HW 3

I

1- $W = (X^T . X)^{-1} X^T . z$ $\Phi^{\dagger} = (\phi^T \phi)^{-1} \Phi^T$

$$\Phi = \begin{bmatrix} 1 & y_{11} y_{12} \\ 1 & y_{21} y_{22} \\ 1 & y_{31} y_{32} \\ 1 & y_{41} y_{42} \\ 1 & y_{51} y_{52} \end{bmatrix} = \begin{bmatrix} 1 & 1\times 1 \\ 1 & 1\times 3 \\ 1 & 3\times 2 \\ 1 & 3\times 3 \\ 1 & 2\times 4 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 3 \\ 1 & 6 \\ 1 & 9 \\ 1 & 8 \end{bmatrix} \Big\} phi$$

$$z = y_{num} = \begin{bmatrix} 1,25 \\ 7,00 \\ 2,70 \\ 3,20 \\ 5,50 \end{bmatrix}$$

$$W = (\Phi^{\dagger} \phi)^{-1} \Phi^T z$$

$$W = \begin{bmatrix} 3,31592 \\ 0,11371 \end{bmatrix} \begin{matrix} \rightarrow w_0 \\ \rightarrow w_1 \end{matrix}$$

2- $phi = \begin{bmatrix} 1 \\ 3 \\ 6 \\ 9 \\ 8 \end{bmatrix}$ $\Phi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ $\lambda = 1$

$$W_{ridge} = (X^T X + \lambda I)^{-1} X^T z \Rightarrow (\Phi^T \Phi + \lambda I)^{-1} \Phi^T z$$

$$= \begin{bmatrix} 1,81809 \\ 0,32376 \end{bmatrix}$$

$$\Phi = \begin{bmatrix} 1 & 1 \\ 1 & 3 \\ 1 & 6 \\ 1 & 9 \\ 1 & 8 \end{bmatrix}$$

λ is the regularization coefficient that controls the balance between the data-dependent error ED(w) and the regularization ofe EW(w). In this case, the ridge regularization weight made the first weight smaller and closer to zero and the second slightly bigger but still is close to zero. Regularization encourages the weights to decay towards zero unless they are supported by the data. If the value of λ is sufficiently large, some coefficients wj will be driven to zero, resulting in a sparse model where certain basis function play no role. This regularization process allows complex models to be trained on small datasets without severe overfitting, effectively limiting the models complexity.

3-

D	y_1	y_2	y_{num}	y_{class}
x_1	1	1	1.25	B
x_2	1	3	7.0	A
x_3	3	2	2.7	C
x_4	3	3	3.2	A
x_5	2	4	5.5	B
x_6	2	2	0.7	
x_7	1	2	1.1	
x_8	5	1	2.2	

$$X_{new} = \begin{bmatrix} 2 & 2 \\ 1 & 2 \\ 5 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & \Phi_0 \\ 1 & \Phi_1 \\ 1 & \Phi_2 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$

$$y\_out\_new = [0,7 \quad 1.1 \quad 2.2]$$

OLS

$$\hat{z}_{train} = \Phi w = \begin{bmatrix} 1 & 1 \\ 1 & 3 \\ 1 & 6 \\ 1 & 9 \\ 1 & 8 \end{bmatrix} \begin{bmatrix} 3,31592 \\ 0,11372 \end{bmatrix} = \begin{bmatrix} 3.42965 \\ 3.65708 \\ 3.99823 \\ 4.33938 \\ 4.22566 \end{bmatrix}$$

$$\hat{z}_{test} = \Phi_{new} w = \begin{bmatrix} 1 & 2\times 2 \\ 1 & 1\times 2 \\ 1 & 5\times 1 \end{bmatrix} \begin{bmatrix} 3,31592 \\ 0,11372 \end{bmatrix} = \begin{bmatrix} 3.77079 \\ 3.54336 \\ 3.88451 \end{bmatrix}$$

$$RMSE(\hat{z}, z) = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (z_i - \hat{z}_i)^2}$$

