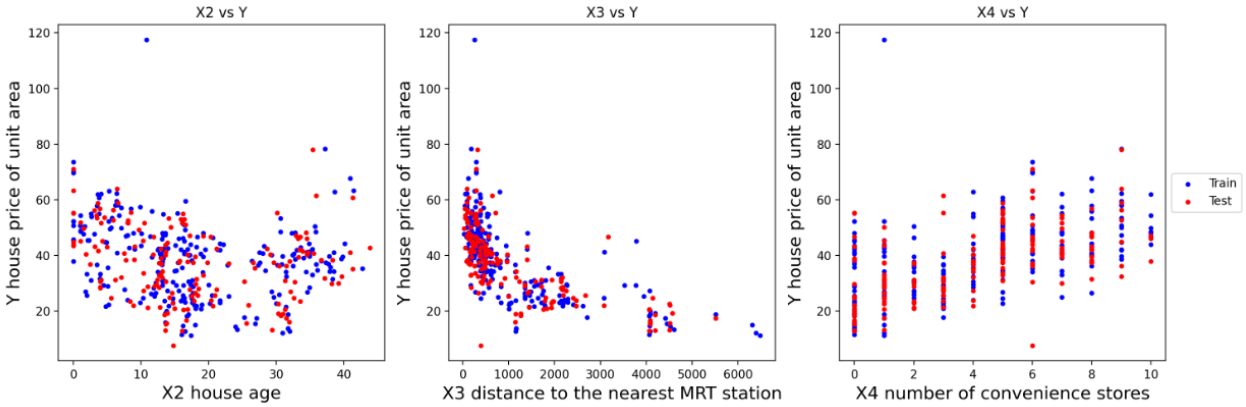
Machine Learning Homework 3

1.

```python
# 1. Dataset Loading & splitting:
# randomly pick 60% training data、40% test data

train_set = df.sample(frac=0.6, random_state = 50) # random_state用來固定同一組隨機數據
test_set  = df.drop(train_set.index)
```

2.



3.

```python
# 3. Define loss function (Mean Square Error)
# Using basic mathematical operations in NumPy

def mse_loss(error):
    MSE = np.mean(np.square(error))
    return MSE

def error_func(predict, real):
    error = predict - real
    return error

def y_pred(X, beta):
    prediction = X.dot(beta)
    return prediction
```

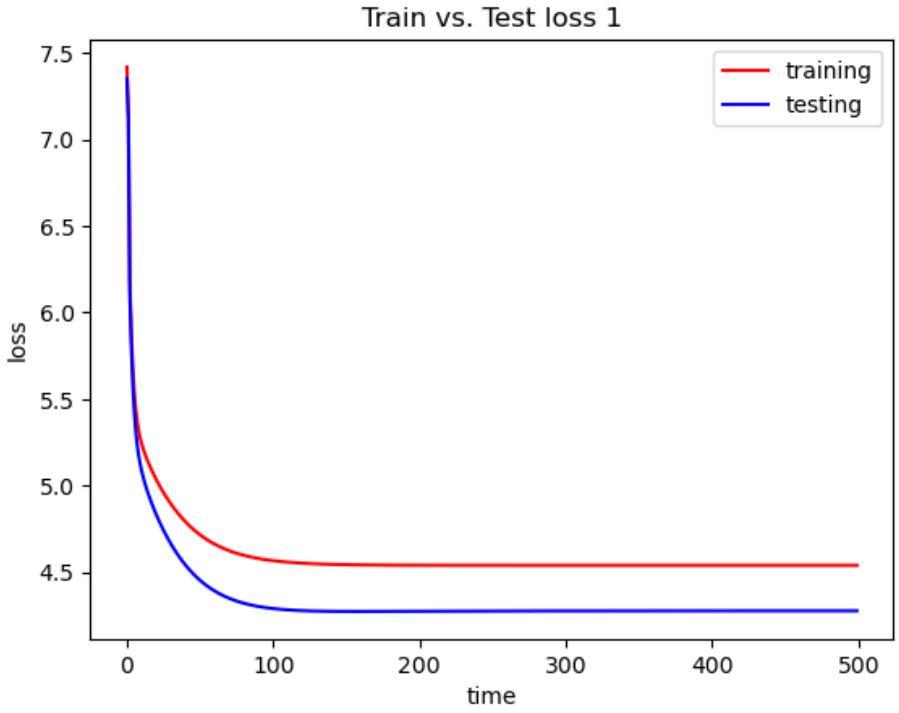
4.

```
Training loss 1:
1 epoch training loss: 1638.8626
2 epoch training loss: 113.7888
3 epoch training loss: 96.5312
4 epoch training loss: 93.9689
5 epoch training loss: 93.5849
6 epoch training loss: 93.5273
7 epoch training loss: 93.5187
8 epoch training loss: 93.5174
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 1:
1 epoch testing loss: 1561.5623
2 epoch testing loss: 85.7634
3 epoch testing loss: 72.8295
4 epoch testing loss: 71.6451
5 epoch testing loss: 71.7162
6 epoch testing loss: 71.8168
7 epoch testing loss: 71.8652
8 epoch testing loss: 71.8849
9 epoch testing loss: 71.8926
10 epoch testing loss: 71.8955
```

