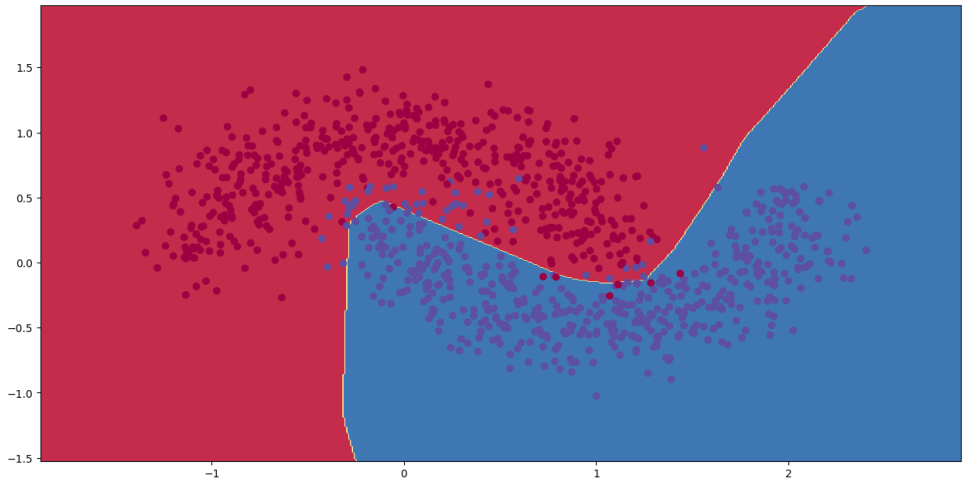
neurons=2



neurons=20



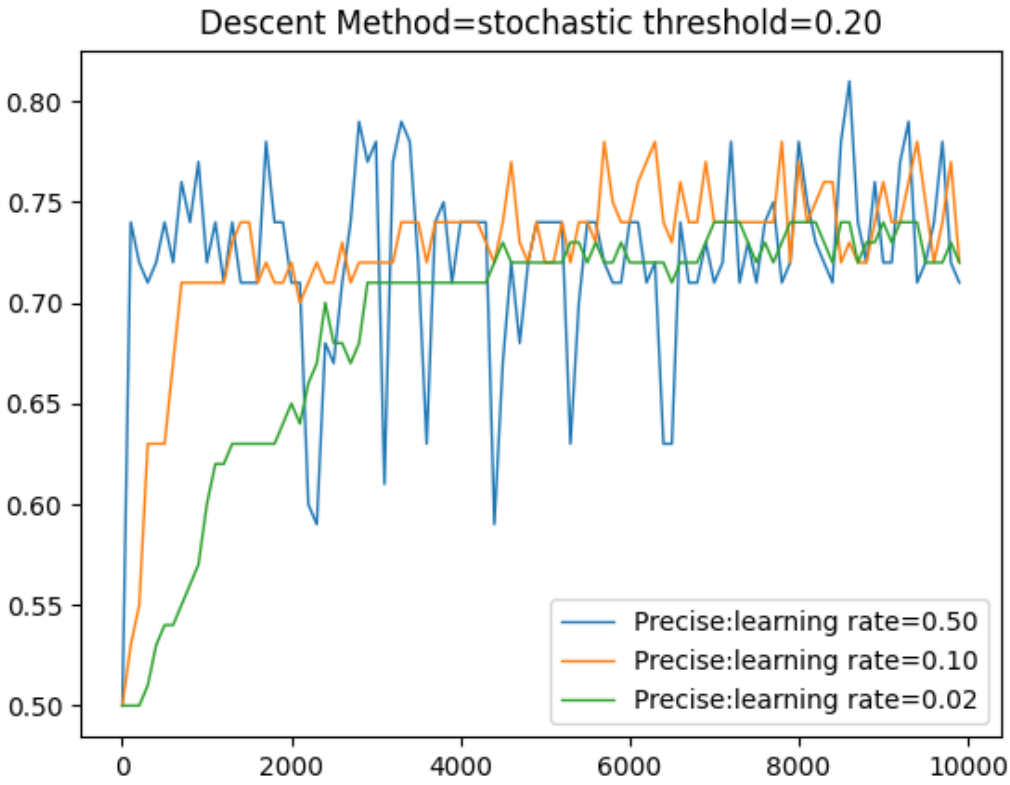
neurons=100

