EE559

HW5

1.(a)

```
The error rate is 0.03
[-57.23451  47.00663  27.1     ]
The testing1 error rate is 0.0
```

Synthetic1 training error rate is 0.03, testing error rate is 0.0, the final weight vector is [-57.23451 47.00663 27.1].

```
The error rate is 0.01
[-1.97798 16.94005  5.1     ]
The testing2 error rate is 0.03
```

Synthetic2 training error rate is 0.01, testing error rate is 0.03, the final weight vector is [-1.97798 16.94005 5.1].

```
The error rate is 0.0
[-13.19333  10.7326    5.1     ]
The testing3 error rate is 0.0
```

Synthetic3 training error rate is 0.0, testing error rate is 0.0, the final weight vector is [-13.19333 10.7326 5.1].

(b)

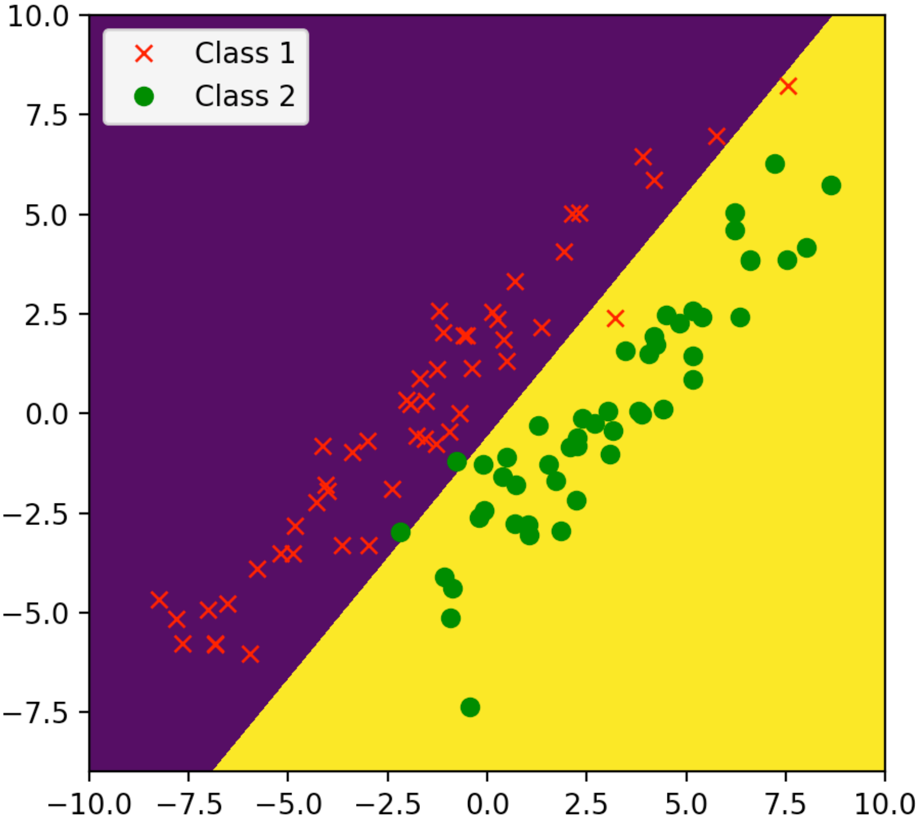

Fig1. Training plot of Synthetic1

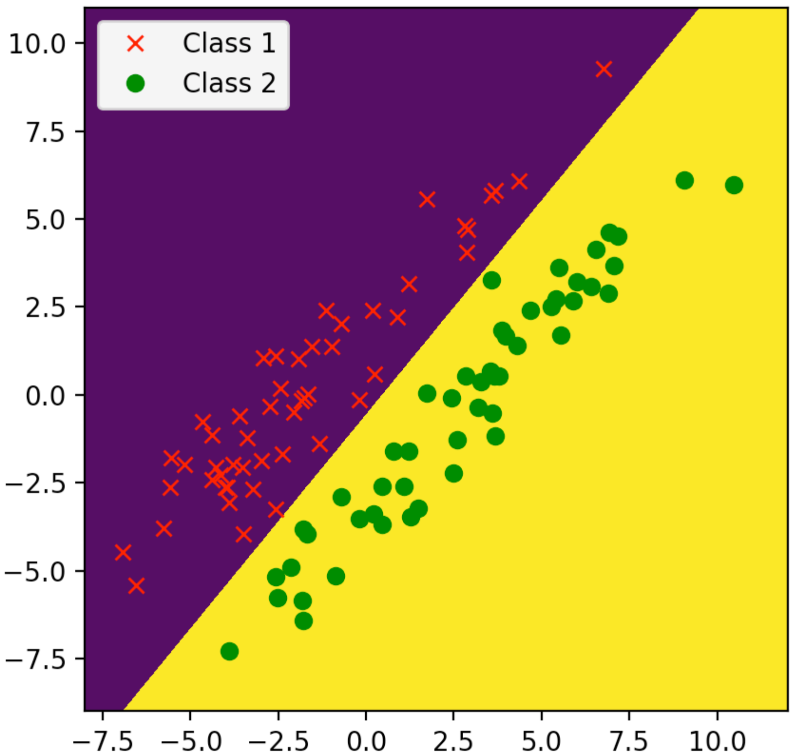

Fig2. Testing plot of Synthetic1

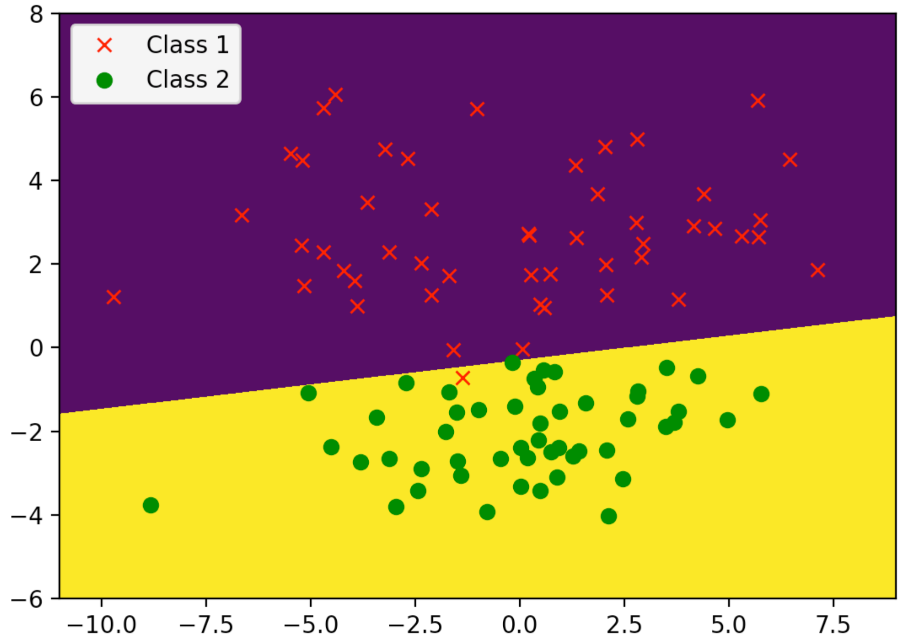

Fig3. Training plot of Synthetic2

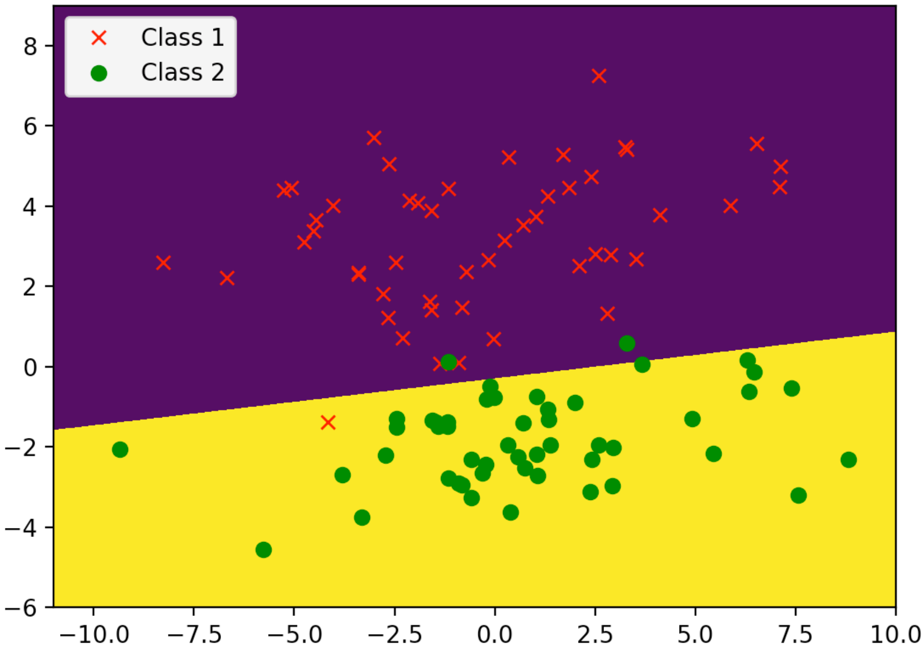

Fig4. Testing plot of Synthetic2

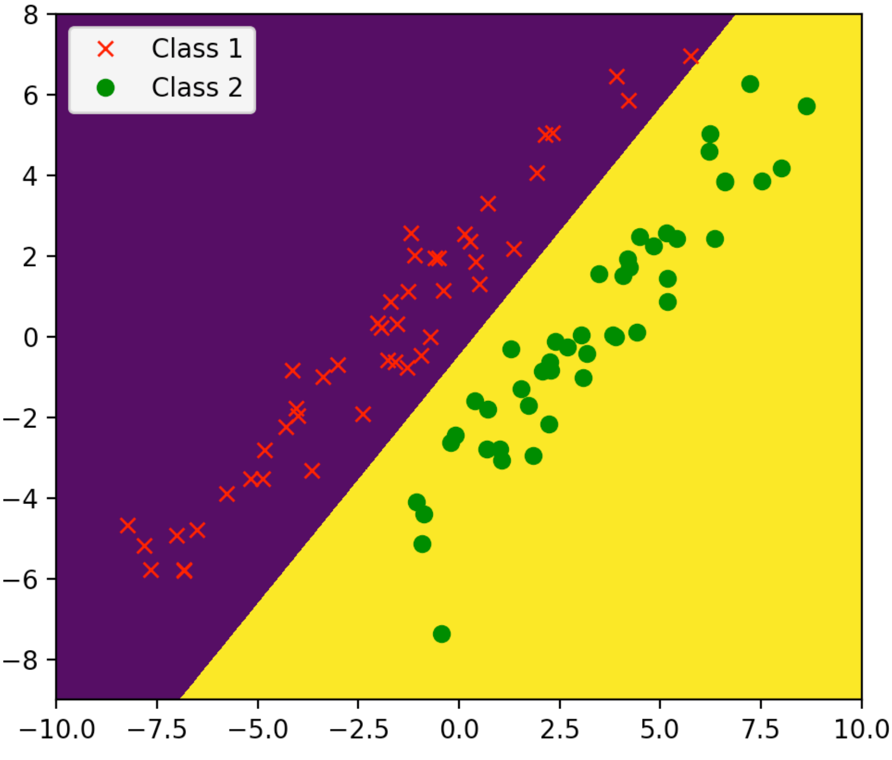