Beta：

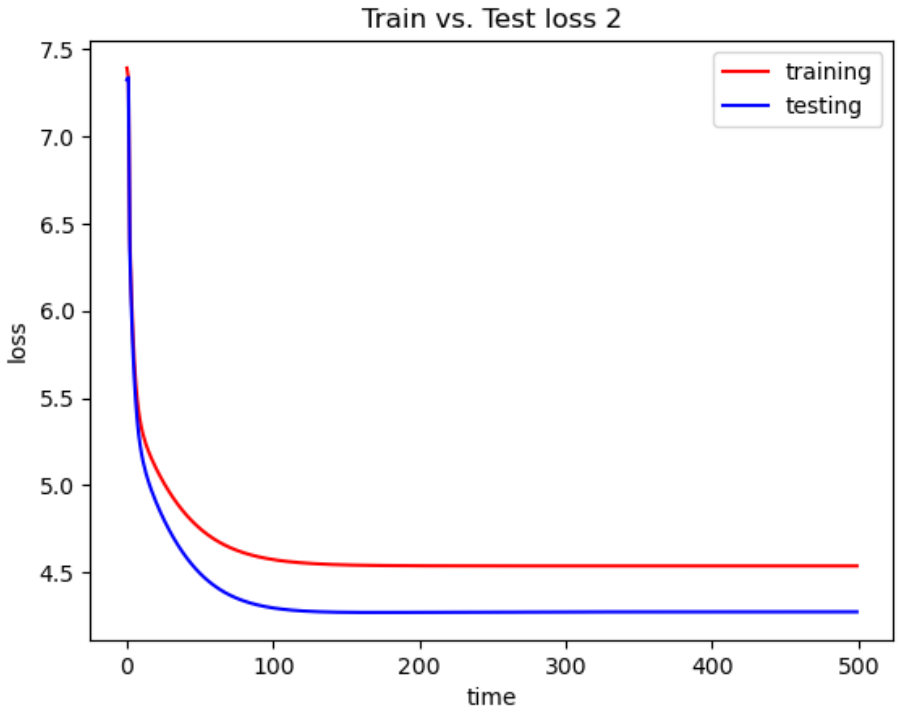
```
array([[ 4.36513734e+01],
       [-2.65753125e-01],
       [-5.43626147e-03],
       [ 1.26150970e+00]])
```



```
Training loss 2:                    Testing loss 2:
1 epoch training loss: 1658.6368    1 epoch testing loss: 1529.8699
2 epoch training loss: 112.1784     2 epoch testing loss: 87.9491
3 epoch training loss: 96.1909      3 epoch testing loss: 73.2008
4 epoch training loss: 93.9032      4 epoch testing loss: 71.6713
5 epoch training loss: 93.5730      5 epoch testing loss: 71.6994
6 epoch training loss: 93.5252      6 epoch testing loss: 71.8039
7 epoch training loss: 93.5183      7 epoch testing loss: 71.8585
8 epoch training loss: 93.5173      8 epoch testing loss: 71.8819
9 epoch training loss: 93.5172      9 epoch testing loss: 71.8913
10 epoch training loss: 93.5172     10 epoch testing loss: 71.8950
```

Beta：

```
array([[ 4.36523312e+01],
       [-2.65763460e-01],
       [-5.43649891e-03],
       [ 1.26140238e+00]])
```

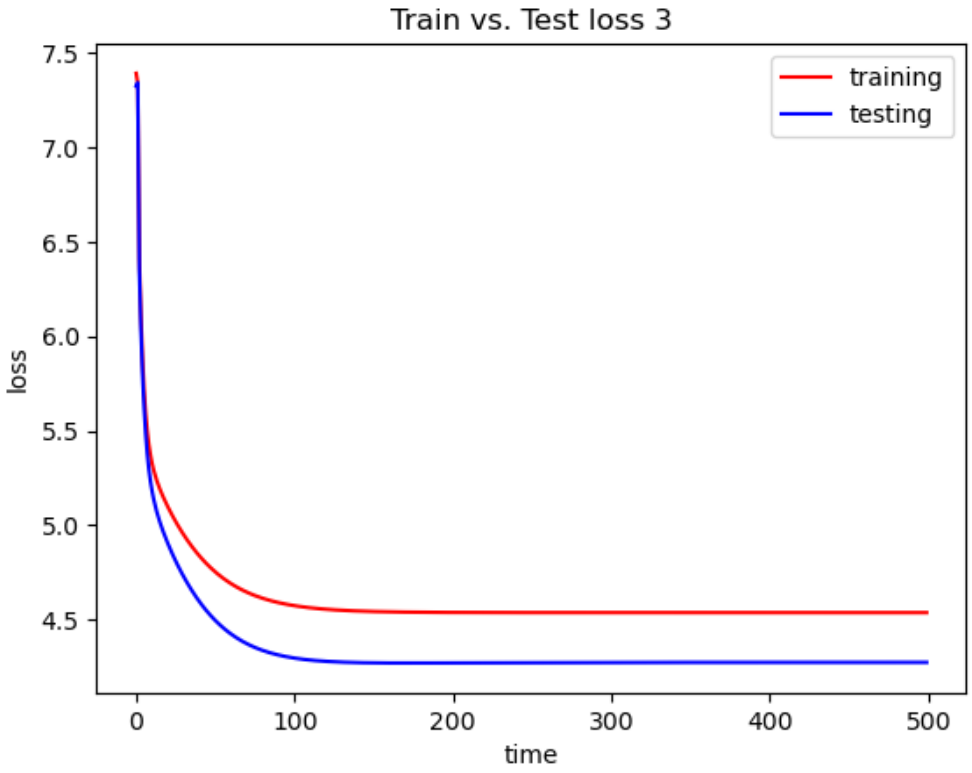


```
Training loss 3:
1 epoch training loss: 1619.7421
2 epoch training loss: 115.9243
3 epoch training loss: 97.0052
4 epoch training loss: 94.0647
5 epoch training loss: 93.6032
6 epoch training loss: 93.5307
7 epoch training loss: 93.5193
8 epoch training loss: 93.5175
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 3:
1 epoch testing loss: 1512.8913
2 epoch testing loss: 89.6924
3 epoch testing loss: 73.5227
4 epoch testing loss: 71.7014
5 epoch testing loss: 71.6878
6 epoch testing loss: 71.7935
7 epoch testing loss: 71.8529
8 epoch testing loss: 71.8792
9 epoch testing loss: 71.8901
10 epoch testing loss: 71.8944
```

