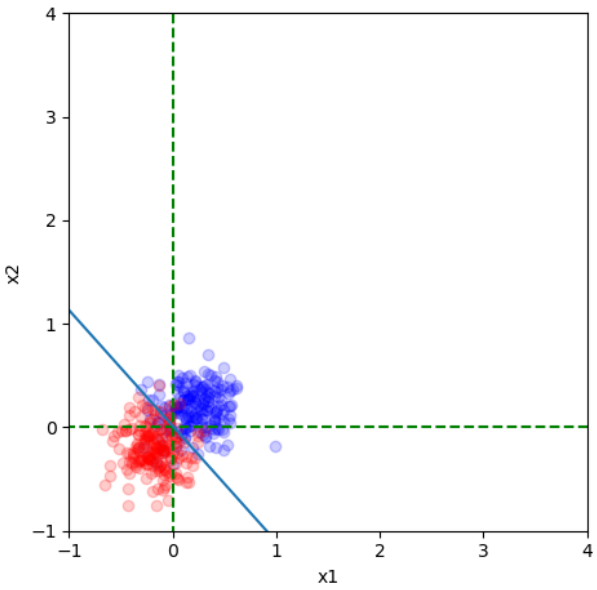
CMPUT 466 Coding Assignment 2

Arun Woosaree

December 7, 2021

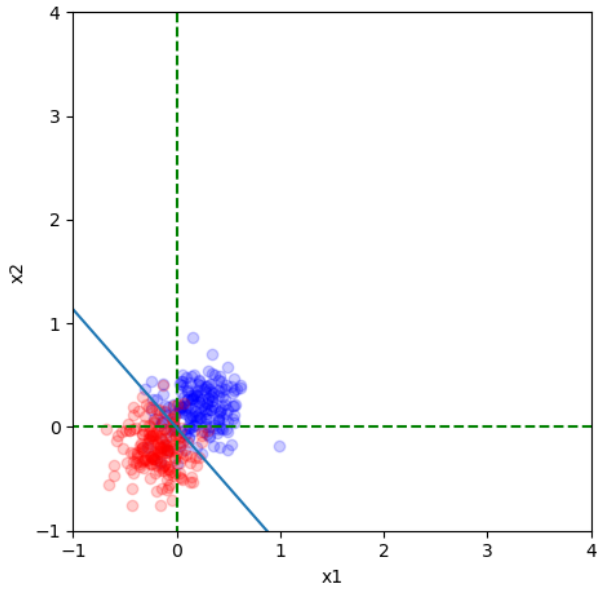
Problem 1

1 Training Accuracy of linear regression on Dataset A



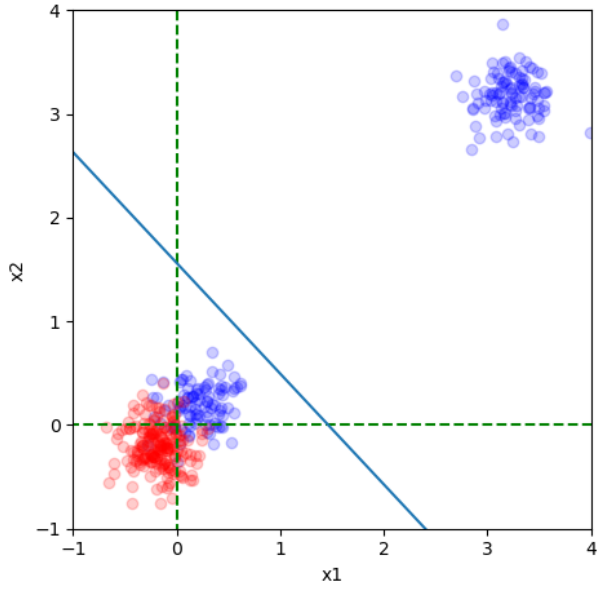
Training accuracy: 0.6025

2 Training Accuracy of logistic regression on Dataset B



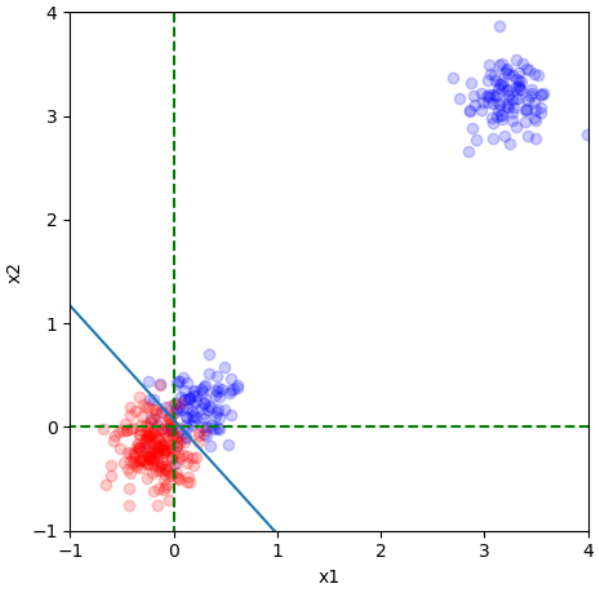
Training accuracy: 0.92

3 Training Accuracy of linear regression on Dataset A



Training accuracy: 0.5

4 Training Accuracy of logistic regression on Dataset B

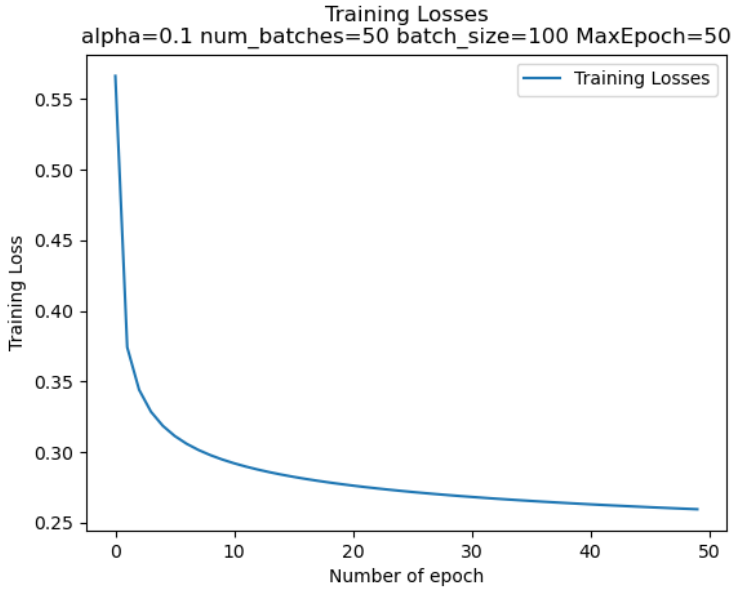


Training accuracy: 0.9375

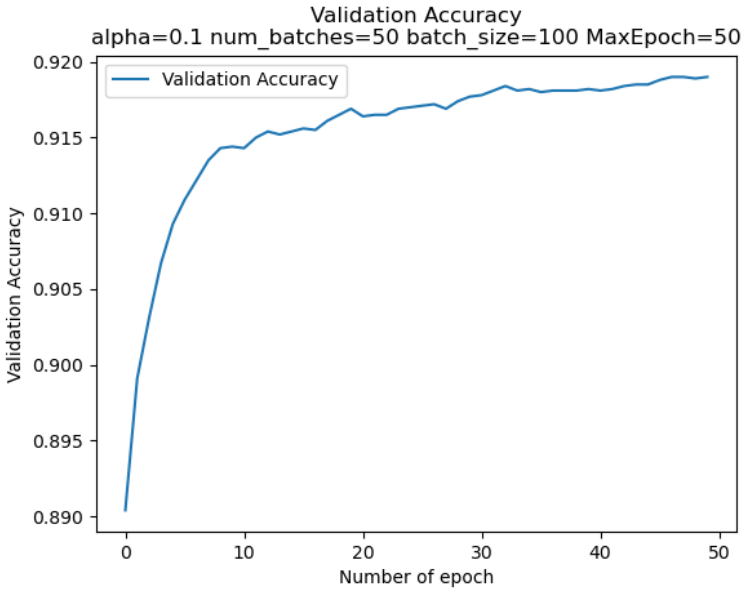
Problem 2

- Best epoch: 49
- Validation accuracy: 0.919
- Test accuracy: 0.9233

1 Loss curve



2 Validation accuracy curve



Scientific Question

Will annealing the step size improve performance?

Hypothesis

Annealing the step size will likely improve the performance of the classifier. However, if the annealing is done too quickly, the performance will likely be worse.

Experiment

We already tried running the experiement without annealing the step size (see above).

We will now try annealing the step size by reducing it a certain amount in each iteration, and another run with annealing the step size too fast.