Fig5. Training plot of Synthetic3

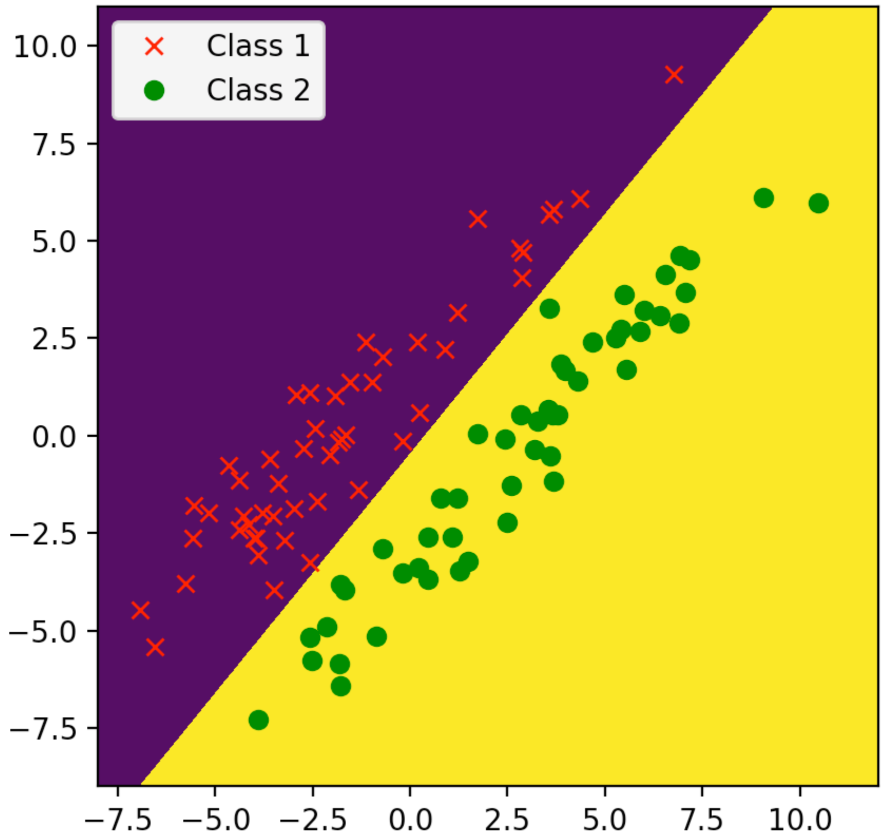

Fig6. Testing plot of Synthetic3

(c)
The Synthetic1 training error rate in HW1 is 0.21, testing error rate is 0.24
The Synthetic1 training error rate is 0.03, testing error rate is 0.0
The Synthetic1 training error rate in HW1 is 0.03, testing error rate is 0.04
The Synthetic1 training error rate is 0.01, testing error rate is 0.03
The result of Synthetic1 by perceptron classifier is much less than the result of nearest means by perceptron classifier. Because the data points in Synthetic1 are scatter of each class but clearly separable for this 2 class.
The result of Synthetic2 by perceptron classifier is similar to the result of nearest means by perceptron classifier. Because the data points in Synthetic2 are close of each class and clearly separable for this 2 class.