Beta：

```
array([[ 4.36491907e+01],
       [-2.65729926e-01],
       [-5.43571371e-03],
       [ 1.26175448e+00]])
```

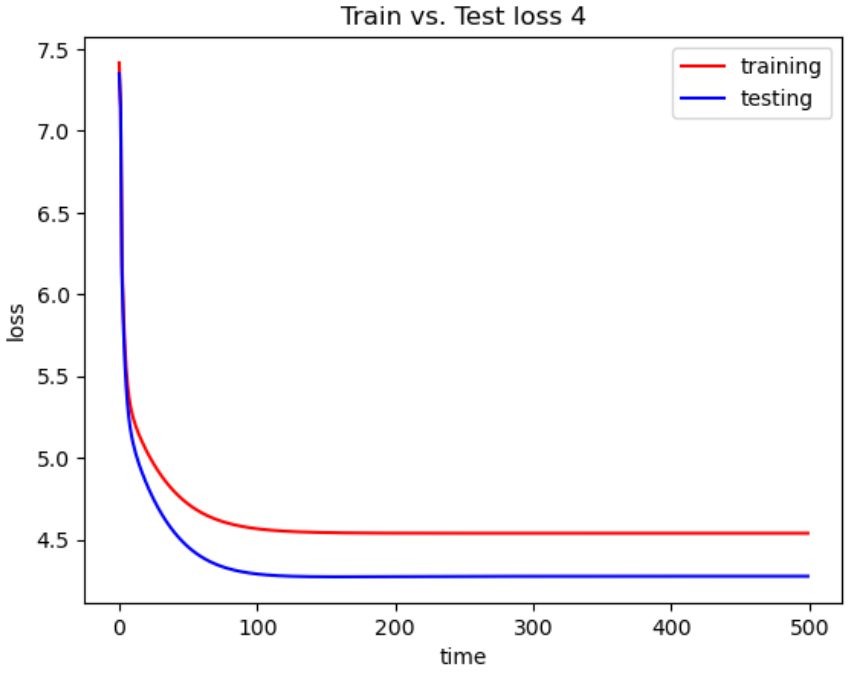


```
Training loss 4:
1 epoch training loss: 1664.6028
2 epoch training loss: 111.7846
3 epoch training loss: 96.1102
4 epoch training loss: 93.8880
5 epoch training loss: 93.5703
6 epoch training loss: 93.5248
7 epoch training loss: 93.5183
8 epoch training loss: 93.5173
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 4:
1 epoch testing loss: 1557.9476
2 epoch testing loss: 85.9667
3 epoch testing loss: 72.8624
4 epoch testing loss: 71.6470
5 epoch testing loss: 71.7146
6 epoch testing loss: 71.8156
7 epoch testing loss: 71.8646
8 epoch testing loss: 71.8847
9 epoch testing loss: 71.8925
10 epoch testing loss: 71.8955
```

Beta：

```
array([[ 4.36524174e+01],
       [-2.65764399e-01],
       [-5.43652017e-03],
       [ 1.26139275e+00]])
```