Results

The following annealing schedulers were tried:

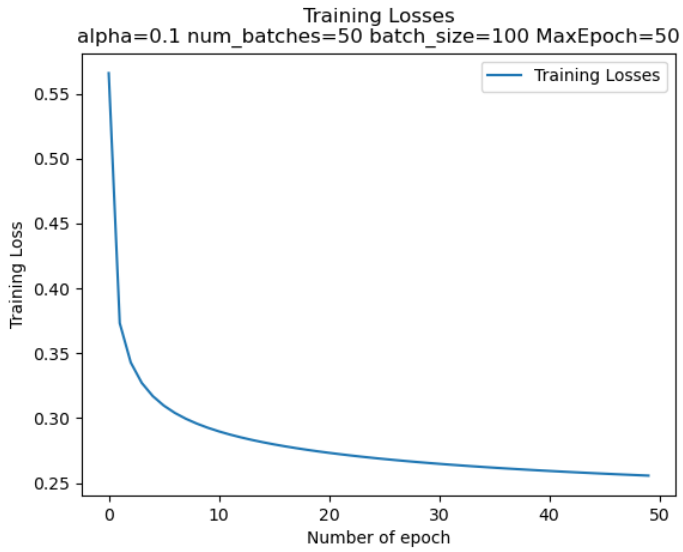
- dividing the step size by 2 each iteration
  - The step size was reduced too quickly Also, doing this resulted in overflow so no useful results were produced

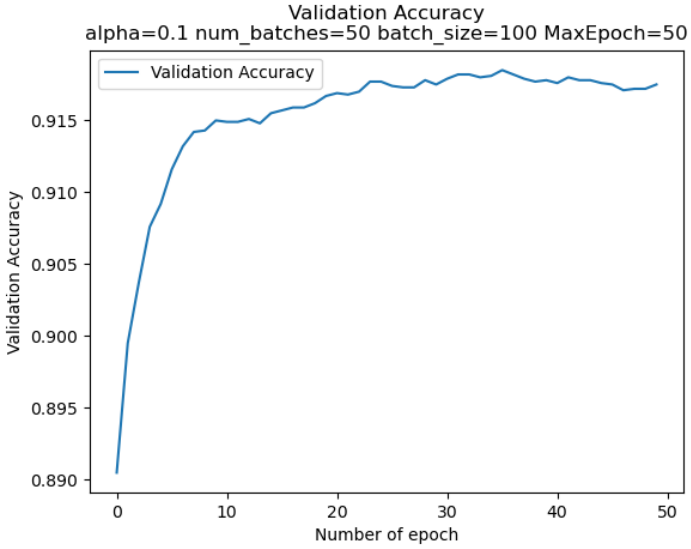
    ```
    No results
    ZeroDivisionError: float division by zero
    ```

- subtracting 5e-6 from the step size each iteration

  the idea here is there are 400 * 50 iterations and the step size started at 0.1. The step size approaches zero on the last iteration.

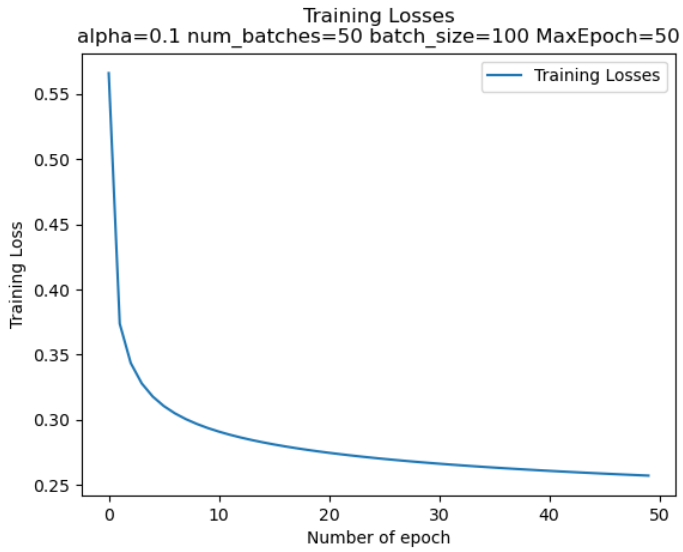
  ```
  At epoch 35 val: 0.9185 test: 0.9244
  ```

