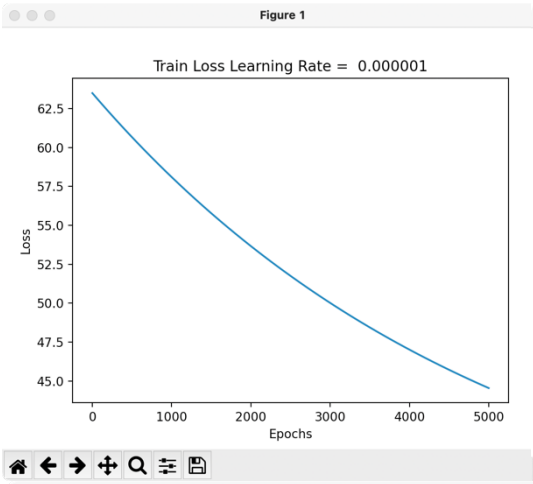
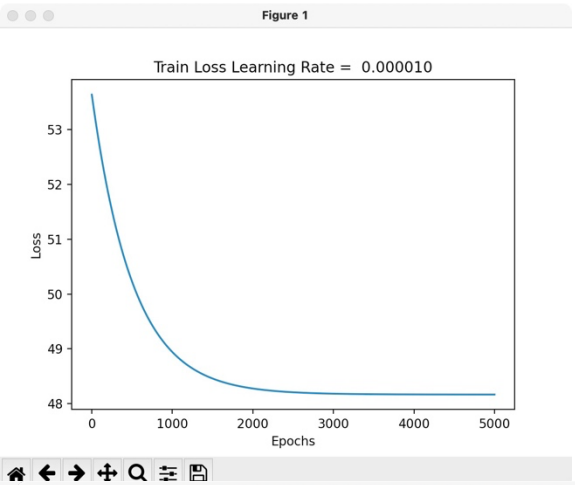
Name: Chengxu Xu
CS 534
IA1

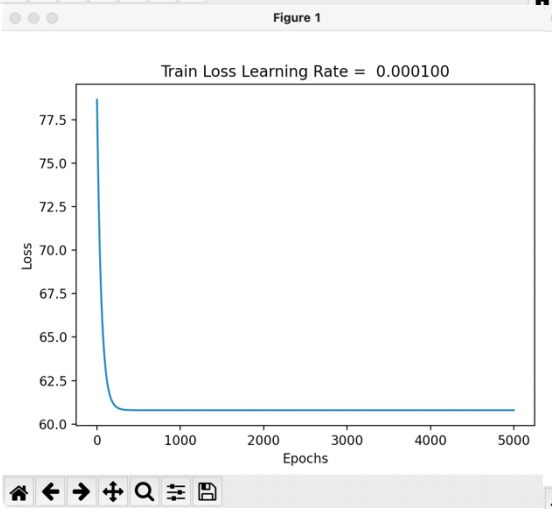
Part1

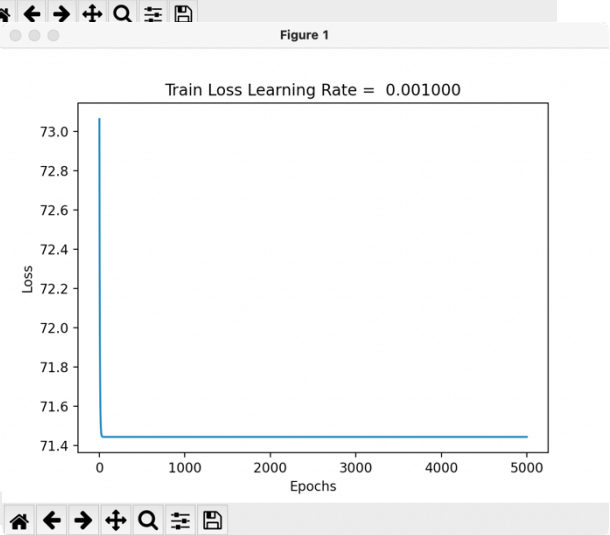
Question(a):