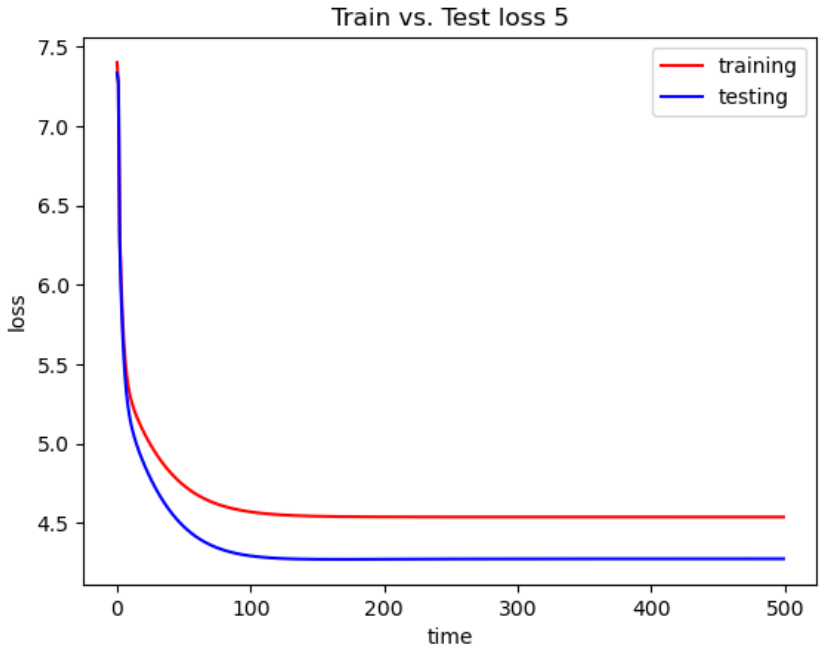


```
Training loss 5:
1 epoch training loss: 1636.3971
2 epoch training loss: 114.0815
3 epoch training loss: 96.5947
4 epoch training loss: 93.9814
5 epoch training loss: 93.5873
6 epoch training loss: 93.5278
7 epoch training loss: 93.5188
8 epoch training loss: 93.5174
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 5:
1 epoch testing loss: 1529.5974
2 epoch testing loss: 88.0297
3 epoch testing loss: 73.2153
4 epoch testing loss: 71.6726
5 epoch testing loss: 71.6988
6 epoch testing loss: 71.8034
7 epoch testing loss: 71.8583
8 epoch testing loss: 71.8818
9 epoch testing loss: 71.8912
10 epoch testing loss: 71.8949
```

Beta：

```
array([[ 4.36507512e+01],
       [-2.65746475e-01],
       [-5.43610614e-03],
       [ 1.26157949e+00]])
```

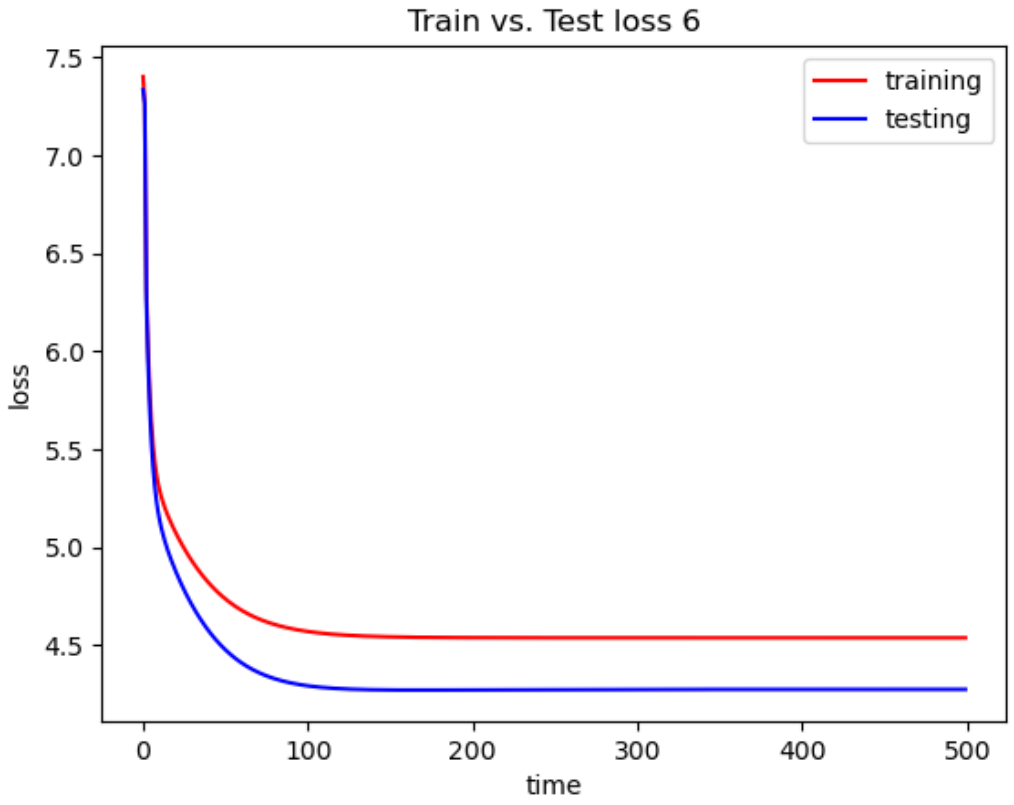


```
Training loss 6:
1 epoch training loss: 1636.9329
2 epoch training loss: 113.9849
3 epoch training loss: 96.5737
4 epoch training loss: 93.9773
5 epoch training loss: 93.5865
6 epoch training loss: 93.5276
7 epoch training loss: 93.5188
8 epoch training loss: 93.5174
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 6:
1 epoch testing loss: 1530.1608
2 epoch testing loss: 87.9429
3 epoch testing loss: 73.1997
4 epoch testing loss: 71.6713
5 epoch testing loss: 71.6995
6 epoch testing loss: 71.8039
7 epoch testing loss: 71.8586
8 epoch testing loss: 71.8819
9 epoch testing loss: 71.8913
10 epoch testing loss: 71.8950
```

Beta：

```
array([[ 4.36508280e+01],
       [-2.65747287e-01],
       [-5.43612539e-03],
       [ 1.26157085e+00]])
```

