Spring 2021
CMPE 462: Machine Learning
Assignment 1 Report
Sertay Akpinar - 2016400075
Part 1:

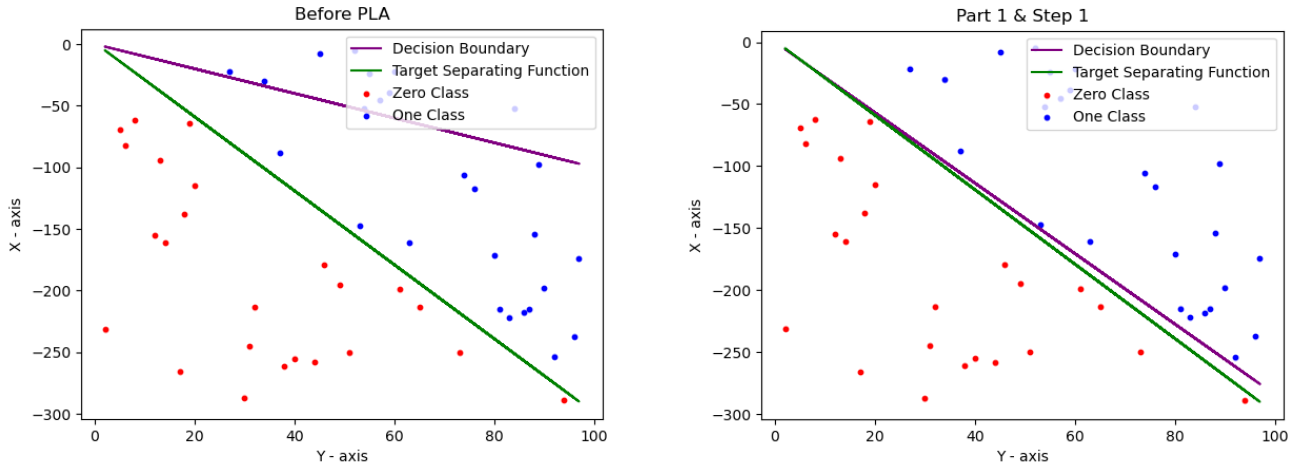

Figure a) 50 data points

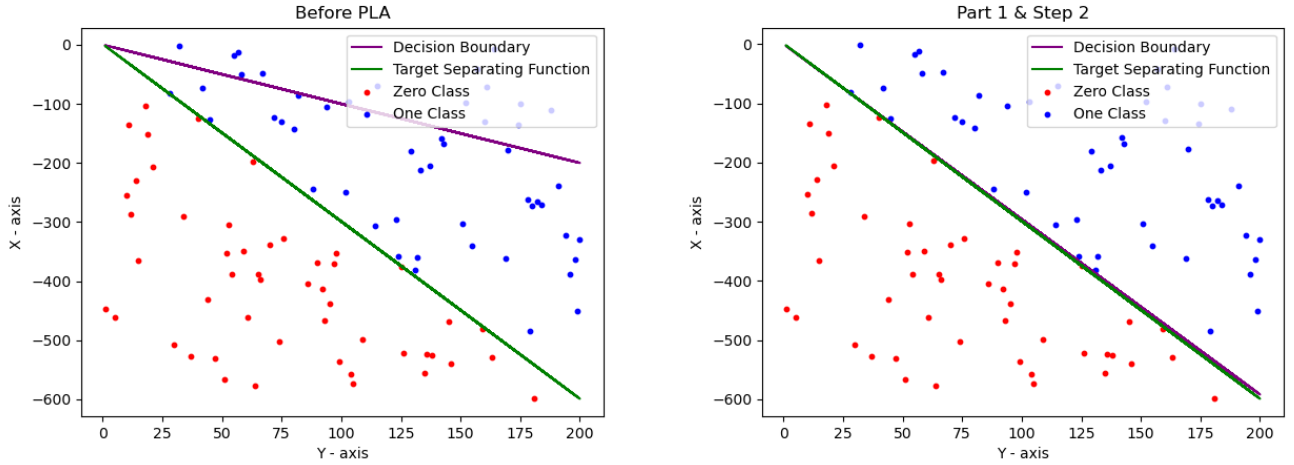

Figure b) 100 data points

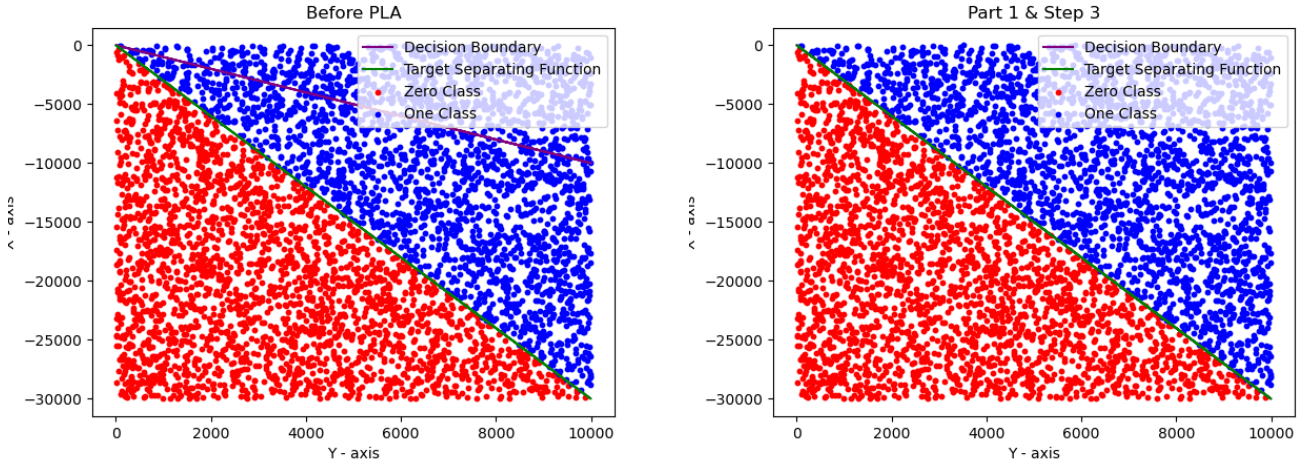

Figure c) 5000 data points

My initial boundary was $y = -x$. Resulting boundaries are getting close to the target separating function as data points increase. The decision boundary can not be seen clearly when there is more data on the graph. That's because the pla algorithm needs more iteration to end up with a good resulting boundary which is close to the target separating function. As we can see from the pictures, from figure a to c the resulting boundaries are getting close to the target function such that at figure c, we can't even differentiate between the target separating function and the resulting boundary.

```
(base) apple@Sertays-MacBook-Air Assignment1 % python3 assignment1.py part1 step1
Total number of iteration needed for step 1: 6
(base) apple@Sertays-MacBook-Air Assignment1 % python3 assignment1.py part1 step2
Total number of iteration needed for step 2: 11
(base) apple@Sertays-MacBook-Air Assignment1 % python3 assignment1.py part1 step3
Total number of iteration needed for step 3: 694
```

Figure d) Number of iterations

As we can see from the figure d, the number of iterations are increased since the total generated data points are increased.

Part 2:

```
(base) apple@Sertays-MacBook-Air Assignment1 % python3 assignment1.py part2 step1
Time to complete step 1: 35 msec
(base) apple@Sertays-MacBook-Air Assignment1 % python3 assignment1.py part2 step2
Time to complete step 2: 174 msec
(base) apple@Sertays-MacBook-Air Assignment1 % python3 assignment1.py part2 step3
Time to complete step 3: 178 msec
```

Figure d) Time spent on each step

I applied the closed form solution in part 2. Since ds2.csv is bigger than ds1.csv, the run time to complete multiple linear regression is way different. In step 1, ds1.csv is used, however in step2 and step3 ds2.csv is used.

Figure e) Size of the datasets