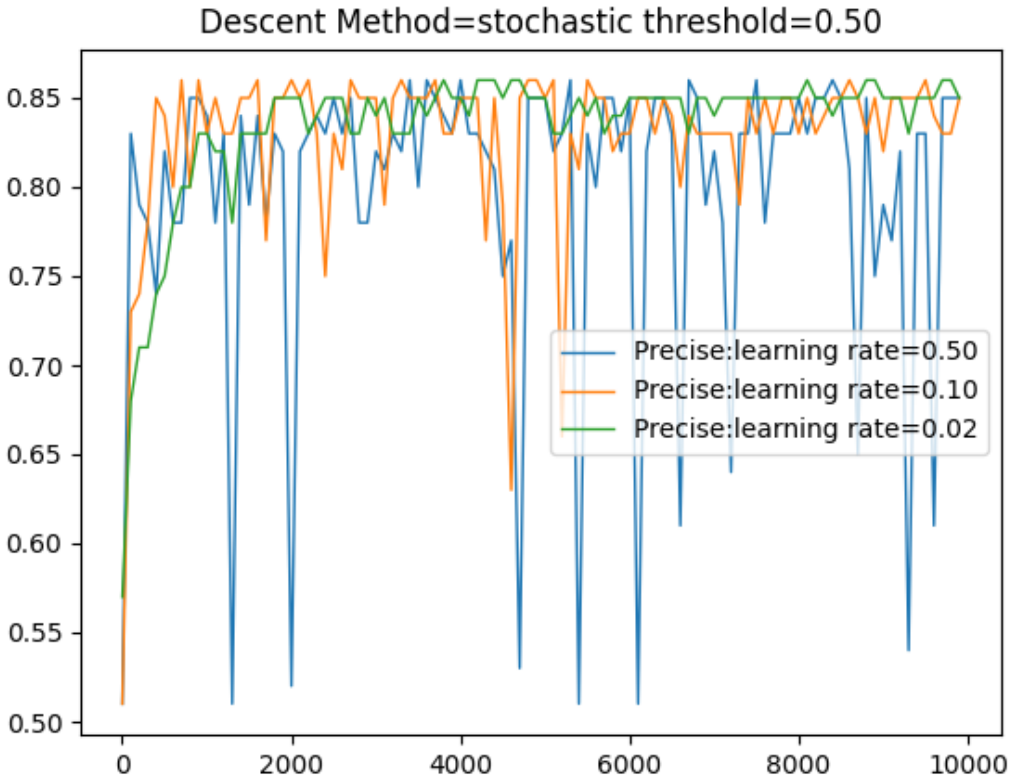


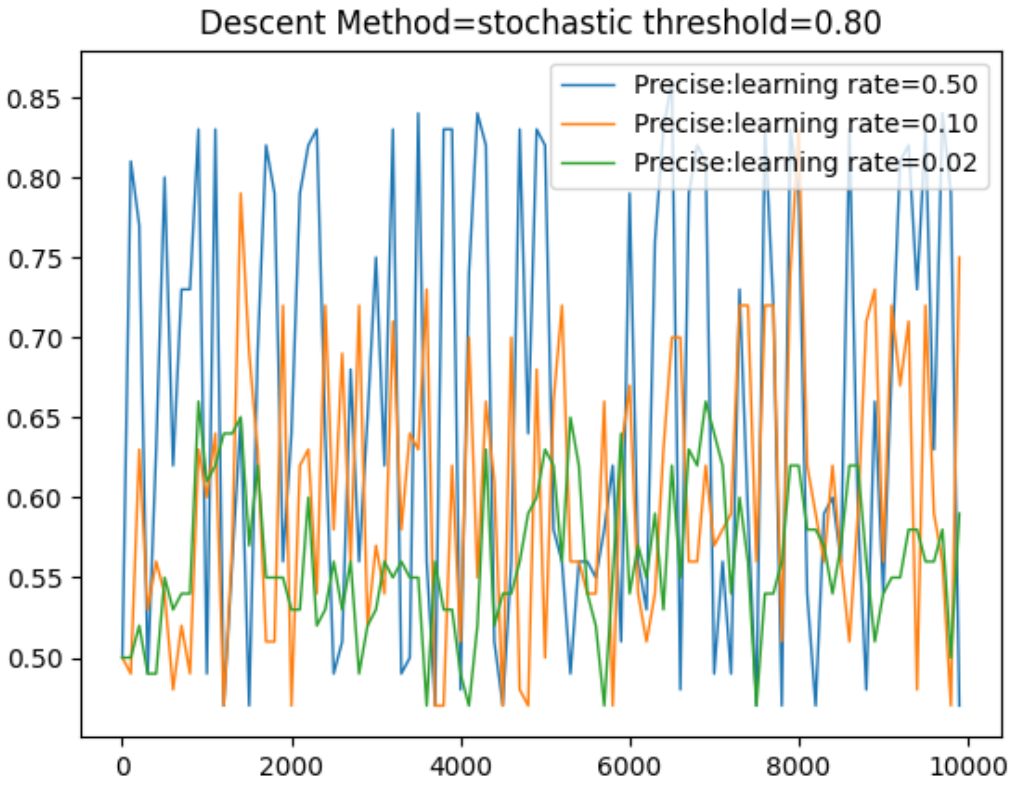
We can find more neurons will make more precise result.

