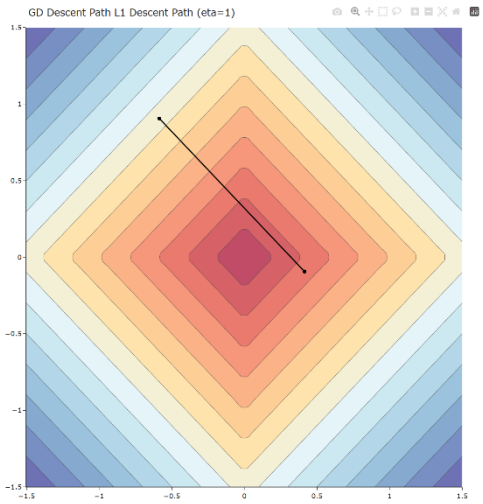
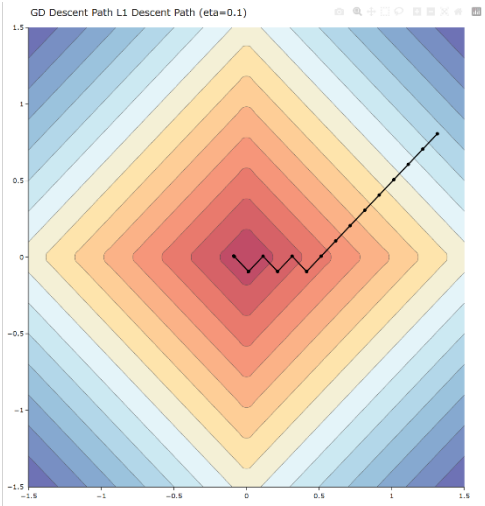
Practical Part:

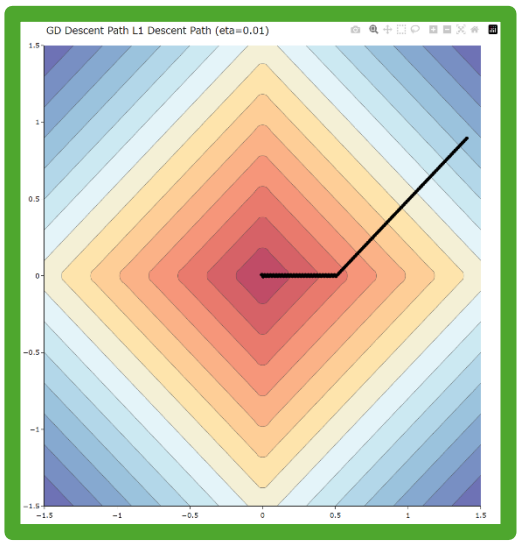
2.1.1:

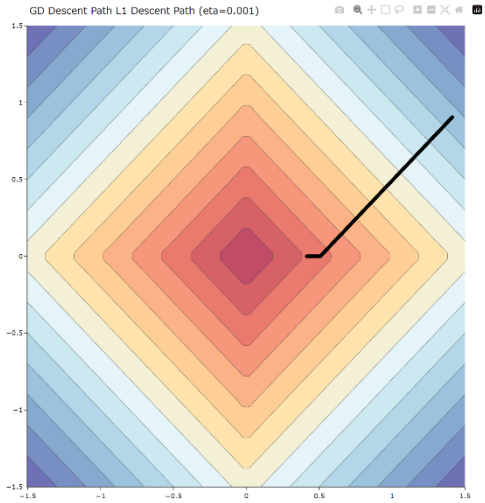
1)

L1:

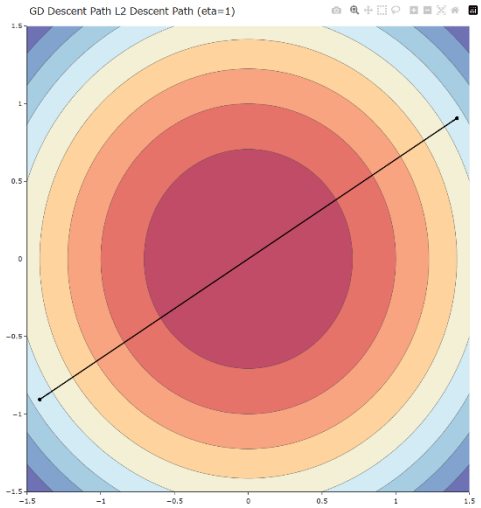


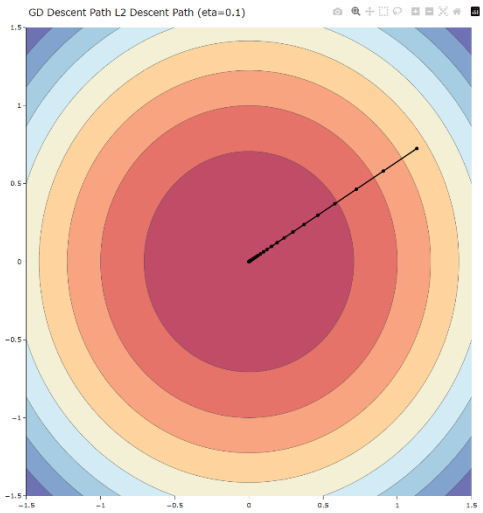


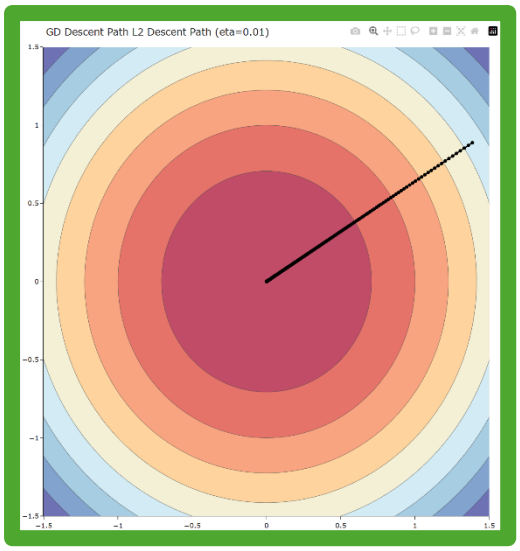


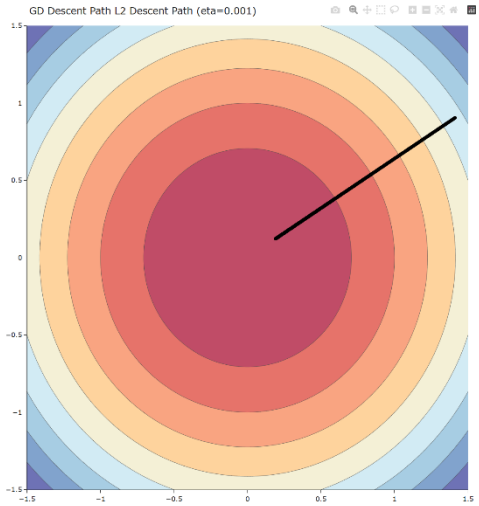


L2:









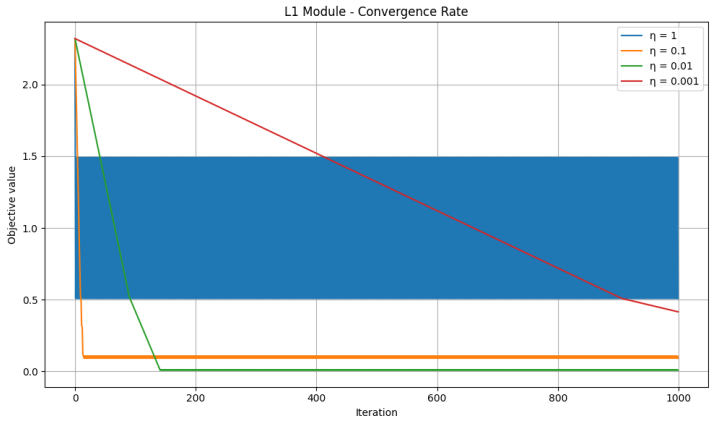
At the plots for **η = 0.01**, we can see a difference between the L1 and L2 module results. In the L1 module, the descent path was not direct and included a sharp turn on the way to the minimum, while the L2 module showed a smooth, direct path with no turns. Additionally, the step sizes behaved differently: in the L1 module, the steps remained constant throughout the descent, whereas in the L2 module, the steps gradually became smaller as the weights approached the minimum. These behaviors are consistent with the gradients of the two functions—L1 has a constant gradient (based on sign), which leads to uniform steps and directional changes, while L2 has a gradient proportional to the weights, resulting in a smooth trajectory that slows down near the minimum.

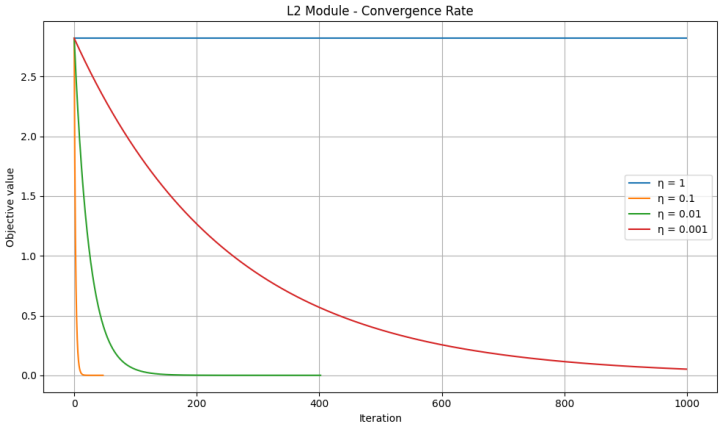
2)

a) sharp turns- the path often includes sharp turns especially when one of the coordinates crosses zero. This happens because the gradient of the L1 module is the sign function that changes instantly when zero is passed to the other sign.

b) constant step size – the step size doesn't change with the progress and that's because the norm gradient keeps going down or up by +-1 times the step size because it's the sign function.