When we compare the sizes, ds2.csv is nearly 5 times bigger than the ds1.csv. So the run time values are consistent with the data set sizes $-> (35x5 \approx 174)$

Cross Validation:

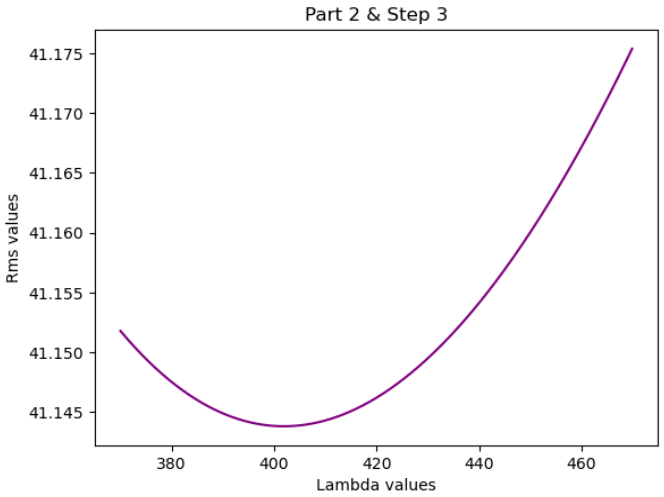

Figure f) Lambda vs. Rms Error

I applied cross validation method to find the reasonable λvalue. First, I shuffled the data set, then I split the dataset into 5 pieces. Training set includes 4 pieces and the test set includes 1 piece. After every run(5 run total, because test and training data sets are changing in each run), I found a 5 RMSe value of the corresponding λvalue. I took the average of this 5 RMSe value and the result was paired to the related λvalue. My λ range was: [370.0, 470.0] and I incremented λvalue by 0.1 in each iteration.

In the end, I plot a graph to find the most optimal λvalue which has the minimum root mean square error. I printed the minimum extremum point of the function in the graph.

```
INFO:__main__:Applying Cross-Validation step please wait...
100%|██████████| 1000/1000 [07:22<00:00,  2.26it/s]
RMSe value: 41.14380688174728
Lambda value: 401.9
Time to complete step 3: 443288 msec
```

Figure g) Optimum Lambda value and corresponding RMSe value

Important Note: Since the cross validation step takes 5-7 min to complete I comment out the related part of the code, to pass this step. Instead, I gave the hardcoded lambda value which is 401.9 to the calc_weight() function. **If you want to apply cross validation, the code has to be like in the figure h instead of figure i**. I submitted the code in figure i to save your time.

```
234        # calc_weight("ds2.csv", 401.9)  # comment this line if you want to apply cross validation
235        apply_crossval("ds2.csv")  # delete comment and apply cross-valid. to find the lambda with the min RMSe value
```

Figure h)

```
234        calc_weight("ds2.csv", 401.9)  # comment this line if you want to apply cross validation
235        # apply_crossval("ds2.csv")  # delete comment and apply cross-valid. to find the lambda with the min RMSe value
```

Figure i)