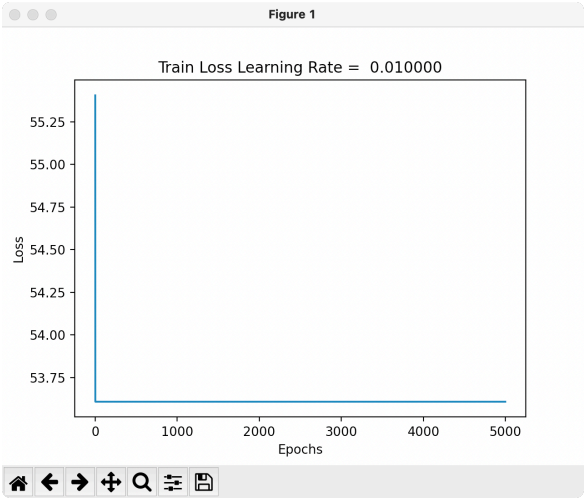
The generated images are as follows：

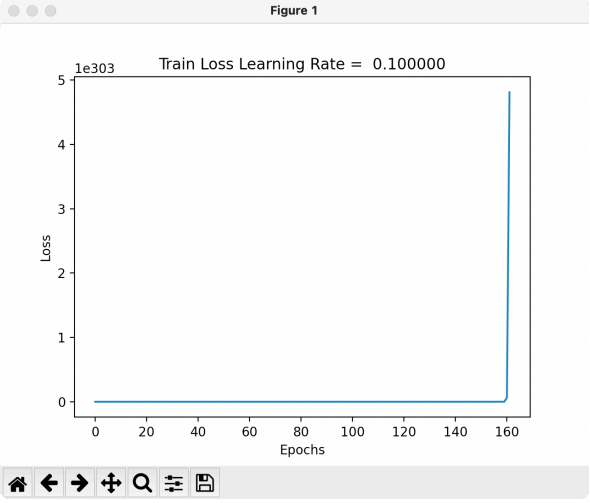


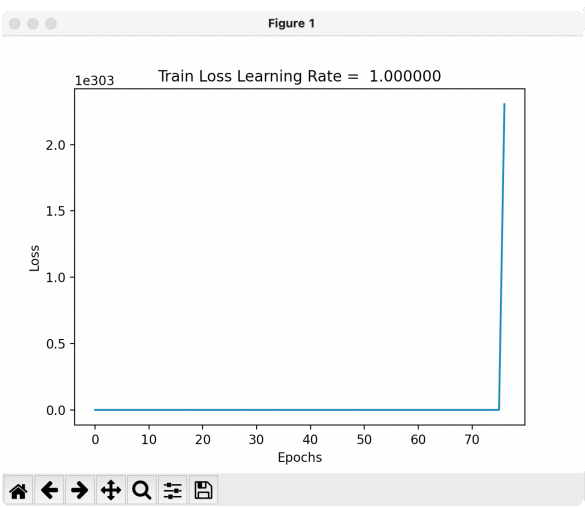


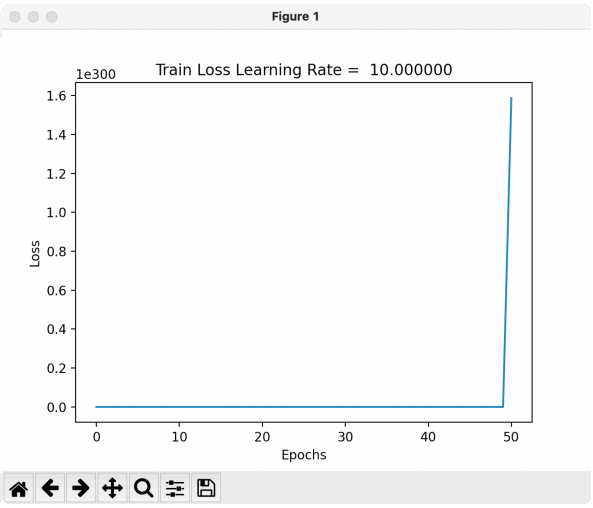












Based on the image observations, a learning rate of 10^{-5} is good for this particular dataset. Learning rate bigger than 10^{-1} will make gradient descent diverge.