3)





We can see a few important patterns in these plots:

L1 module – With a fixed learning rate of 1, the optimization moved back and forth, overshooting the minimum on both sides. This makes sense, as the step size was too large and the gradient is constant in sign, causing the algorithm to "bounce" past the minimum. On the other hand, the learning rate of 0.001 was too small, and the optimizer failed to reach a sufficiently low loss even after 1000 iterations. In fact, all learning rates show constant-rate movement toward the minimum, but none reach a very low loss. This can be explained by the fact that the gradient of the L1 norm is simply ±1, and when multiplied by a fixed step size, it results in constant updates that don't shrink as the algorithm approaches the minimum. Therefore, the loss flattens out above the optimal value.

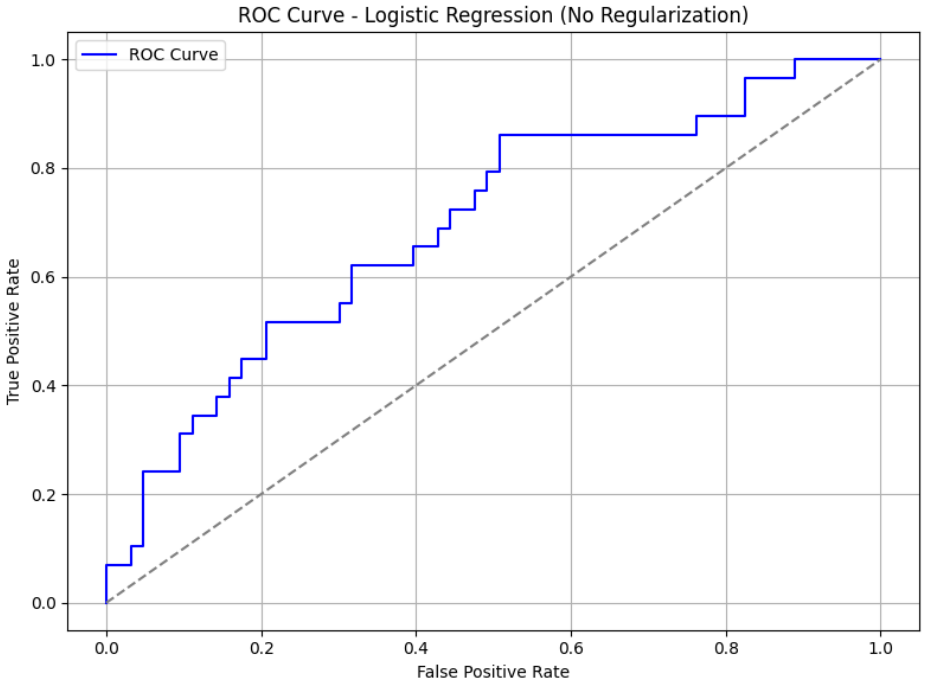
L2 module – Here too, a learning rate of 1 was too large and caused the optimizer to overshoot the minimum repeatedly, leading to a flat or erratic loss curve. However, because the L2 norm is always positive and smooth, the curve stays in the positive range. At the other extreme, η = 0.001 again led to very slow progress and failed to minimize the loss within 1000 iterations. The other learning rates (0.1 and 0.01) performed well and reached low loss values early, indicating they were in a good range. Unlike the L1 module, the L2 loss decreased with non-constant step sizes — the curves are not linear — which makes sense because the L2 gradient is proportional to 2w. As the weights get smaller, the gradient also gets smaller, so the step size decreases naturally, allowing smooth convergence toward the minimum.

4)

The L1 module achieved a lowest loss of 0.008120 using a fixed learning rate of 0.01, while the L2 module reached a minimum loss of 0.000000002, achieved by 0.01 learning rate. This difference stems from the nature of the gradient in each objective. As explained earlier, the L1 gradient is based on the sign function, which results in a constant-magnitude update. Because the step size remains fixed, the optimizer cannot make increasingly smaller updates as it nears the minimum, causing it to overshoot or stall before reaching a very small loss. In contrast, the L2 gradient is 2w, which scales with the magnitude of the weights. As the weights shrink, the gradient naturally becomes smaller, allowing the step size to decrease accordingly. This enables gradient descent to approach the true minimum smoothly without overshooting, which is reflected in the fact that the L2 loss reached zero in our results.

2.2:

5)



6)

```
Optimal α* = 0.7126
Test error with α* = 0.7126: 0.3913
```

7)

```
λ = 0.001: train error = 0.2905, validation error = 0.3054
λ = 0.002: train error = 0.2858, validation error = 0.2973
λ = 0.005: train error = 0.2892, validation error = 0.3081
λ = 0.01: train error = 0.2878, validation error = 0.3108
λ = 0.02: train error = 0.2959, validation error = 0.3108
λ = 0.05: train error = 0.3041, validation error = 0.3135
λ = 0.1: train error = 0.2946, validation error = 0.3189
Best λ : 0.002
Test error using best λ = 0.002: 0.2826
```