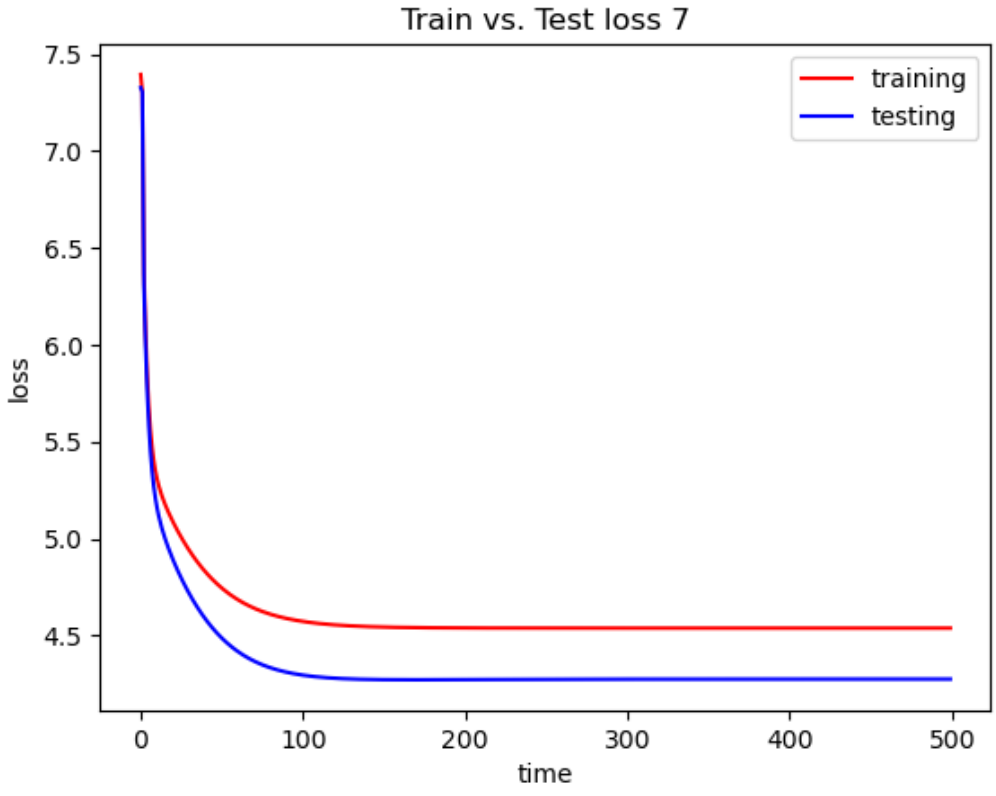


```
Training loss 7:
1 epoch training loss: 1627.3549
2 epoch training loss: 114.9701
3 epoch training loss: 96.7904
4 epoch training loss: 94.0207
5 epoch training loss: 93.5947
6 epoch training loss: 93.5291
7 epoch training loss: 93.5190
8 epoch training loss: 93.5175
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 7:
1 epoch testing loss: 1520.5462
2 epoch testing loss: 88.8309
3 epoch testing loss: 73.3609
4 epoch testing loss: 71.6855
5 epoch testing loss: 71.6933
6 epoch testing loss: 71.7986
7 epoch testing loss: 71.8557
8 epoch testing loss: 71.8806
9 epoch testing loss: 71.8907
10 epoch testing loss: 71.8947
```

Beta：

```
array([[ 4.36500235e+01],
       [-2.65738718e-01],
       [-5.43592373e-03],
       [ 1.26166104e+00]])
```

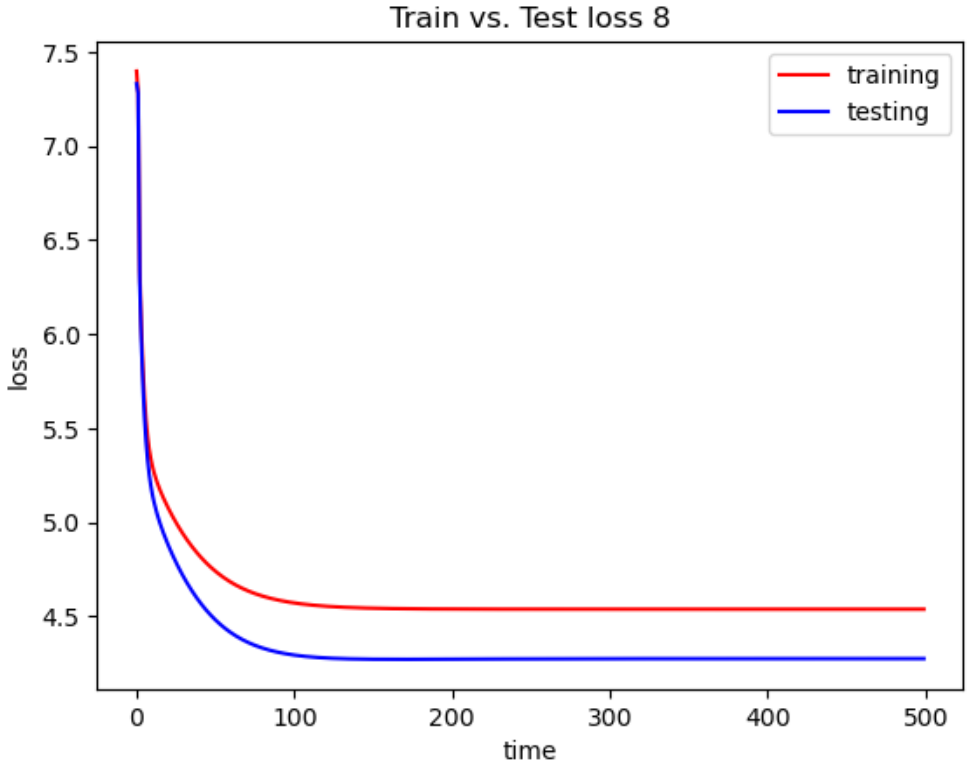


```
Training loss 8:
1 epoch training loss: 1632.8604
2 epoch training loss: 114.4062
3 epoch training loss: 96.6659
4 epoch training loss: 93.9956
5 epoch training loss: 93.5900
6 epoch training loss: 93.5282
7 epoch training loss: 93.5189
8 epoch training loss: 93.5174
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 8:
1 epoch testing loss: 1526.0503
2 epoch testing loss: 88.3224
3 epoch testing loss: 73.2680
4 epoch testing loss: 71.6771
5 epoch testing loss: 71.6968
6 epoch testing loss: 71.8016
7 epoch testing loss: 71.8573
8 epoch testing loss: 71.8813
9 epoch testing loss: 71.8910
10 epoch testing loss: 71.8949
```