$$RMSE_{Test} = \sqrt{\frac{1}{3} \sum_{i=1}^{3} (z_{test} - \hat{z}_{test})^2}$$

$$= \sqrt{\frac{1}{3} (0.7 - 3,77079)^2 + (1.1 - 3.54336)^2 + (2.2 - 3.88451)^2}$$

$$= 2.46559$$

$$RMSE_{Train} = \sqrt{\frac{1}{5} \sum_{i=1}^{5} (z_{train} - \hat{z}_{train})^2}$$

$$= \sqrt{\frac{1}{5} (1.25 - 342965)^2 + (7.0 - 3.65708)^2 + (2.7 - 3.99823)^2 + (3.2 - 4.33938)^2 + (5.5 - 4.22566)^2}$$

$$= 2.026499$$

Ridge

$$\hat{z}_{train} = \Phi W_{ridge} = \begin{bmatrix} 1 & 1 \\ 1 & 3 \\ 1 & 6 \\ 1 & 9 \\ 1 & 8 \end{bmatrix} \begin{bmatrix} 1.81809 \\ 0.32376 \end{bmatrix} = \begin{bmatrix} 2.14184 \\ 2.78936 \\ 3.76064 \\ 4.73191 \\ 4.40816 \end{bmatrix}$$

$$\hat{z}_{test} = \Phi_{new} W = \begin{bmatrix} 1 & 4 \\ 1 & 2 \\ 1 & 5 \end{bmatrix} \begin{bmatrix} 1.81809 \\ 0.32376 \end{bmatrix} = \begin{bmatrix} 3.11312 \\ 2.46560 \\ 3.43688 \end{bmatrix}$$

$$RMSE_{Test} = \sqrt{\frac{1}{3} \sum_{i=1}^{3} (z_{test} - \hat{z}_{test})^2}$$

$$= \sqrt{\frac{1}{3} (0.7 - 3.11312)^2 + (1.1 - 2.46560)^2 + (2.2 - 3.43688)^2}$$

$$= 1.75289$$

$$RMSE_{Train} = \sqrt{\frac{1}{5} \sum_{i=1}^{5} (z_{train} - \hat{z}_{train})^2}$$

$$= \sqrt{\frac{1}{5} (1.25 - 2.14184)^2 + (7.0 - 2.78936)^2 + (2.7 - 3.76064)^2 + (3.2 - 4.73191)^2 + (5.5 - 4.40816)^2}$$

$$= 2.15354$$

As expected, Ridge Regression demonstrates better performance on the test data compared to Ordinary Least Squares (OLS). On the training data, OLS has a slightly lower RMSE, suggesting that it fits the training data more closely. This may indicate that OLS is overfitting the training data, capturing noise rather than the underlying trend. Ridge Regression, on the other hand, effectively balances bias and variance, resulting in better generalization to unseen data.

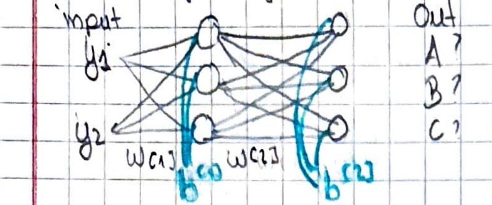
4- $x_1 = (1, 1)$

ac function: $softmax(z_c^{[out]}) = \frac{e^{z_c^{[out]}}}{\sum_{l=1}^{|c|} e^{z_l^{[out]}}}$

$\eta = 0.1$

Entropy loss: $E(w) = -\sum_{i=1}^{n} \sum_{l=1}^{|c|} t^{(i)} \log(x_l^{[out](i)})$