Question(b):

```
part 1 (b)

When Learning rate =  0.000001
MSE = 44.72213216223253
When Learning rate =  0.000010
MSE = 32.36112094259405
When Learning rate =  0.000100
MSE = 39.78601177850811
When Learning rate =  0.001000
MSE = 36.45457522606115
When Learning rate =  0.010000
MSE = 37.2189496567443
```

Based on the results of the test, The difference in validation MSE between different learning rates is not significant, but a learning rate of 10^{-5} resulting in the best validated MSE at different convergent learning rates.

When the validation MSE is nearly identical, we try different learning rates and observe their curves. We should choose as large a learning rate as possible to reduce the number of iterations while avoiding overstep, i.e., while maintaining the converge.

Question(c):

With the learning rate set to 10^{-5}, the learning weights for each feature are as follows:

```
part 1 (c)

bedrooms : -0.2496845244881061
bathrooms : 0.3355547476349569
sqft_living : -0.01790390011088288
sqft_lot : -1.312320969758126
floors : 0.3091111479094686
view : 0.31268232981009175
condition : 0.2666585474336688
grade : 1.5268651864933933
sqft_above : -0.16117690222720138
sqft_basement : 1.0170029207197782
yr_built : 1.3598778178630386
age_since_renovated : 0.17253740908148538
zipcode : 0.6577312342169692
lat : 0.3642871743261038
long : -0.5163600790411721
sqft_living15 : 1.3486253510258126
sqft_lot15 : 1.0341501114766687
month : 0.7250785661283199
day : 1.7833015940128318
year : 1.0224012240357905
waterfront : 1.1531028282276958
dummy : 0.07936754940904583
```

In terms of data, day feature is the most important factor in determining house prices, as it accounts for the largest weight.

Part2

Question(a):