Beta：

```
array([[ 4.36504896e+01],
       [-2.65743679e-01],
       [-5.43604071e-03],
       [ 1.26160880e+00]])
```

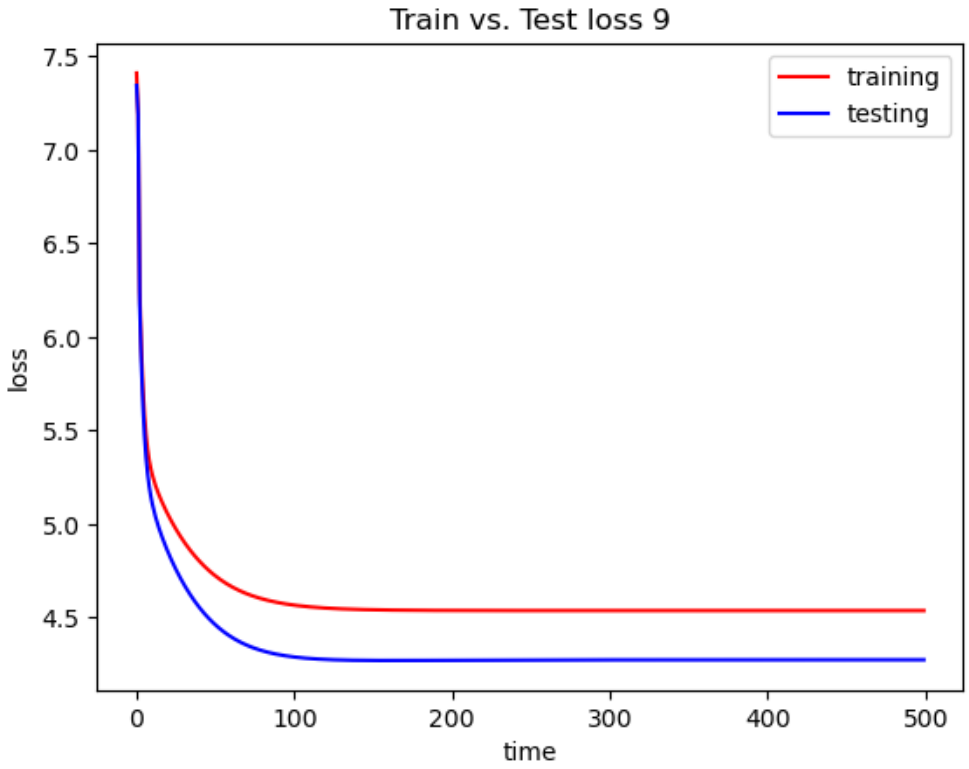


```
Training loss 9:
1 epoch training loss: 1653.8247
2 epoch training loss: 112.5328
3 epoch training loss: 96.2645
4 epoch training loss: 93.9172
5 epoch training loss: 93.5755
6 epoch training loss: 93.5257
7 epoch training loss: 93.5184
8 epoch training loss: 93.5174
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 9:
1 epoch testing loss: 1547.1189
2 epoch testing loss: 86.6375
3 epoch testing loss: 72.9734
4 epoch testing loss: 71.6541
5 epoch testing loss: 71.7092
6 epoch testing loss: 71.8116
7 epoch testing loss: 71.8626
8 epoch testing loss: 71.8838
9 epoch testing loss: 71.8921
10 epoch testing loss: 71.8953
```

Beta：

```
array([[ 4.36519086e+01],
       [-2.65758892e-01],
       [-5.43639438e-03],
       [ 1.26144974e+00]])
```