Foward propagation



$$X^{[0]} = X = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$X^{[1]} = W^{[1]} X^{[0]} + b^{[1]} = \begin{bmatrix} 0.1 & 0.1 \\ 0.1 & 0.2 \\ 0.2 & 0.1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0 \\ 0.1 \end{bmatrix} = \begin{bmatrix} 0.3 \\ 0.3 \\ 0.4 \end{bmatrix}$$

$$X^{[2]} = W^{[2]} X^{[1]} + b^{[2]} = \text{softmax}\left(\begin{bmatrix} 1 & 2 & 2 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0.3 \\ 0.3 \\ 0.3 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right)$$

$$= \text{softmax}\left(\begin{bmatrix} 2.7 \\ 2.3 \\ 2 \end{bmatrix} \right) = \begin{bmatrix} 0.46149 \\ 0.30934 \\ 0.22917 \end{bmatrix}$$

Backward propagation (update)

Target output

x_0 in B

$$\Rightarrow t = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

$$W^{[2]}_{new} = W^{[2]} - \eta \frac{\partial E^{[2]}}{\partial W^{[2]}} = \begin{bmatrix} 1 & 2 & 2 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} - 0.1 \cdot \delta^{[2]} \; X^{[1]^T}$$

$$= \begin{bmatrix} 1 & 2 & 2 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} - 0.1 \cdot \begin{bmatrix} O_1 - t_1 \\ O_2 - t_2 \\ O_3 - t_3 \end{bmatrix} \cdot \begin{bmatrix} 0.3 & 0.3 & 0.4 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 2 & 2 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} - 0.1 \cdot \underbrace{\begin{bmatrix} 0.46149 \\ -0.69066 \\ 0.22917 \end{bmatrix}}_{\delta^{[2]}} \cdot \begin{bmatrix} 0.3 & 0.3 & 0.4 \end{bmatrix}$$

$$W^{[2]}_{new} = \begin{bmatrix} 0.98616 & 1.98615 & 1.98154 \\ 1.02072 & 2.02072 & 1.02763 \\ 0.99313 & 0.99313 & 0.99083 \end{bmatrix}$$

$$W^{[1]}_{new} = W^{[1]} - \eta \frac{\partial E^{[1]}}{\partial W^{[1]}} = W^{[1]} - \eta \, \delta^{[1]} X^T = W^{[1]} - \eta \underbrace{\left[W^{[2]^T} \delta^{[2]} \circ \phi'^{[1]} (z^{[1]}) \right] X^T}_{\delta 1}$$

$$W^{[1]} - \eta \left[W^{[2]^T} \cdot \delta^{[2]} \right] X^T = \begin{bmatrix} 0.1 & 0.1 \\ 0.1 & 0.2 \\ 0.2 & 0.1 \end{bmatrix} - 0.1 \cdot \left(\begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 1 \\ 2 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0.46149 \\ -0.69066 \\ 0.22917 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right)$$