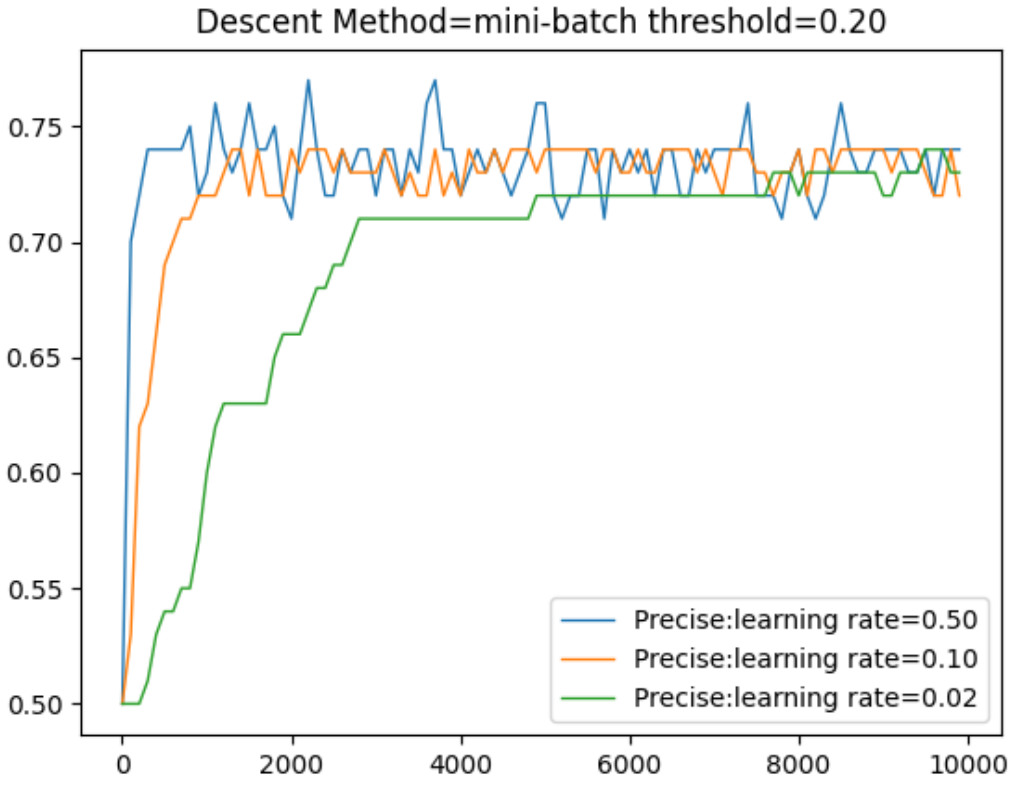
But it also cause more calculations

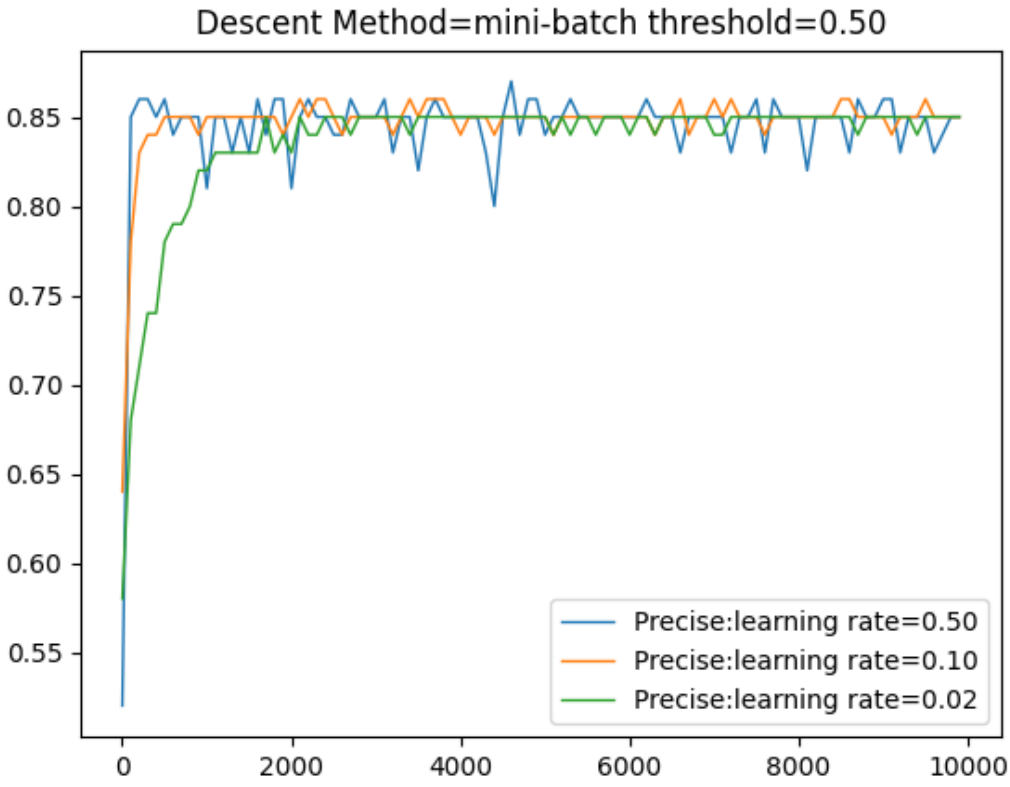
Appendice

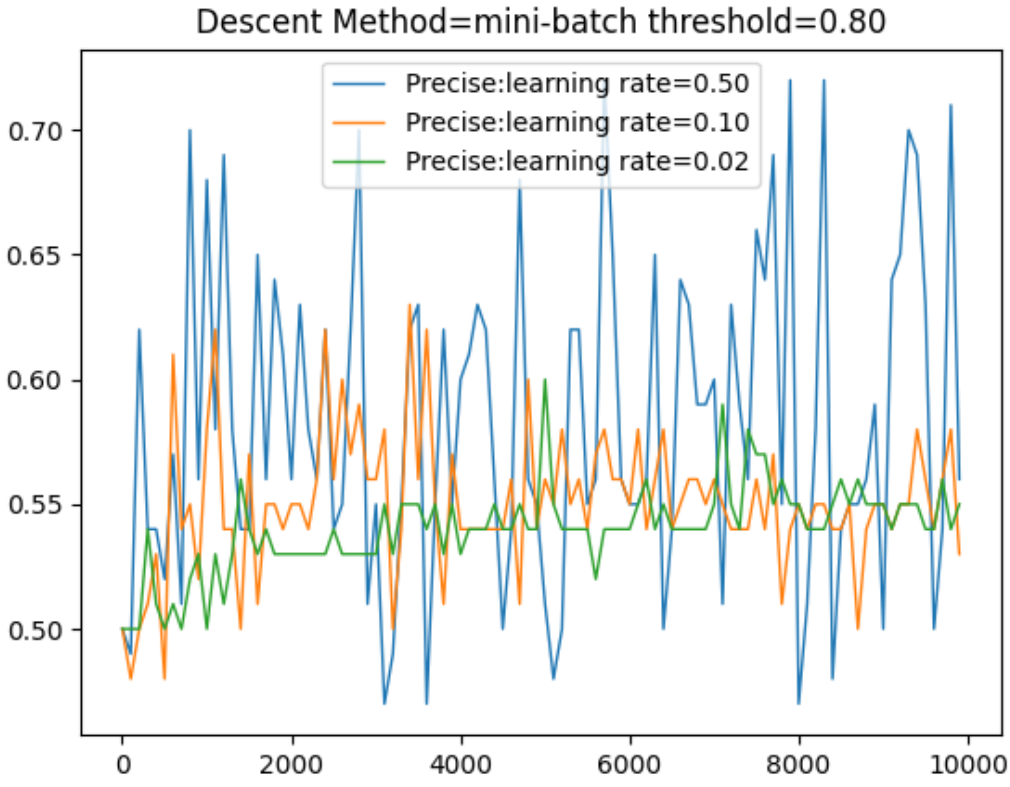

Descent Method=stochastic threshold=0.20
0.80
0.75
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
8000
10000
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02