Because the decision boundary of nearest means classifier depends on the perpendicular bisector of the sample mean, which means it depends on the sample means of two classes, there might be some of data points which are far away from its sample mean and are incompatible with the decision boundary. In sum, nearest means classifier doesn't fit for the data points which are very scatter of each class. It fits for the data points which are very close to each other of each class. It fit the small number of data points. This algorithm spends less time (just depends on the sample mean)
And the decision boundary of perceptron classifier depends on each data points, which means the parameter w is fixed by each data point. So it can show a small error rate compared with the nearest means classifier. However this algorithm spends much time

(depends on each data points may for many epochs). It fits for the data points which are clearly separate with each other of each class.

2.

2.

(a). For stochastic gradient descent, variant 2.

$\underline{w}(i+1) = \underline{w}(i) - \eta \nabla_{\underline{w}} J_n(\underline{w}) \quad \eta(i) > 0.$ for single-sample update.

$\Delta \underline{w}(i) = \underline{w}(i+1) - \underline{w}(i).$

$= -\eta \nabla_{\underline{w}} J_n(\underline{w})$

$E\{\Delta \underline{w}(i)\} = \sum_{n=1}^{N} -\eta \nabla_{\underline{w}} J_n(\underline{w}) \frac{1}{N-1+1}$

$\because$ In q2 said $J(\underline{w}) = \sum_{n=1}^{N} J_n(\underline{w}) \Rightarrow E\{\Delta \underline{w}(i)\} = \frac{-\eta \nabla_{\underline{w}} J(\underline{w})}{N}. = \frac{-\eta \nabla_{\underline{w}} J(\underline{w})}{N}$

(b) $\Delta \underline{w}(i)$ for batch gradient descent.

$w(i+1) = w(i) - \eta \nabla_{\underline{w}} J(\underline{w})$

$E\{\Delta \underline{w}(i)\} = E\{w(i+1) - w(i)\} = -\eta \nabla_{\underline{w}} J(\underline{w})$

$\Rightarrow$ The expected value of bath update GD is N times of the expected value of stochastic GD variant 2 update.
which means SGD speeds much less time compared with bath updated GD
But SGD's path to the minimal ~~many~~ noiser than batch update GD because it's just random choose data pt with replacement.