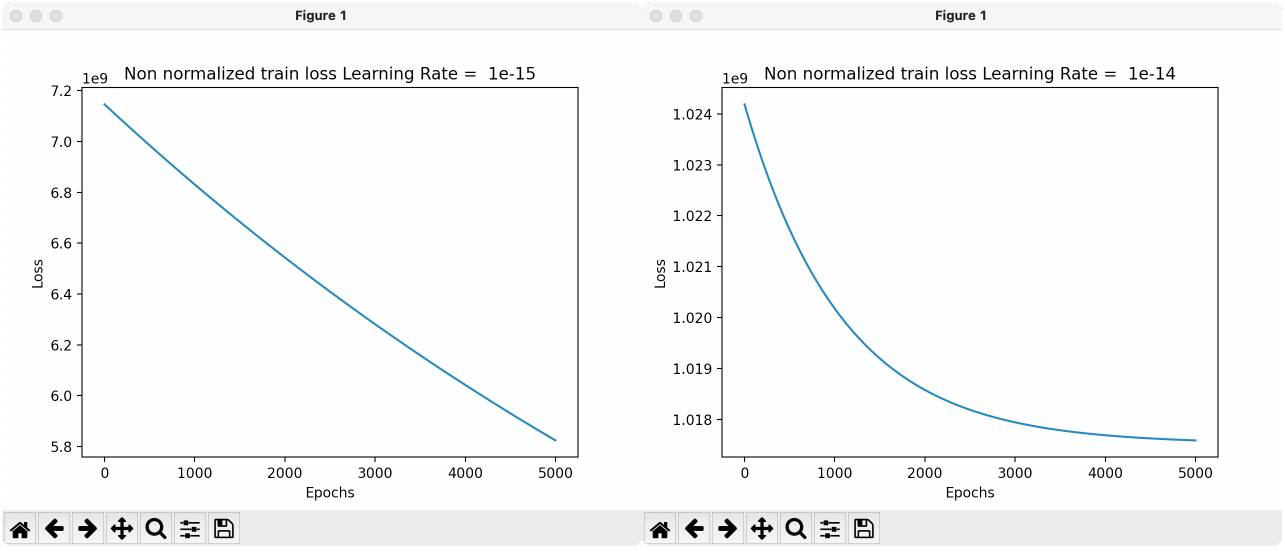


Figure 1

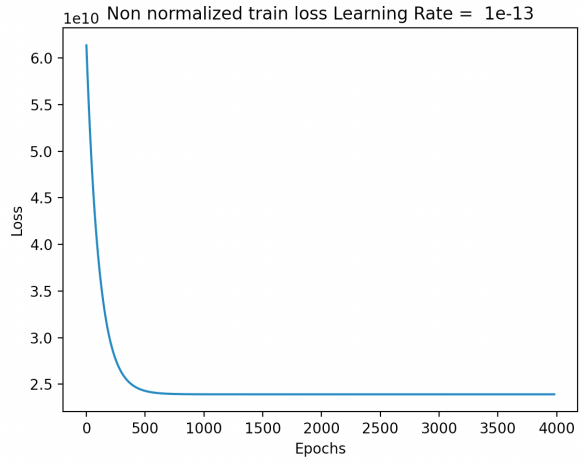


Figure 1

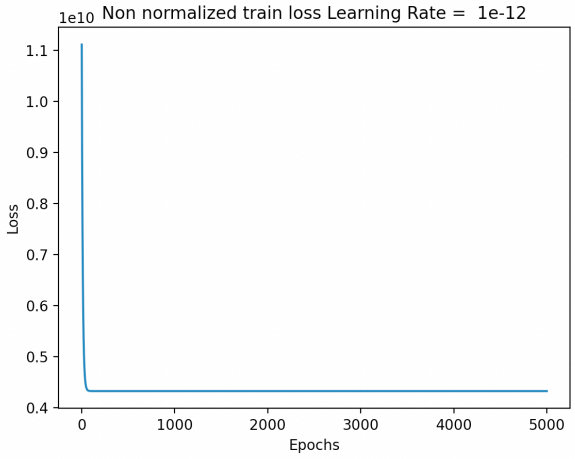


Figure 1