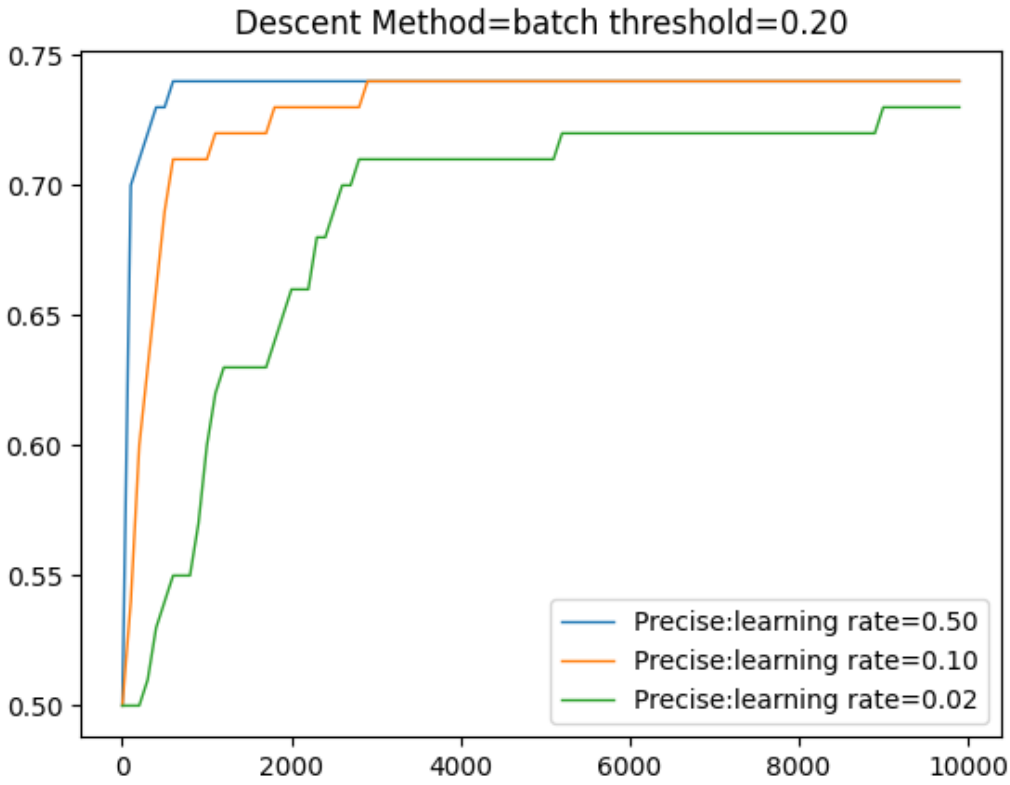

Descent Method=stochastic threshold=0.50
0.85
0.80
0.75
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
8000
10000
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02

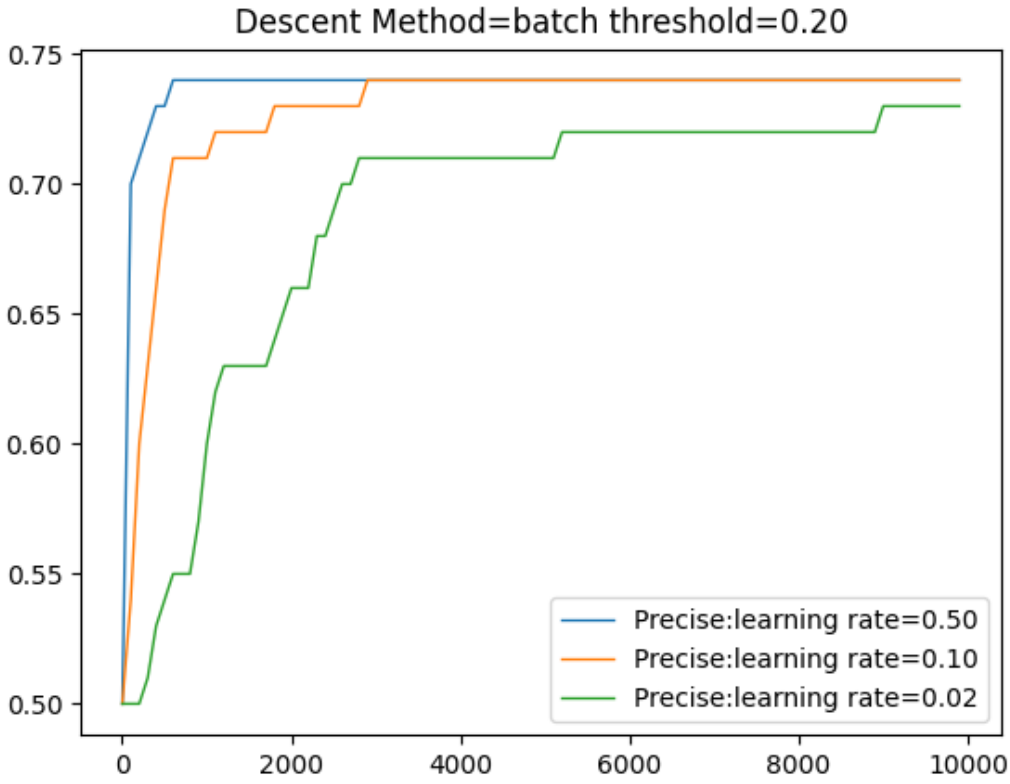

Descent Method=stochastic threshold=0.80
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02
0.85
0.80
0.75
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
8000
10000

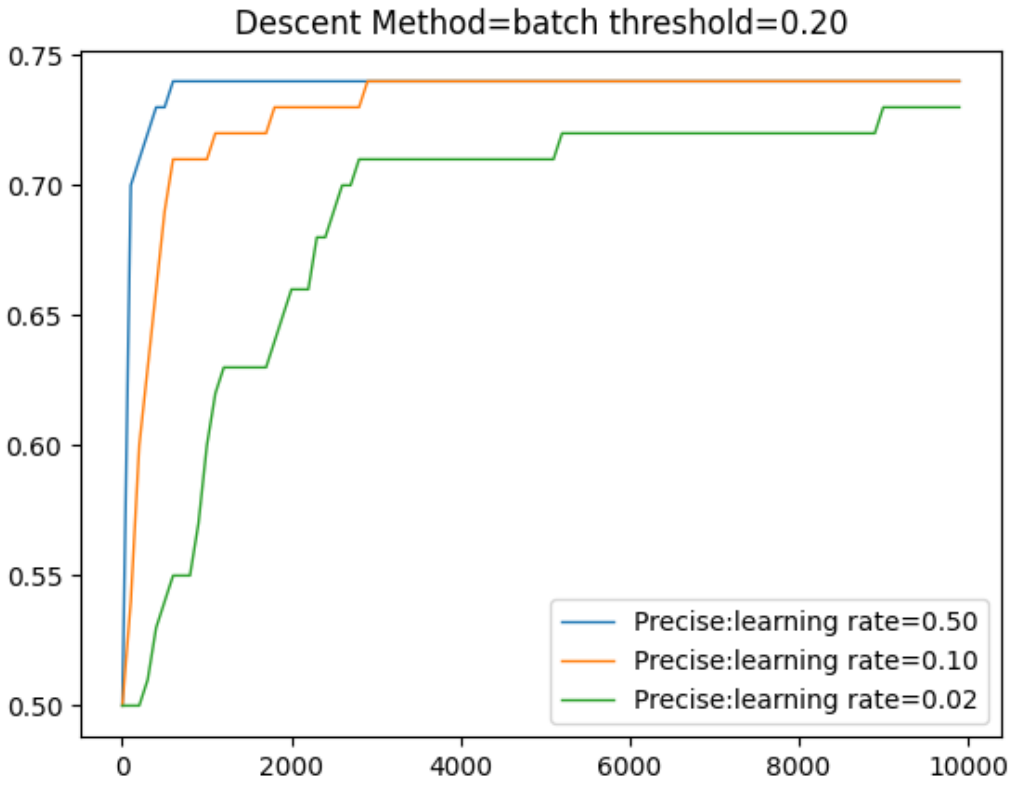

Descent Method=mini-batch threshold=0.20
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02
0.75
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
8000
10000