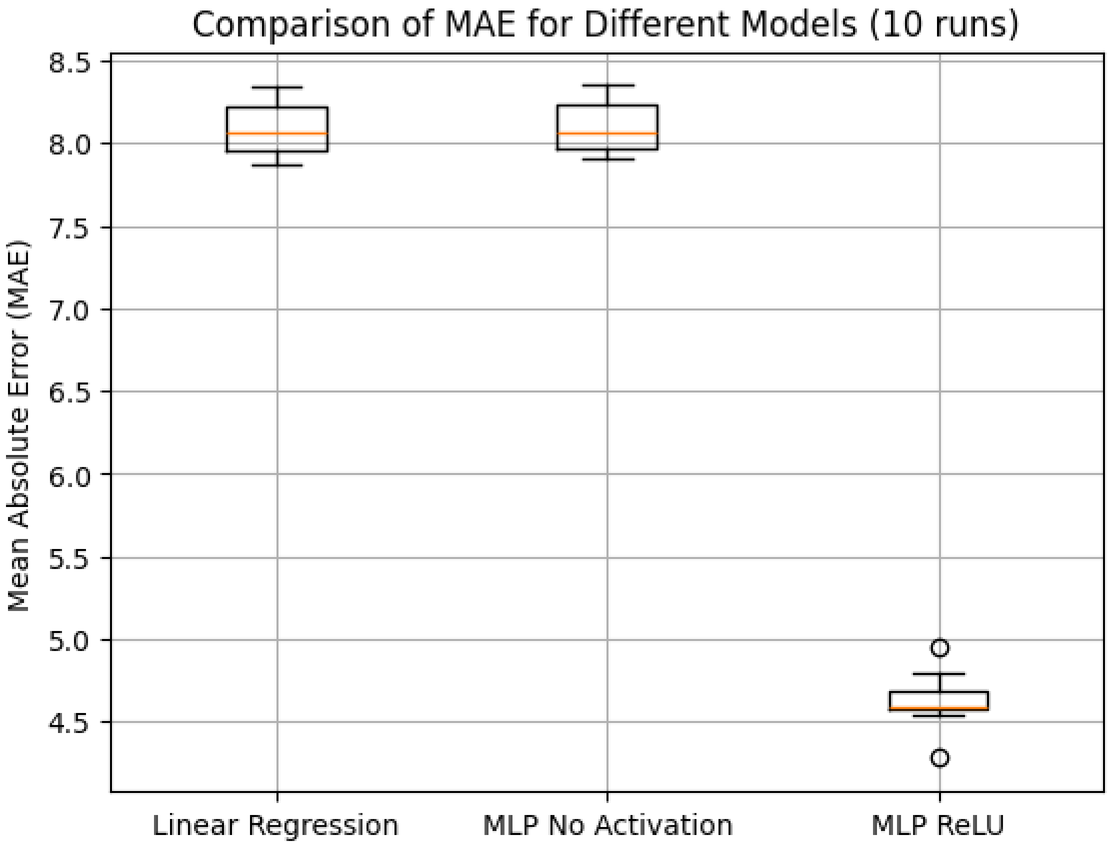
$$\delta^{[1]} = \begin{bmatrix} 0 \\ -0.22917 \\ 0.46149 \end{bmatrix} \qquad W^{[1]}_{new} = \begin{bmatrix} 0.1 & 0.1 \\ 0.32916 & 0.42917 \\ -0.26149 & -0.36149 \end{bmatrix}$$

$$b^{[2]}_{new} = b^{[2]} - \eta \frac{\partial E^{[2]}}{\partial b^{[2]}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - 0.1 \begin{bmatrix} 0.46149 \\ -0.69066 \\ 0.22917 \end{bmatrix} = \begin{bmatrix} 0.95381 \\ 1.06907 \\ 0.97708 \end{bmatrix} \Rightarrow b^{[2]}_{new}$$

$$b^{[1]}_{new} = b^{[1]} - \eta \frac{\partial E^{[1]}}{\partial b^{[1]}} = \begin{bmatrix} 0.1 \\ 0 \\ 0.1 \end{bmatrix} - 0.1 \begin{bmatrix} 0 \\ -0.22917 \\ 0.46149 \end{bmatrix} = \begin{bmatrix} -0.36149 \\ 0.69066 \\ -0.12917 \end{bmatrix} \Rightarrow b^{[1]}_{new}$$