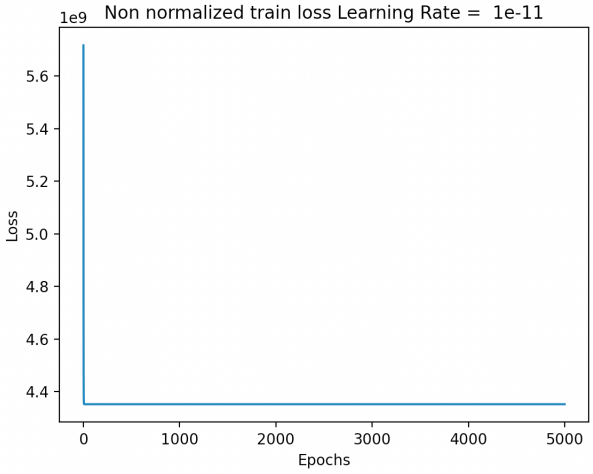


Figure 1

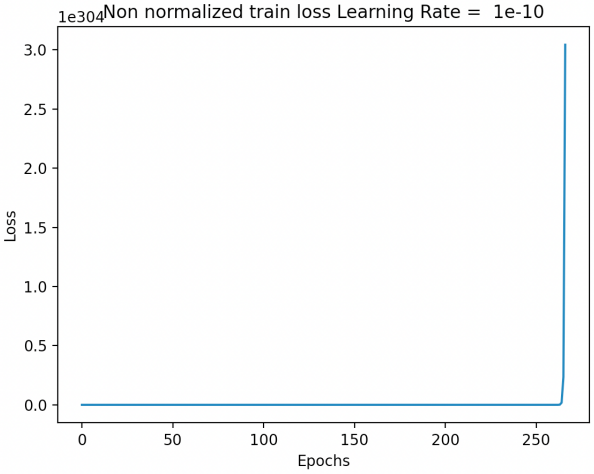


Figure 1

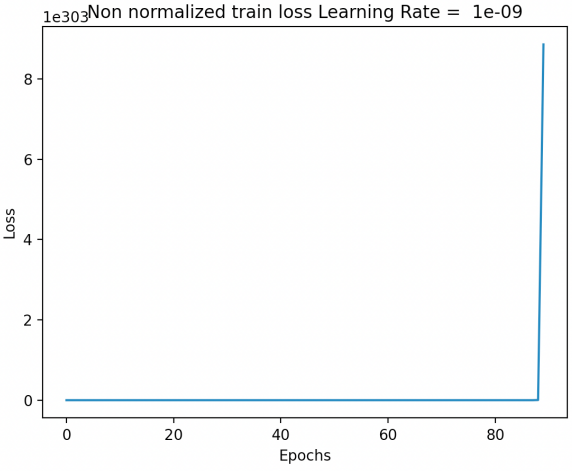
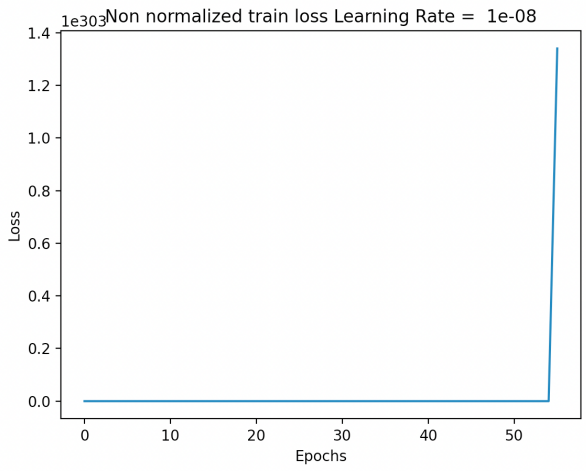