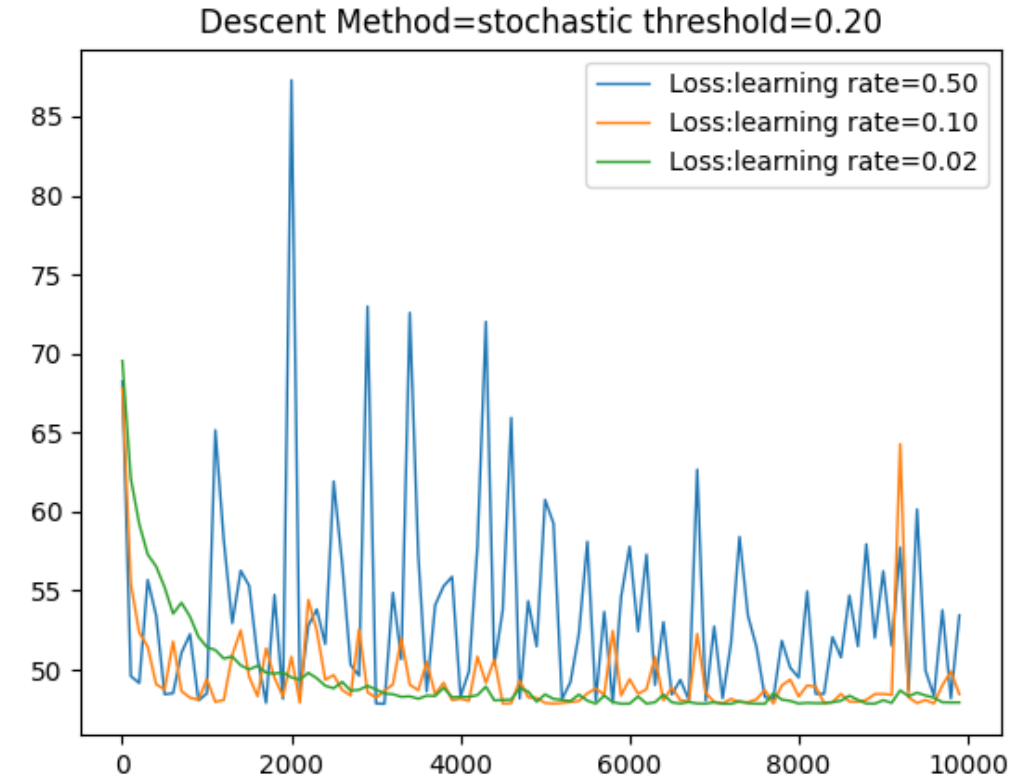

Descent Method=mini-batch threshold=0.50
0.85
0.80
0.75
0.70
0.65
0.60
0.55
0
2000
4000
6000
8000
10000
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02

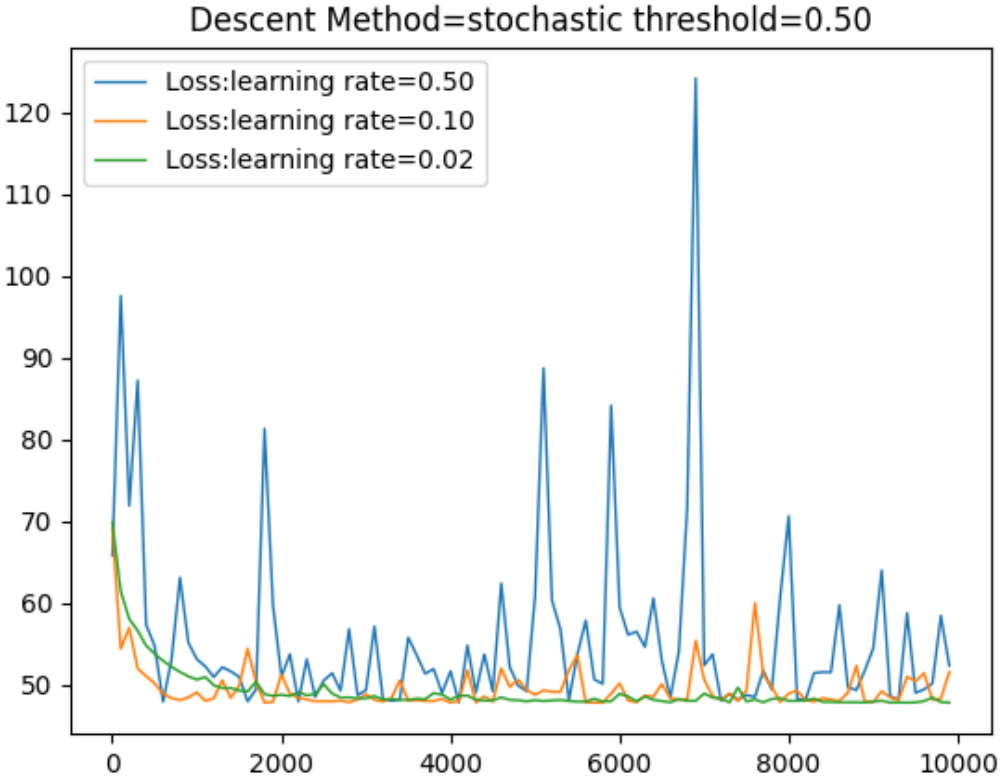

Descent Method=mini-batch threshold=0.80
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
8000
10000
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02