Parte II

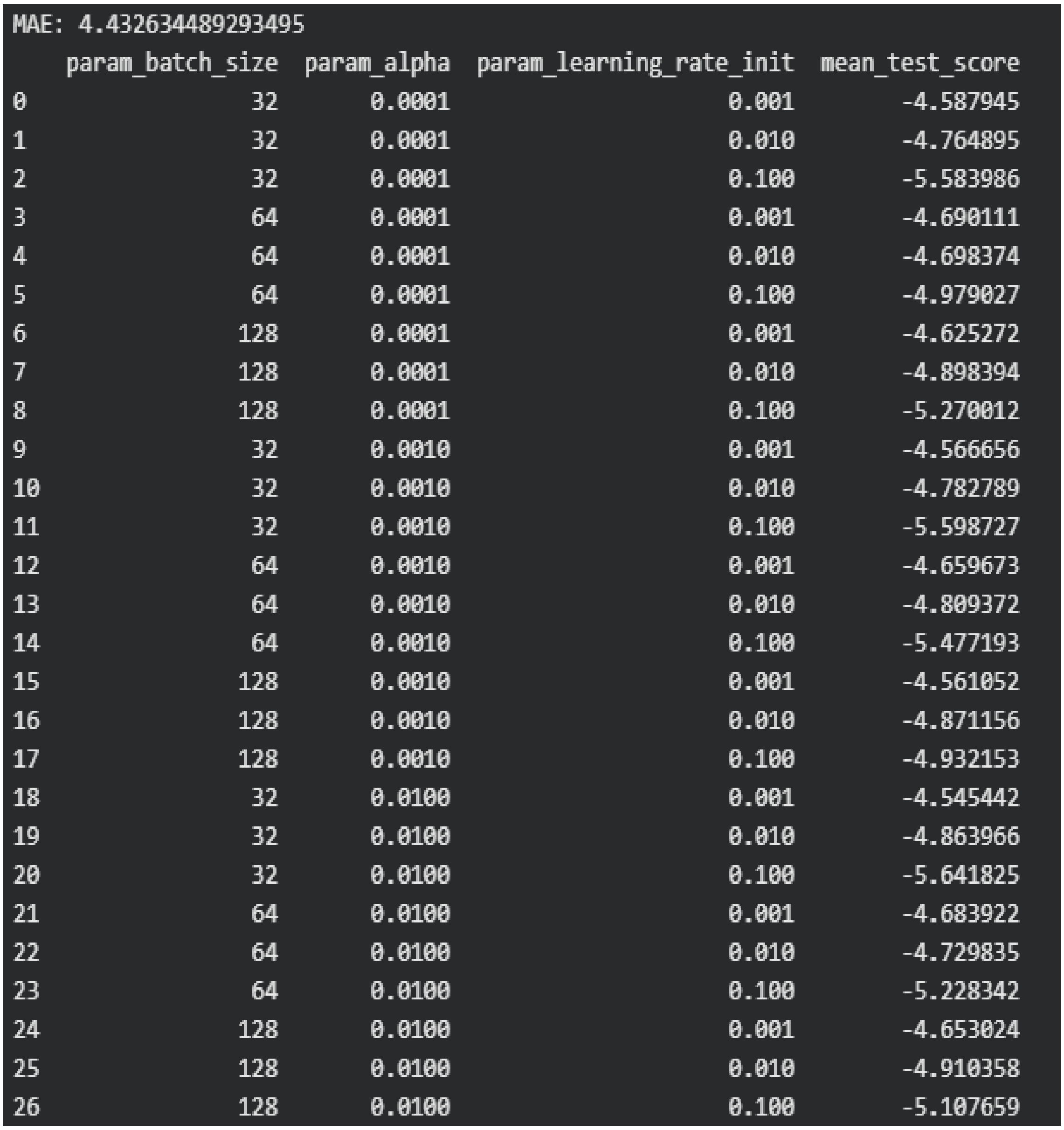
5.