Figure 1



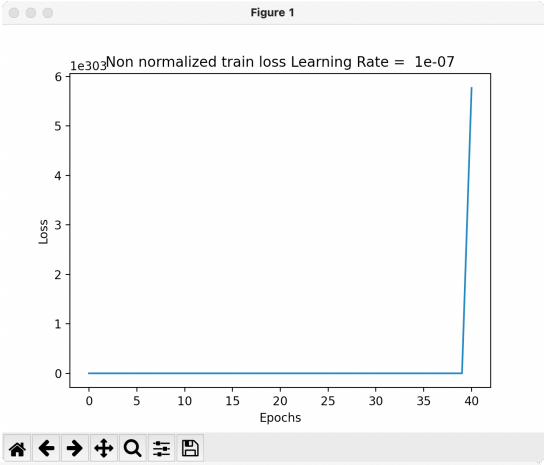


Figure 1
Non normalized train loss Learning Rate = 1e-07
1e303
Loss
Epochs

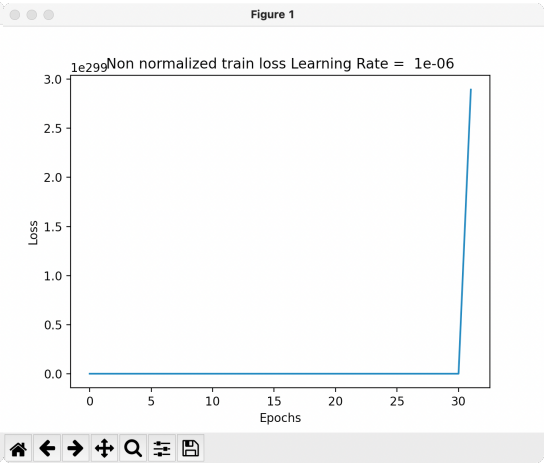


Figure 1
Non normalized train loss Learning Rate = 1e-06
1e299
Loss
Epochs

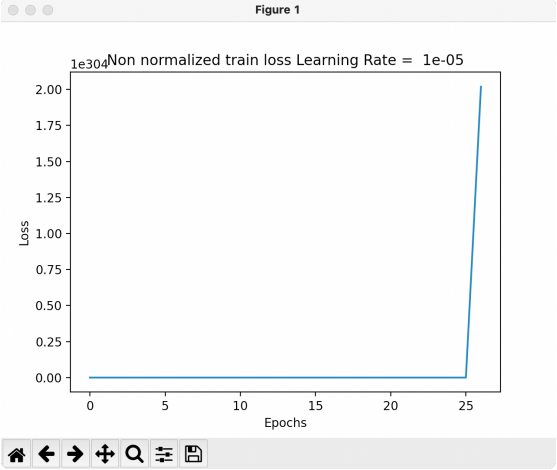


Figure 1
Non normalized train loss Learning Rate = 1e-05
1e304
Loss
Epochs

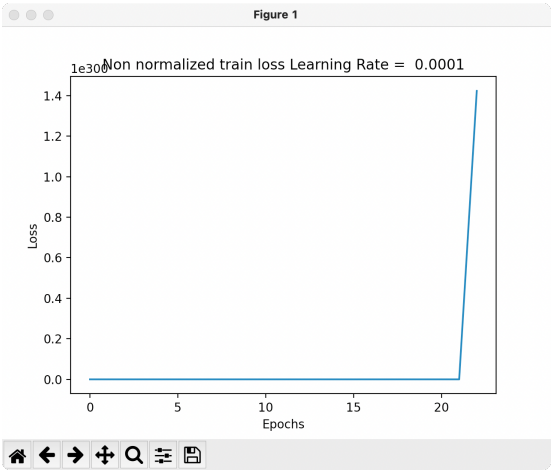


Figure 1
Non normalized train loss Learning Rate = 0.0001
1e300
Loss
Epochs

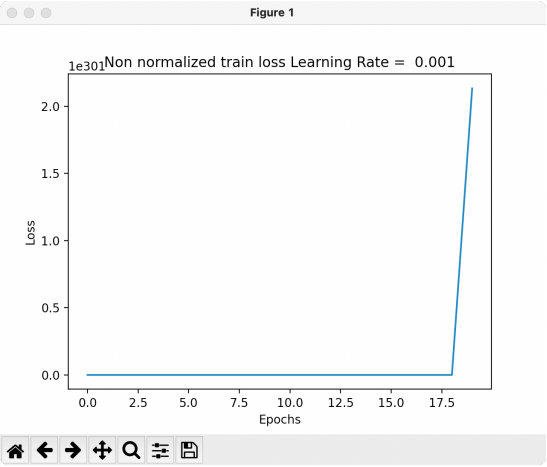


Figure 1
Non normalized train loss Learning Rate = 0.001
1e301
Loss
Epochs