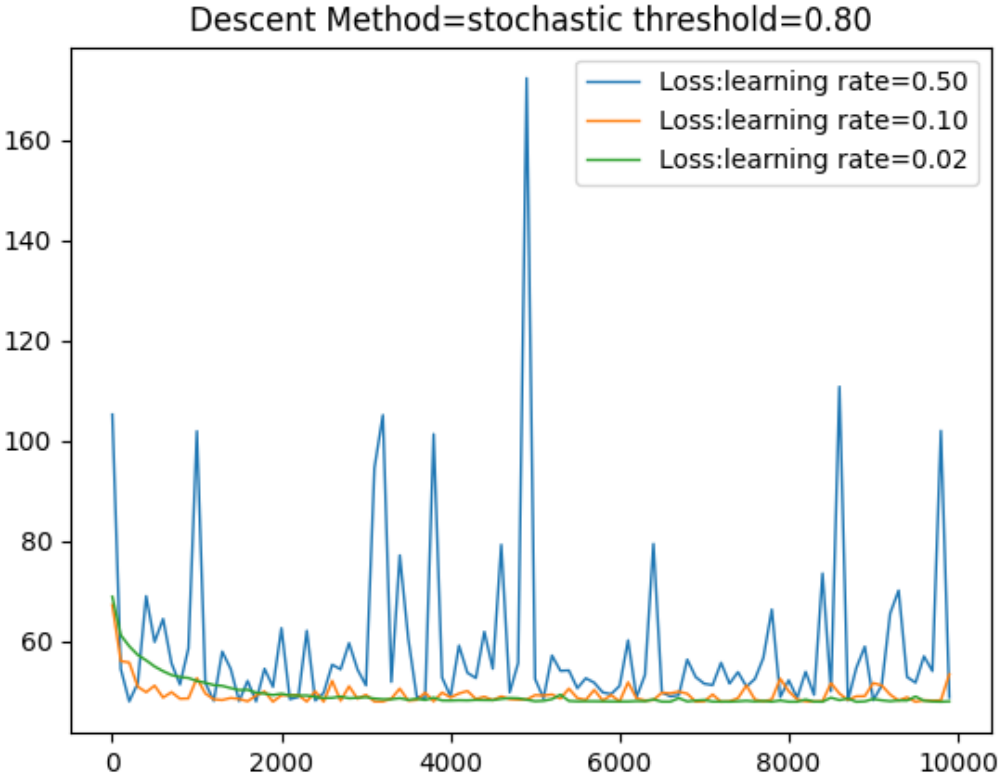

Descent Method=batch threshold=0.20
0.75
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
8000
10000
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02

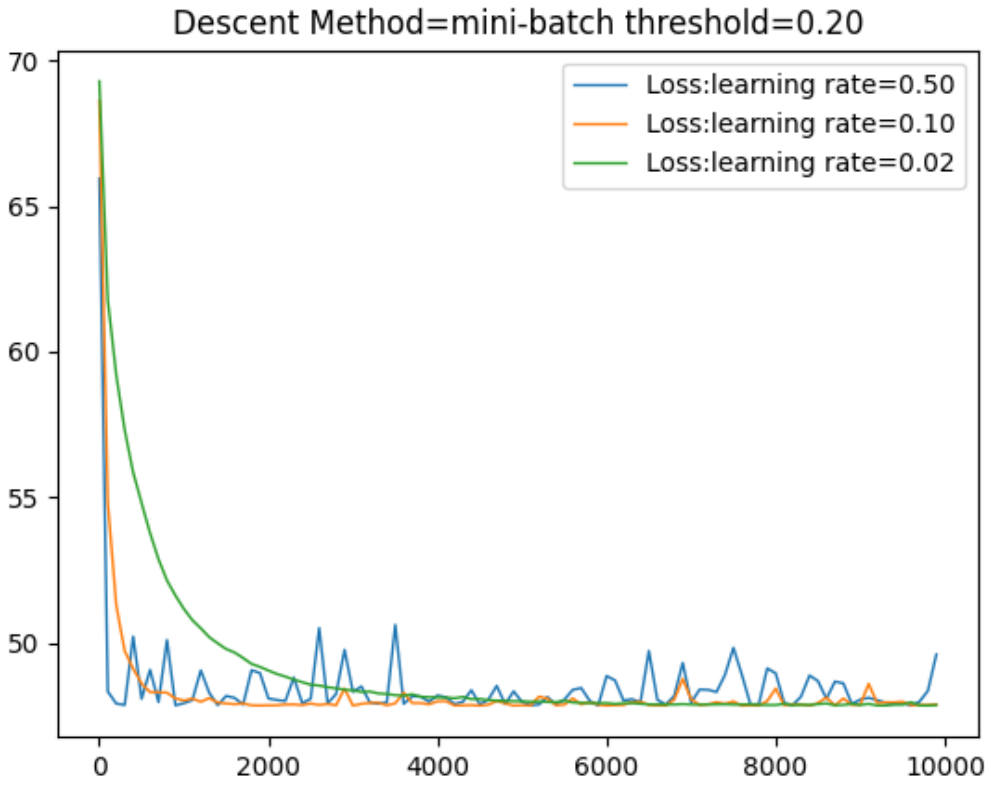

Descent Method=batch threshold=0.20
0.75
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
8000
10000
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02