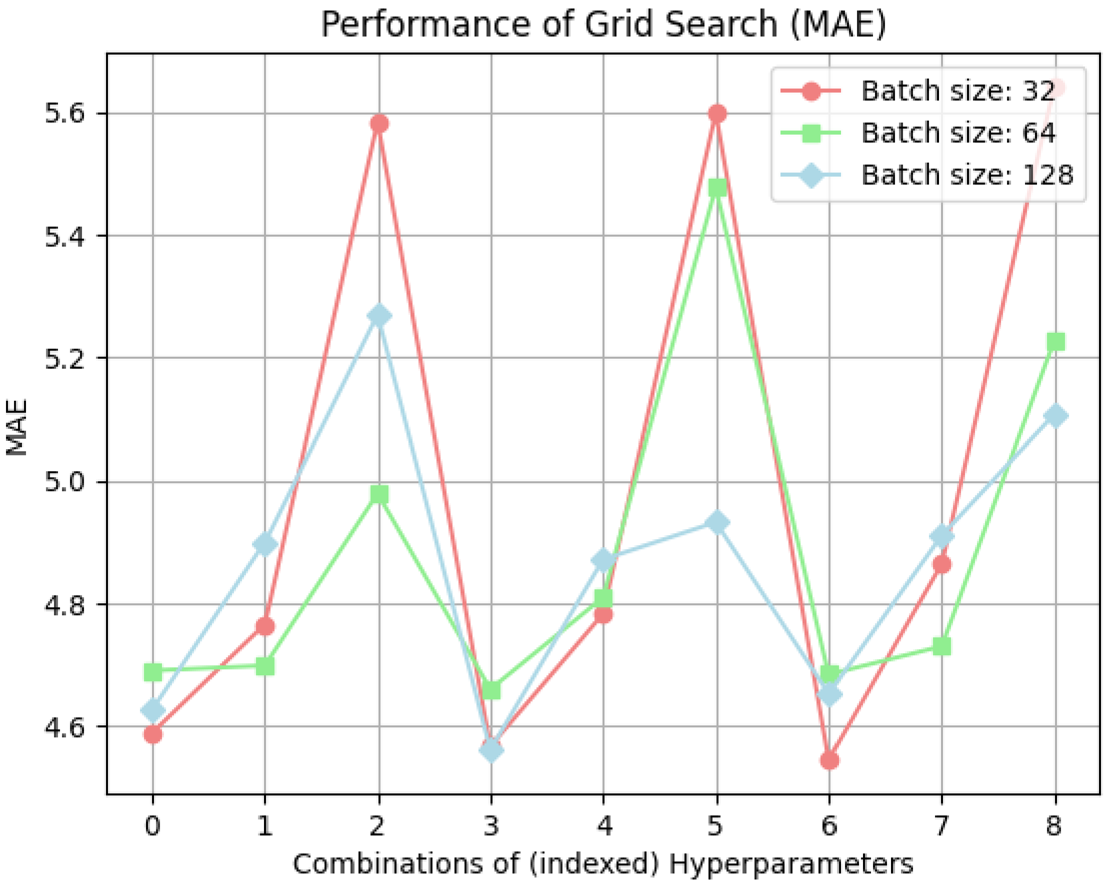
6.