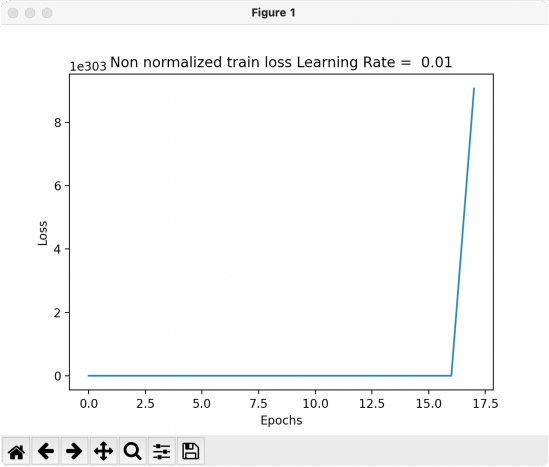
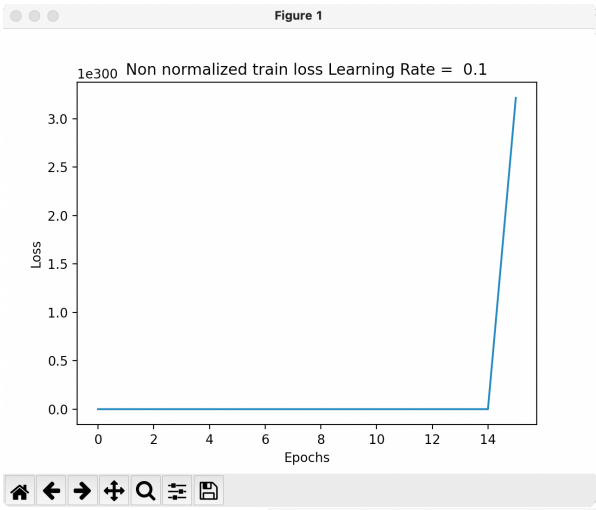
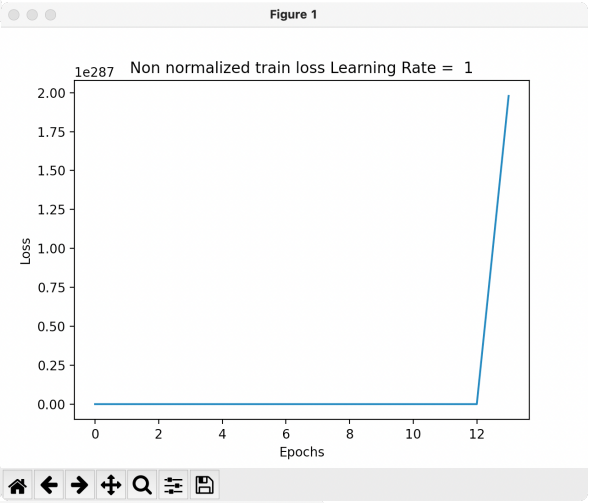
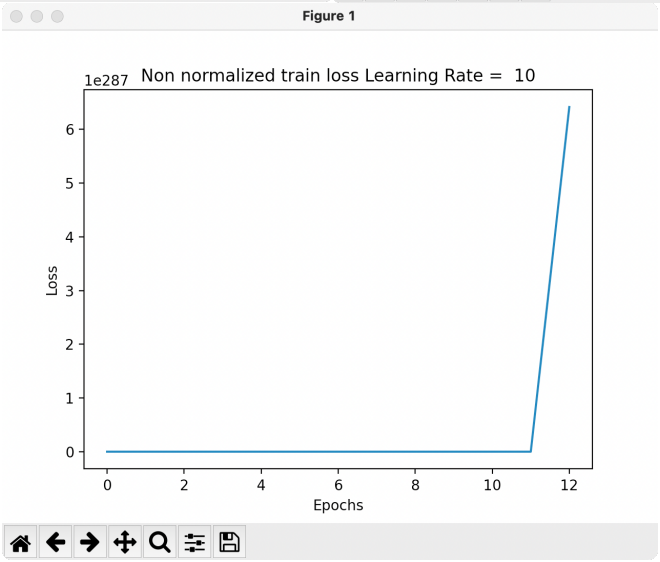


Figure 1
Non normalized train loss Learning Rate = 0.01
1e303
Loss
Epochs







According to the analysis of the training test results, the learning rate of 10^{-14} is a suitable learning rate for un-normalized data, and the phenomenon of divergence appears when the learning rate is greater than 10^{-10}

Normalized datasets are easier to train, because unnormalized datasets require smaller learning rates, which means more iterations and slower training progress.