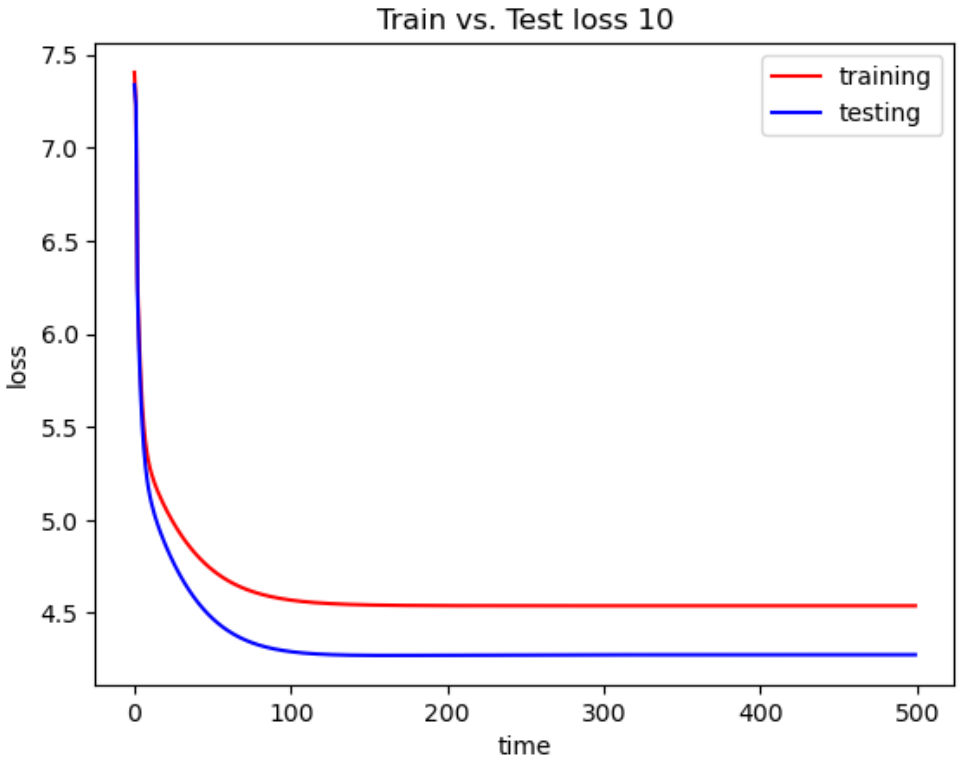


```
Training loss 10:
1 epoch training loss: 1644.1486
2 epoch training loss: 113.2993
3 epoch training loss: 96.4261
4 epoch training loss: 93.9483
5 epoch training loss: 93.5811
6 epoch training loss: 93.5267
7 epoch training loss: 93.5186
8 epoch training loss: 93.5174
9 epoch training loss: 93.5172
10 epoch training loss: 93.5172
```

```
Testing loss 10:
1 epoch testing loss: 1537.4148
2 epoch testing loss: 87.3260
3 epoch testing loss: 73.0911
4 epoch testing loss: 71.6626
5 epoch testing loss: 71.7040
6 epoch testing loss: 71.8076
7 epoch testing loss: 71.8605
8 epoch testing loss: 71.8828
9 epoch testing loss: 71.8917
10 epoch testing loss: 71.8951
```

Beta：

```
array([[ 4.36513539e+01],
       [-2.65752915e-01],
       [-5.43625660e-03],
       [ 1.26151189e+00]])
```



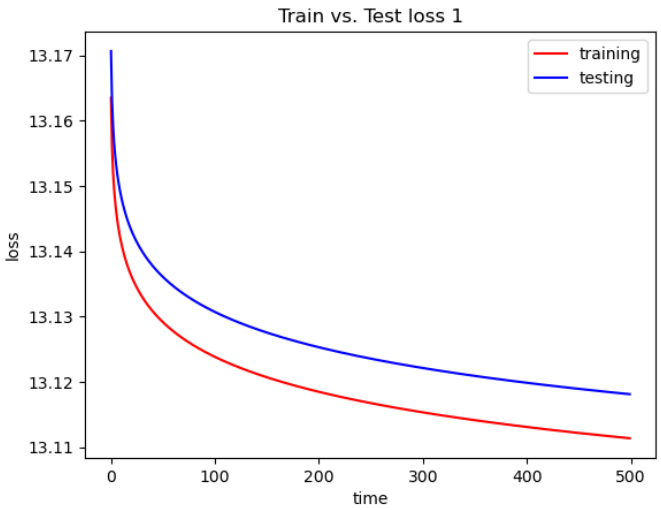
5.

```
Least Square Method

Model 1:
betas: [[ 4.36577429e+01 -2.65825847e-01 -5.43776072e-03  1.26079838e+00]],
Train loss: 93.5172, Test loss: 71.8973, R Square: 0.5218

Model 2:
betas: [[ 4.86562639e+01 -1.12230729e+00  2.15915059e-02 -4.76640083e-03
   1.24039850e+00]],
Train loss: 86.6068, Test loss: 67.9734, R Square: 0.5571

Model 3:
betas: [[ 5.11693952e+01 -2.83577957e-01 -1.40409553e-02  1.63456916e-06
   6.73212945e-01]],
Train loss: 82.9494, Test loss: 66.2943, R Square: 0.5758
```

Disscusion:

1.