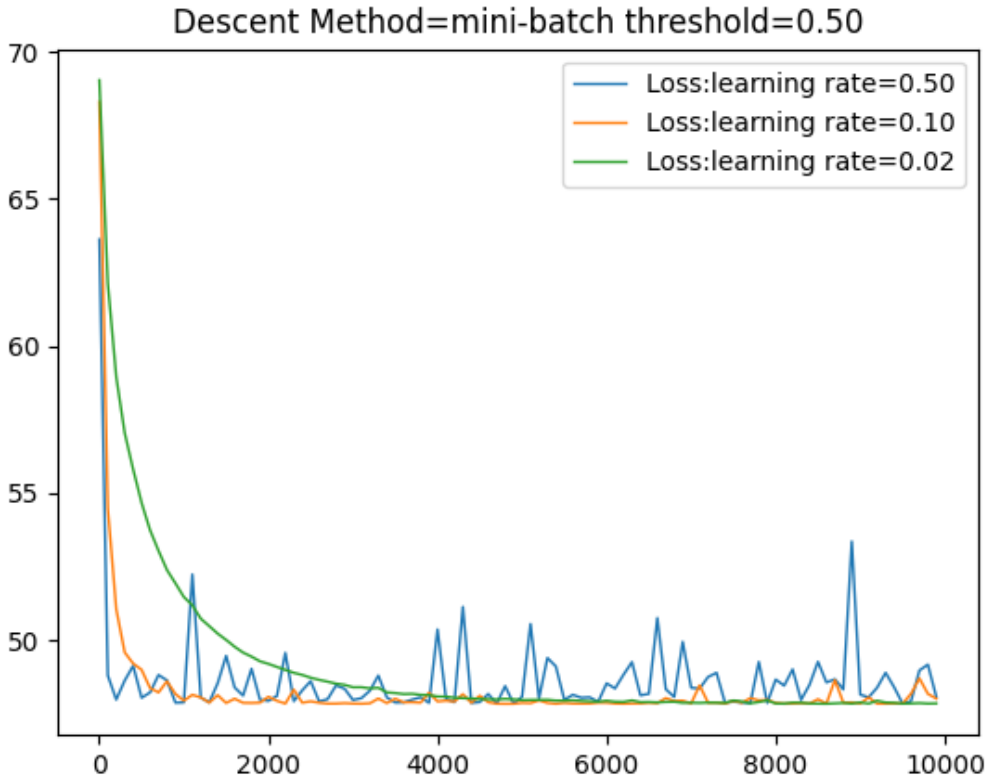

Descent Method=batch threshold=0.20
0.75
0.70
0.65
0.60
0.55
0.50
0
2000
4000
6000
8000
10000
Precise:learning rate=0.50
Precise:learning rate=0.10
Precise:learning rate=0.02

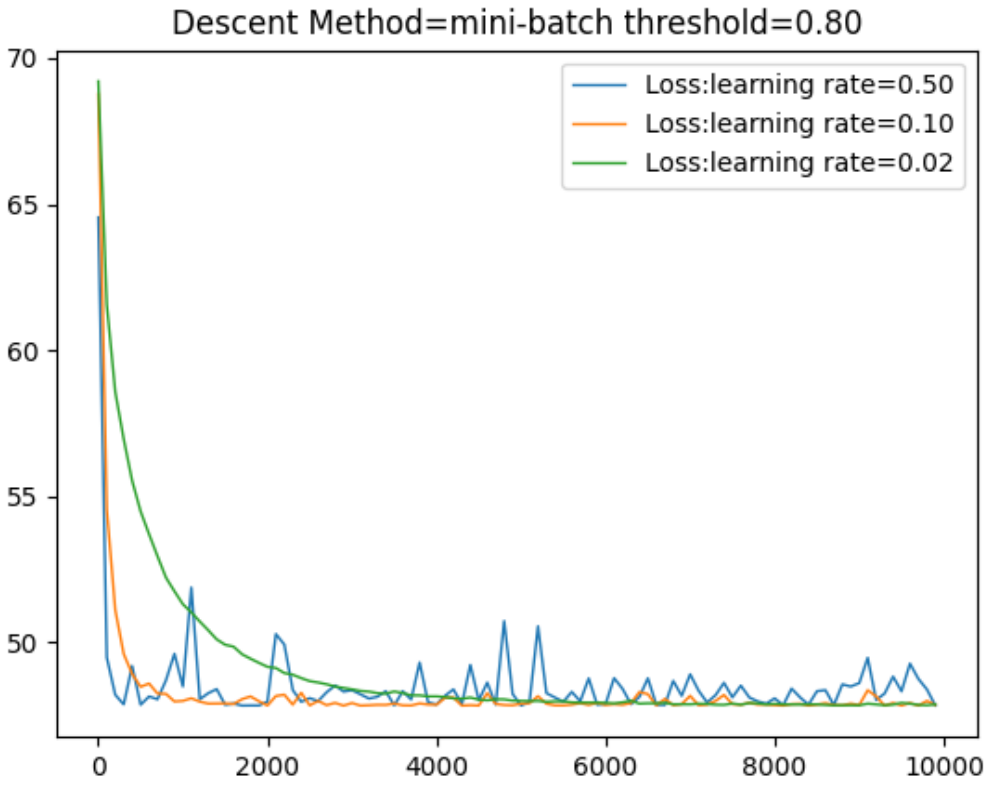

Descent Method=stochastic threshold=0.20
85
80
75
70
65
60
55
50
0
2000
4000
6000
8000
10000
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02

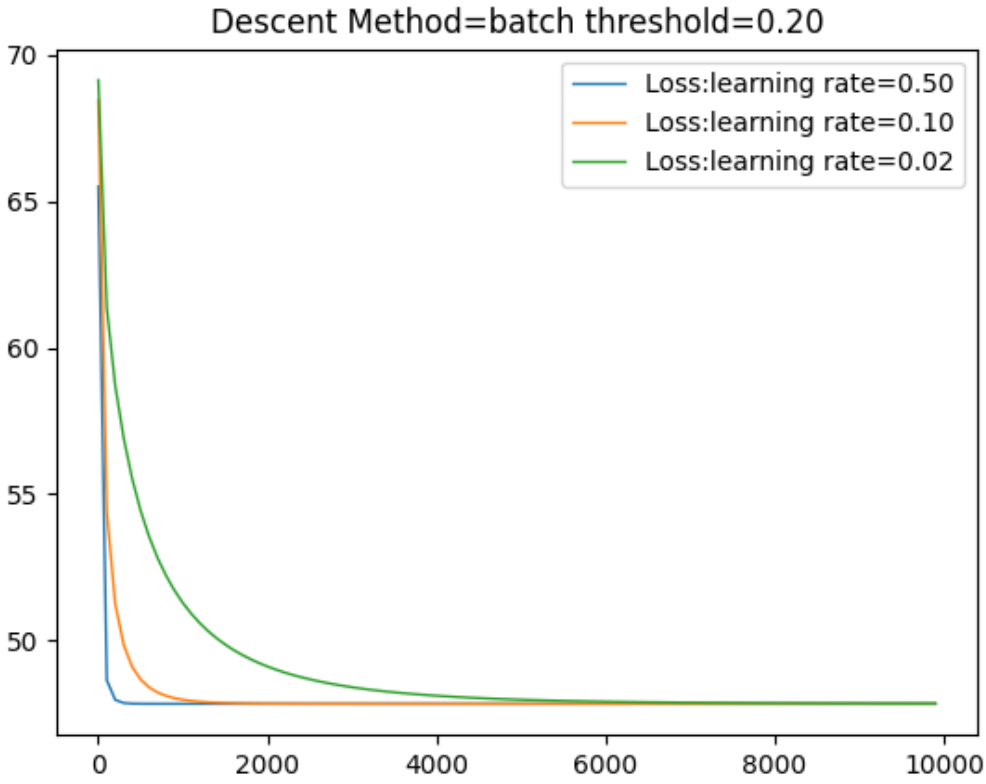

Descent Method=stochastic threshold=0.50
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02
120
110
100
90
80
70
60
50
0
2000
4000
6000
8000
10000