Question(b):

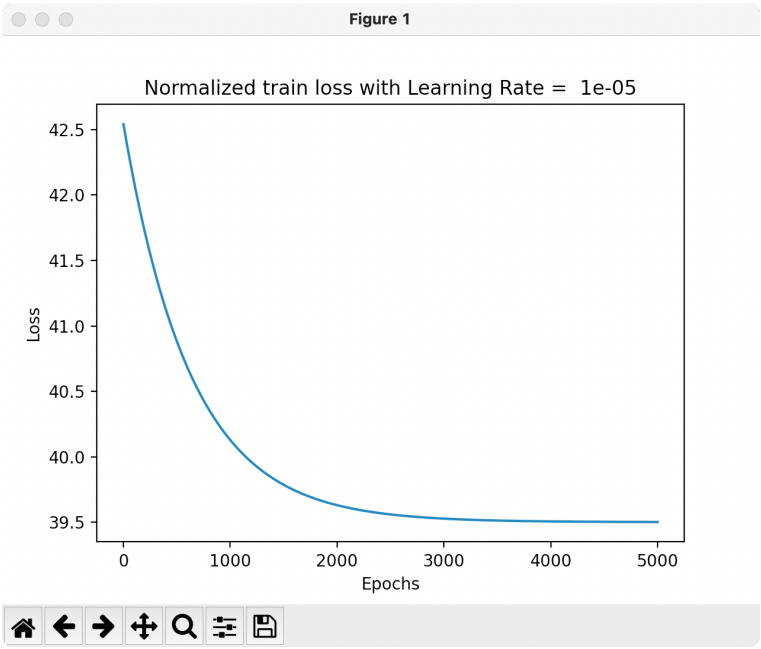
```
part 2 (b)

When Learning rate =  1e-15
MSE = 5824268823.636486
When Learning rate =  1e-14
MSE = 1017585236.3791627
When Learning rate =  1e-13
MSE = 5299372454.0855
When Learning rate =  1e-12
MSE = 1976526231.1108575
When Learning rate =  1e-11
MSE = 1465094909.8461988
When Learning rate =  1e-10
MSE = nan
```

Normalized data is relatively more concentrated and less noisy, making it more suitable for training and analysis, whereas the data distribution within an unstandardized dataset is too diffuse and therefore yields a much larger MSE.

Therefore, it is necessary to normalize the data before training or otherwise pre-process them to make them easier to train and analyze.

Part3

Question:



```
part 3

Remove sqft_living15 feature, learning rate =  1e-05
 MSE = 39.502957594261304
bedrooms : -0.4951190620760806
bathrooms : 0.970342610039644
sqft_living : 0.9589745691601774
sqft_lot : 0.6628180285451251
floors : 0.1258871479889667
view : 0.08406816288664617
condition : -0.7080275727977798
grade : 1.5539036342618942
sqft_above : 0.04967773022853434
sqft_basement : 0.9321777638622906
yr_built : 0.30196356760811405
age_since_renovated : 1.3791276850791885
zipcode : -0.8229042311297592
lat : 0.9311872626756301
long : 0.7384771213786574
sqft_lot15 : 0.2815798174718652
month : -1.2570975444237824
day : -0.6035751283462114
year : 0.6138421824049672
waterfront : 0.12888662765874348
dummy : 0.1429553622448076
```

Compared with part1 (c), it performs better and the trend of 'MSE' decrease with increasing number of iterations is more desirable from the image.
After removing the feature 'sqft_living15', the weight of feature 'sqrt_living' increases significantly probably because correlated features impress each other and cause the final weight in the function to decrease.

I think the weights (w1 and w2) when learning with two features, will be lower in comparison with w1 which is learned with just x1, because the two redundant features will affect each other, causing the algorithm to assign them weights decrease