根據以上的結果，可以判斷(β0, β1, β2, β3)和(β0', β1', β2', β3')算出來的結果相同。由於題目沒有規定 random 取值(weighting)的範圍，因此我取 0-0.001 較符合我的 regression model。比如當我超出以上的 random 範圍時，train 和 test 的 loss 不一定能夠找到最好的值。如下圖我們可以發現它在經過一段時間便趨近平緩，並未達到最好的結果。由此我理解到 learning rate 的設定和 weighting factor 很有關係，當 weight-ing factor 的初始值距離 minium 很遠的話，learing rate 的初始值也要跟著改變。

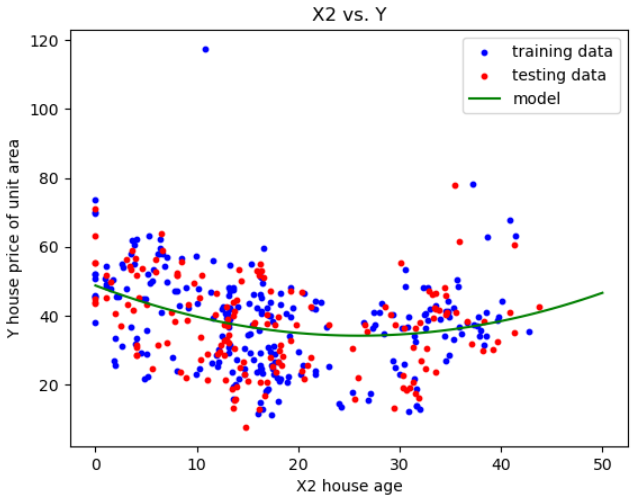


而換句話說，當初使條件設定不良的話，gradient descend 不一定能找到一個最小的值。而 Least Square Method 的目標是最小化預測值和實際值之間差的平方和，可以直接通過解矩陣方程式來找到係數，也就代表可以直接通過計算得到最優解，不需要迭代過程。以上便是兩者的差異。

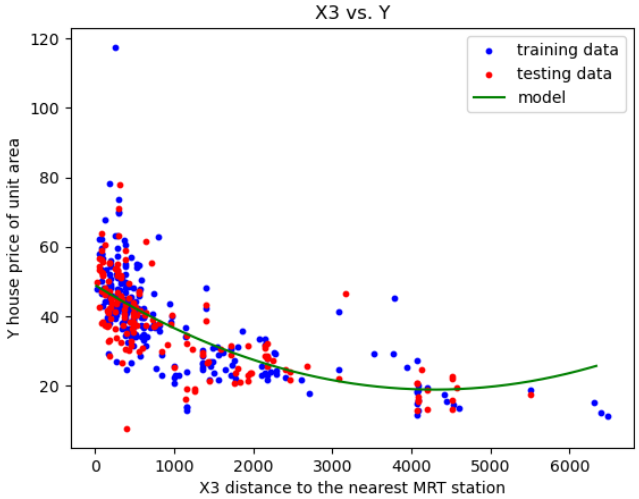
2.

以下三張圖為我所找到的每個 feature 的 best fitting model。首先我在程式中跑了九張圖，也就是每個 feature 對每個 model 的圖，然後透過觀察去判斷。X2、X3 分別選擇含有 2 次項的 model，(X2 為 model 2，X3 為 model 3)，因為我從圖上觀察到它們的分布比較像 2 次曲線。而 X4 選擇 model 1 的原因是根據 model 的公式，由於 model 2 和 model 3 都含有 2 次項，因此 X4 在那個 model 的權重也會因此改變，也就會讓它相比 model 1 來的差

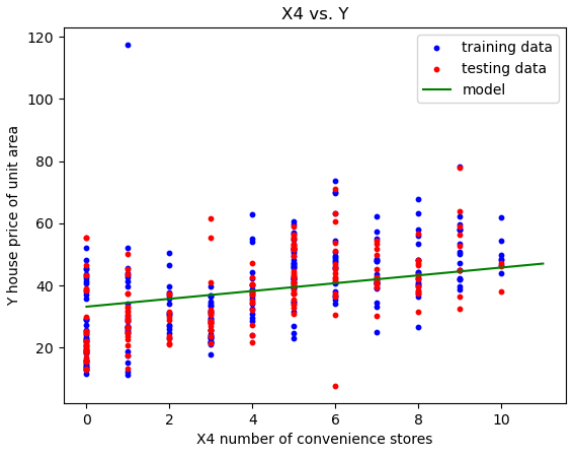
從以下的圖觀察，首先我發現房價和屋齡的關係呈現 U 型，如預想的一樣，屋齡越小的房子價格越高，但比較特別的是在屋齡 30~40 的區段，房價有回升的跡象，我猜測是因為能維持如此久的房子本身用料就不差，所以這個區段的房子本來就屬於高價房。而房價與距離捷運站遠近也有一定的關聯性，可以看到距離捷運越近的房價很高，且有許多房屋集中於接近捷運站的地方，代表交通是房價的一個重要考量之一。而便利商店的數量和房價的關係相較之下比較不是最主要的原因，但也可以觀察到便利商店的數量和房價也是呈正比關係，也就代表便利商店的數量越多也會影響房價，便利商店的數量多寡可能代表都市化的程度，畢竟都市的便利商店數量一定大於鄉村，而都市的房價也大於鄉村。



$$Y = \beta_4 + \beta_5 X_2 + \beta_6 {X_2}^2 + \beta_7 X_3 + \beta_8 X_4$$



$$Y = \beta_9 + \beta_{10} X_2 + \beta_{11} X_3 + \beta_{12} {X_3}^2 + \beta_{13} X_4$$



$$Y = \beta_0' + \beta_1' X_2 + \beta_2' X_3 + \beta_3' X_4$$