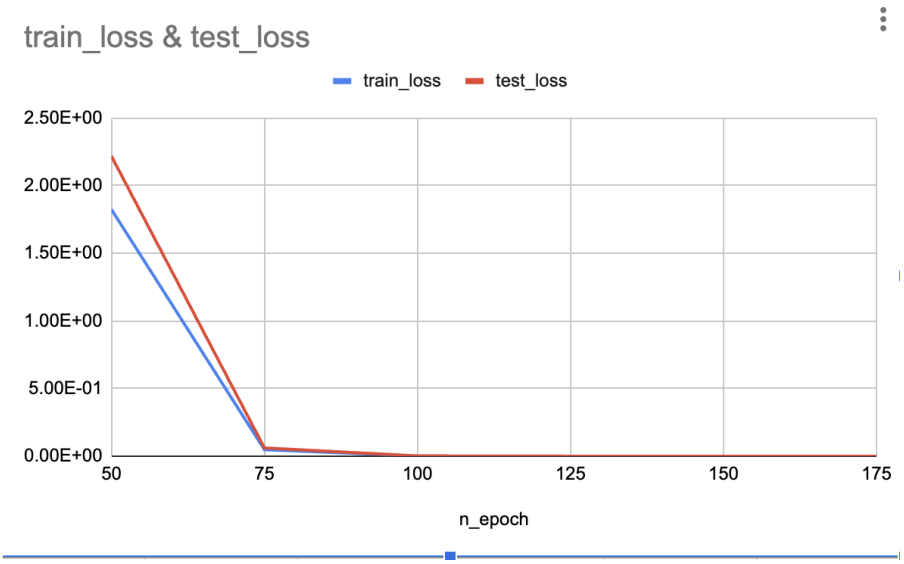
539ML HW2

```
alpha ==  0.1
train_loss ==  [[1.82430292]]
test_loss ==  [[2.21835191]]
n_epoch ==: 50

alpha ==  0.1
train_loss ==  [[0.04758335]]
test_loss ==  [[0.06074704]]
n_epoch ==: 75

alpha ==  0.1
train_loss ==  [[0.0014202]]
test_loss ==  [[0.0018851]]
n_epoch ==: 100

alpha ==  0.1
train_loss ==  [[4.51092368e-05]]
test_loss ==  [[6.14148846e-05]]
n_epoch ==: 125

alpha ==  0.1
train_loss ==  [[1.46893525e-06]]
test_loss ==  [[2.03081331e-06]]
n_epoch ==: 150

alpha ==  0.1
train_loss ==  [[4.82855929e-08]]
test_loss ==  [[6.736461e-08]]
n_epoch ==: 175
```

```
alpha ==  0.1
train_loss ==  [[4.82855929e-08]]
test_loss ==  [[6.736461e-08]]
n_epoch ==: 175

alpha ==  0.2
train_loss ==  [[0.00084606]]
test_loss ==  [[0.00113479]]
n_epoch ==: 50

alpha ==  0.2
train_loss ==  [[4.24435391e-16]]
test_loss ==  [[5.99758514e-16]]
n_epoch ==: 150
```

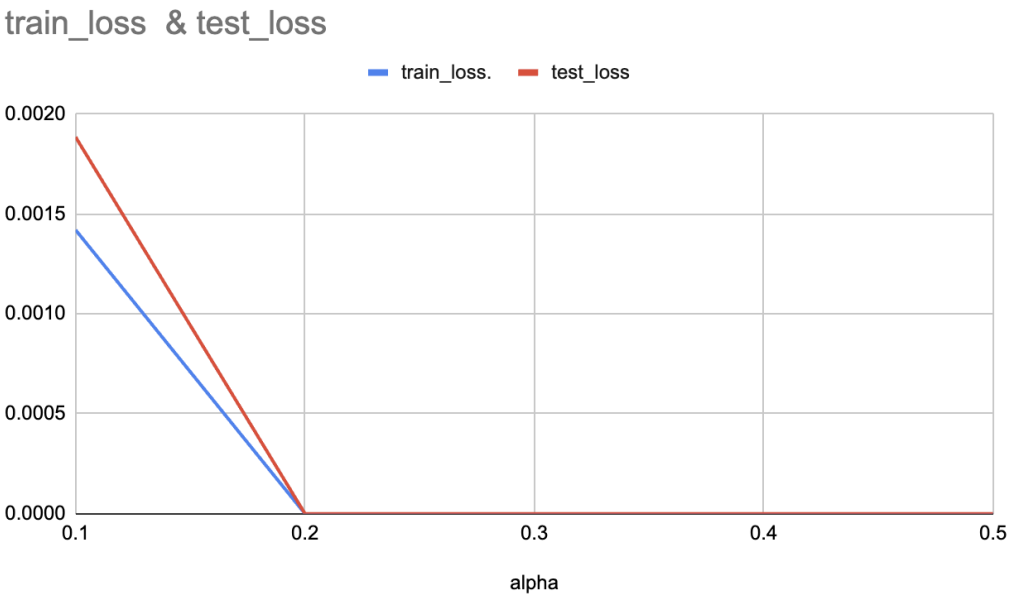


From the output we can see:

Keeping the Learning rate (alpha) constant at 0.1, the number of epochs are increased from 50 till 125 increasing by 25 each time

- Effect of alpha on Loss: As alpha decreases from 0.1 to 0.2, both the training and testing losses decrease. This suggests that smaller values of alpha (learning rate) lead to better convergence and reduced loss.
- Effect of n_epoch on Loss: For a fixed alpha value, as the number of epochs (n_epoch) increases, the training and testing losses continue to decrease. This implies that allowing the model to undergo more training epochs results in better performance, at least within the range of epochs considered.
- Interactions: There is a consistent pattern across different values of alpha and n_epoch. For example, for alpha = 0.1, as n_epoch increases, the losses decrease. This trend is also observed for alpha = 0.2.
- Convergence: The training and testing losses reach very small values, approaching numerical precision (e.g., 1.46893525e-06), indicating that the model is converging well on the given task.

```
alpha ==  0.1
train_loss ==  [[0.0014202]]
test_loss ==  [[0.0018851]]
n_epoch ==: 100

alpha ==  0.2
train_loss ==  [[5.86394303e-10]]
test_loss ==  [[8.24522311e-10]]
n_epoch ==: 100

alpha ==  0.30000000000000004
train_loss ==  [[8.10421368e-17]]
test_loss ==  [[1.145517e-16]]
n_epoch ==: 100

alpha ==  0.4
train_loss ==  [[2.88247834e-24]]
test_loss ==  [[4.0787881e-24]]
n_epoch ==: 100

alpha ==  0.5
train_loss ==  [[2.94817042e-28]]
test_loss ==  [[3.03127691e-28]]
n_epoch ==: 100
```



Keeping the number of epochs constant at 100, the learning rate (alpha) is increased from 0.1 till 0.7increasing by 0.1 each time

- Effect of alpha on Loss: As alpha decreases from 0.1 to 0.5, both the training and testing losses continue to decrease. This is consistent with the typical behavior of learning rates: smaller learning rates often lead to slower but more stable convergence.
- Convergence: The training and testing losses reach very small values, approaching numerical precision (e.g., 2.94817042e-28). This suggests that the model is converging well for the given learning rates.
- Consistency: The relationship between alpha and the losses appears consistent; as alpha decreases, the losses decrease. This consistency aligns with the expectation that lower learning rates allow the model to converge more effectively.