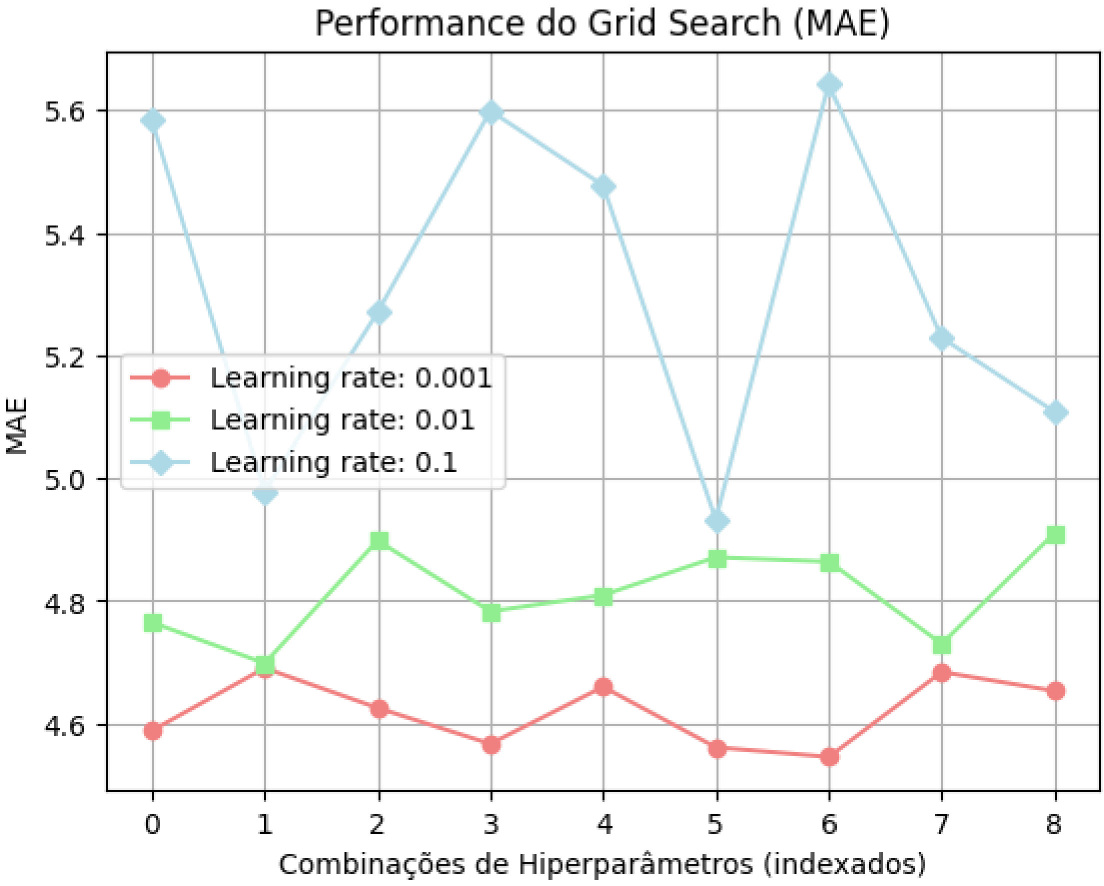
The linear regression model and the MLP (multi-layer perceptron) without an activation function produce nearly identical performance. This is expected because an MLP without activation functions effectively reduces to a series of linear regressions. However, the MLP with a ReLU activation function performs significantly better, highlighting the critical role activation functions play in enhancing the model's predictive capabilities.