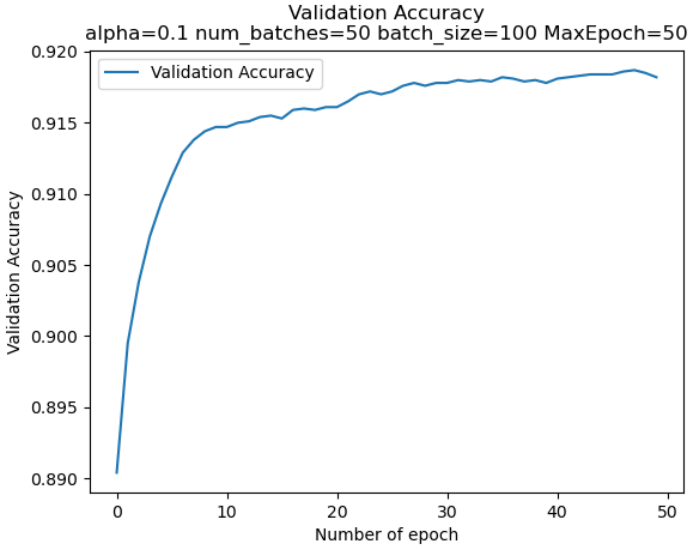




- subtracting 2.5e-6 from the step size each iteration

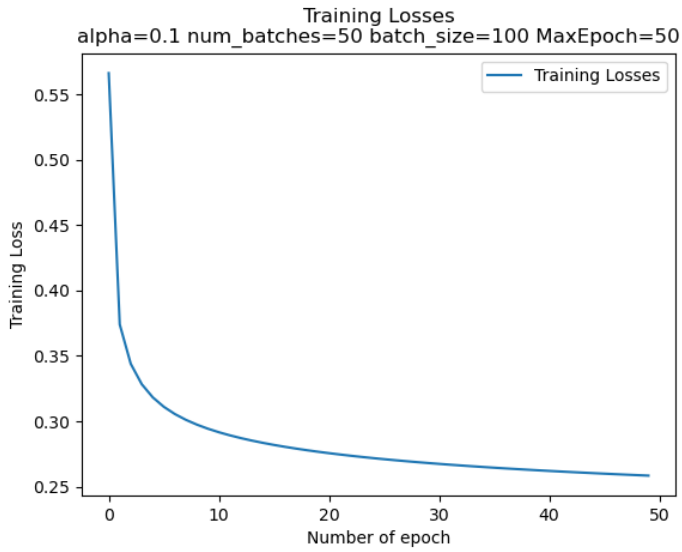
```
At epoch 47 val:  0.9187 test: 0.9235
```

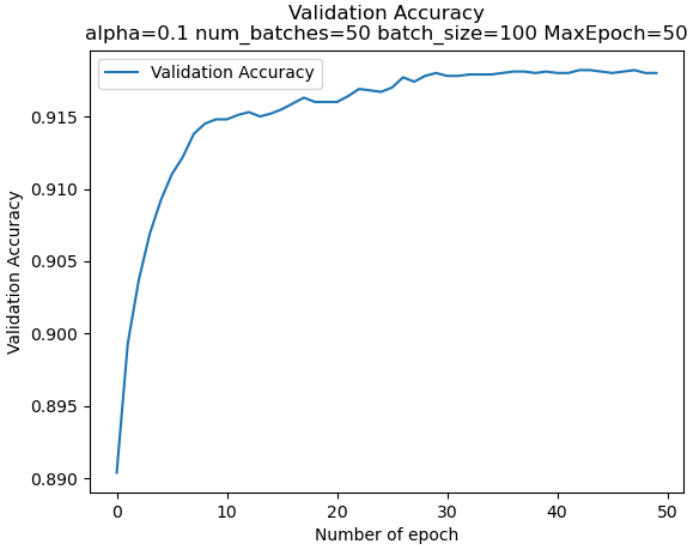




- subtracting 1e-6 from the step size each iteration

```
At epoch 47 val:  0.9182 test: 0.9234
```





- Summary of results

Annealing Scheduler	Best Epoch	Validation Accuracy	Test Accuracy
no annealing	49	0.919	0.9233
$\text{step}_{\text{size}}$ /= 2	n/a	n/a	n/a
$\text{step}_{\text{size}}$ -= 5e-6	35	0.9185	0.9244
$\text{step}_{\text{size}}$ -= 2.5e-6	47	0.9187	0.9235
$\text{step}_{\text{size}}$ -= 1e-6	47	0.9182	0.9234

Conclusion

Changing the annealing scheduler had little impact on the performance of the classifier after training. The annealing scheduler where the step size was reduced by 5e-6 each iteration ended up having the best test accuracy, however, this difference was small and could be due to just chance. The validation accuracy was still worse than the no annealing scheduler. Annealing the step size in this case might have helped slightly with making the learned classifier better.

As predicted, annealing the step size too quickly produced results that were either strictly worse, or even produced no results at all due to arithmetic overflow.

Annealing the step size is a useful technique in gradient descent methods, however, in this specific case it did not seem to have much of an effect.