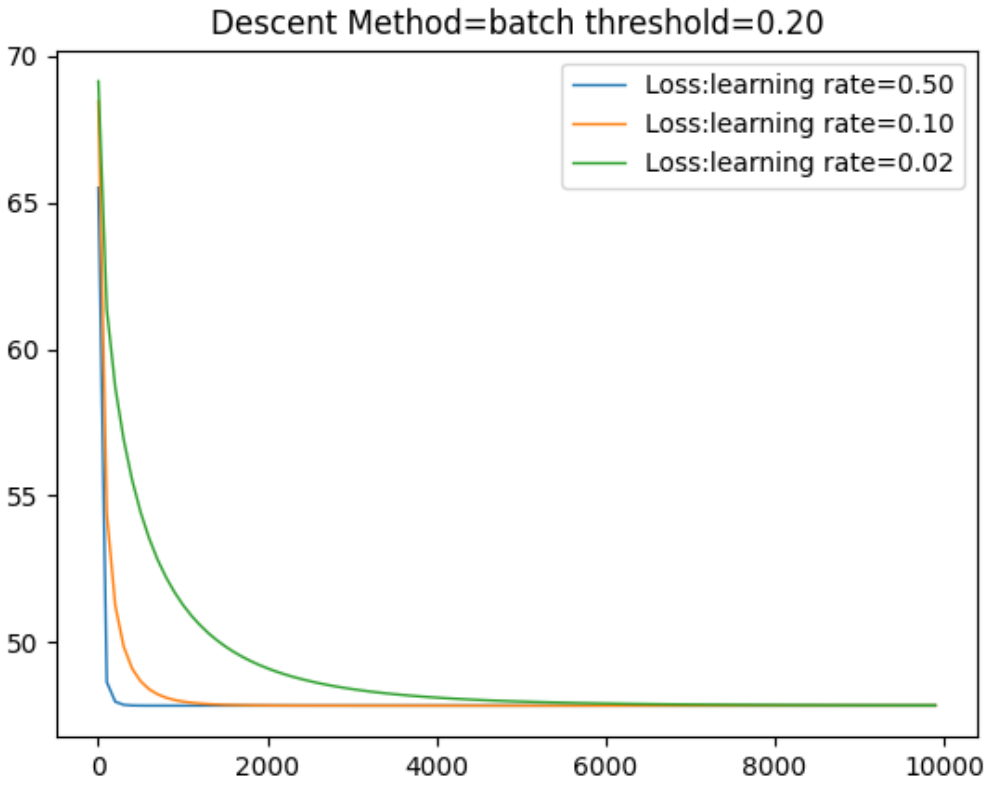

Descent Method=stochastic threshold=0.80
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02
160
140
120
100
80
60
0
2000
4000
6000
8000
10000

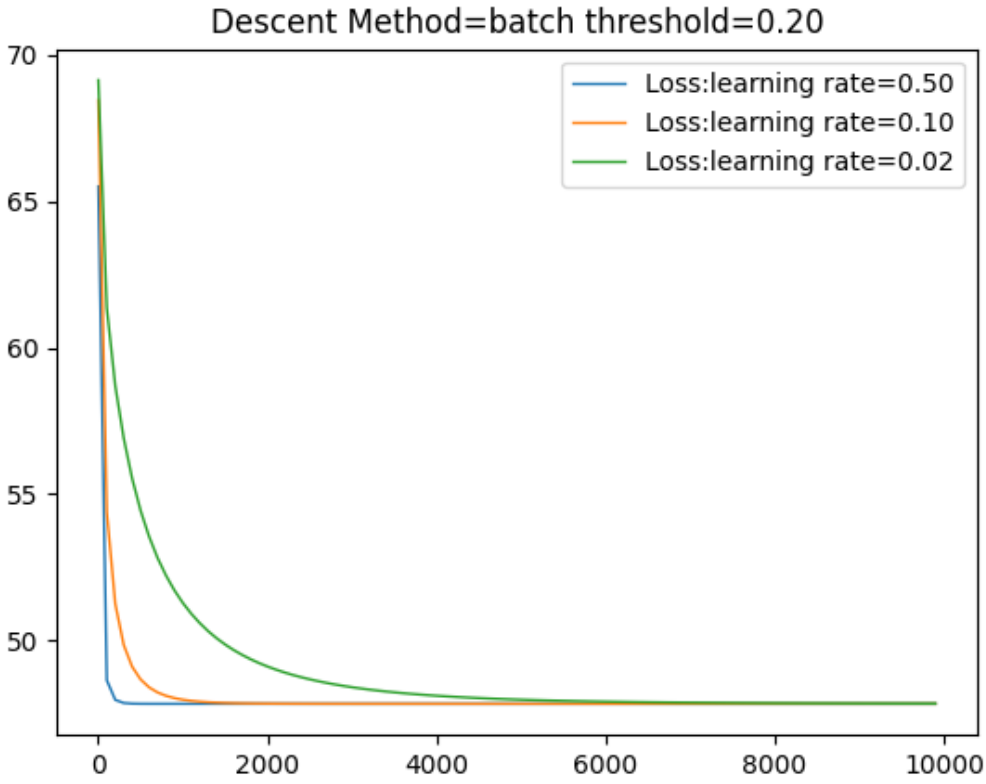

Descent Method=mini-batch threshold=0.20
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02
70
65
60
55
50
0
2000
4000
6000
8000
10000


Descent Method=mini-batch threshold=0.50
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02
70
65
60
55
50
0
2000
4000
6000
8000
10000


Descent Method=mini-batch threshold=0.80
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02
70
65
60
55
50
0
2000
4000
6000
8000
10000


Descent Method=batch threshold=0.20
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02
70
65
60
55
50
0
2000
4000
6000
8000
10000


Descent Method=batch threshold=0.20
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02
70
65
60
55
50
0
2000
4000
6000
8000
10000


Descent Method=batch threshold=0.20
Loss:learning rate=0.50
Loss:learning rate=0.10
Loss:learning rate=0.02
70
65
60
55
50
0
2000
4000
6000
8000
10000