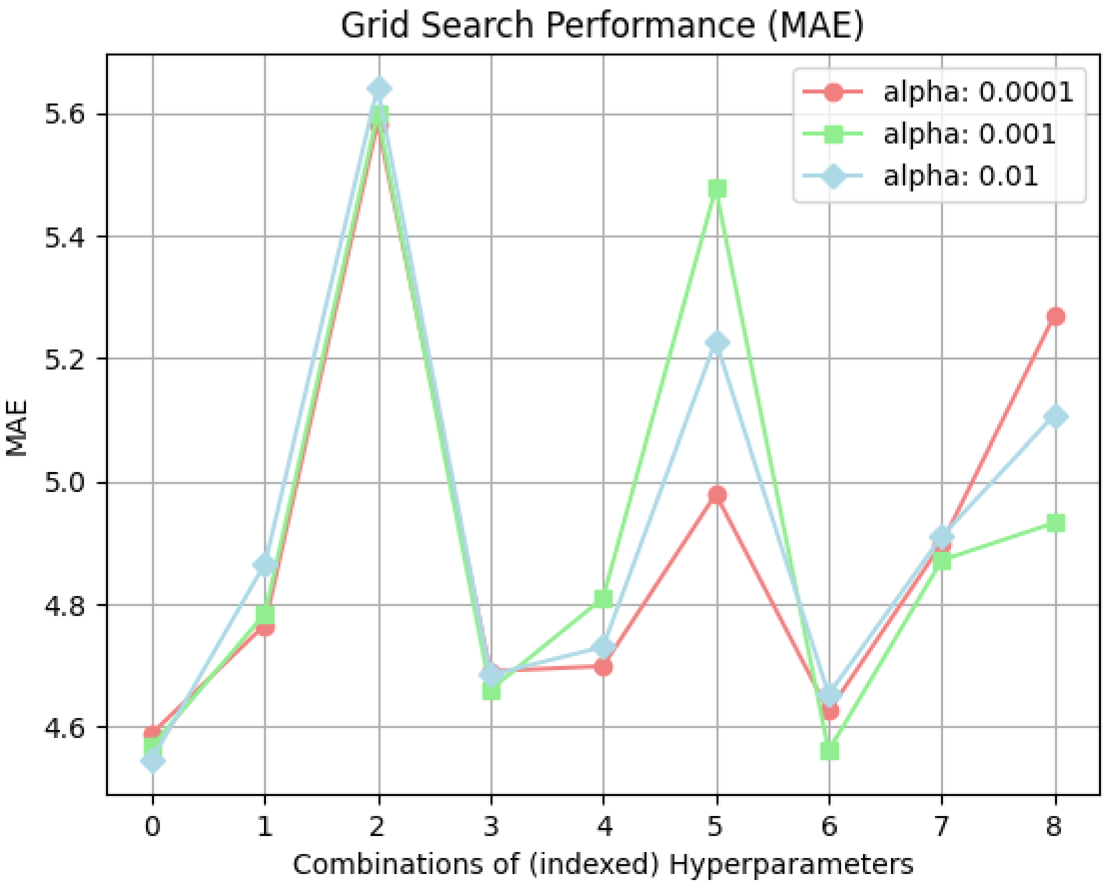
7.



```
MAE: 4.432634489293495
    param_batch_size  param_alpha  param_learning_rate_init  mean_test_score
0                 32       0.0001                     0.001        -4.587945
1                 32       0.0001                     0.010        -4.764895
2                 32       0.0001                     0.100        -5.583986
3                 64       0.0001                     0.001        -4.690111
4                 64       0.0001                     0.010        -4.698374
5                 64       0.0001                     0.100        -4.979027
6                128       0.0001                     0.001        -4.625272
7                128       0.0001                     0.010        -4.898394
8                128       0.0001                     0.100        -5.270012
9                 32       0.0010                     0.001        -4.566656
10                32       0.0010                     0.010        -4.782789
11                32       0.0010                     0.100        -5.598727
12                64       0.0010                     0.001        -4.659673
13                64       0.0010                     0.010        -4.809372
14                64       0.0010                     0.100        -5.477193
15               128       0.0010                     0.001        -4.561052
16               128       0.0010                     0.010        -4.871156
17               128       0.0010                     0.100        -4.932153
18                32       0.0100                     0.001        -4.545442
19                32       0.0100                     0.010        -4.863966
20                32       0.0100                     0.100        -5.641825
21                64       0.0100                     0.001        -4.683922
22                64       0.0100                     0.010        -4.729835
23                64       0.0100                     0.100        -5.228342
24               128       0.0100                     0.001        -4.653024
25               128       0.0100                     0.010        -4.910358
26               128       0.0100                     0.100        -5.107659
```


Performance of Grid Search (MAE)
MAE
5.6
5.4
5.2
5.0
4.8
4.6
0
1
2
3
4
5
6
7
8
Combinations of (indexed) Hyperparameters
Batch size: 32
Batch size: 64
Batch size: 128


Performance do Grid Search (MAE)
MAE
5.6
5.4
5.2
5.0
4.8
4.6
0
1
2
3
4
5
6
7
8
Combinações de Hiperparâmetros (indexados)
Learning rate: 0.001
Learning rate: 0.01
Learning rate: 0.1



The optimal hyperparameters found during model tuning were a batch size of 32, a learning rate of 0.001, and an alpha value of 0.001. From the learning rate graph, it's evident that the best performance was achieved with the smallest learning rate of 0.001. This is because the maximum number of iterations was increased, allowing the model more time to converge. In contrast, a larger learning rate of 0.1 converges too quickly, overshooting the local minimum and resulting in higher errors. Although the smaller learning rate of 0.001 converges more slowly, increasing the number of iterations allows the model to reach closer to the desired local minimum in consequence having a better performance On the batch size graph, it can be observed that the batch size of 32 consistently results in the smallest error. However, there are specific combinations of parameters that occasionally cause the error to spike, surpassing those of other batch sizes. Despite these fluctuations, batch size 32 generally outperforms the others.