TÉCNICO LISBOA

Aprendizagem 2021/22
Homework I – Group 081

I. Pen-and-paper

Consider the problem of learning a regression model from 5 univariate observations $((0.8), (1), (1.2), (1.4), (1.6))$ with targets $(24, 20, 10, 13, 12)$.

1) Exercicio 1

1) [5v] Consider the basis function, $\phi_j(x) = x^j$, for performing a 3-order polynomial regression,

$$\hat{z}(x, \mathbf{w}) = \sum_{j=0}^{3} w_j \phi_j(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 \,.$$

Learn the Ridge regression (l_2 regularization) on the transformed data space using the closed form solution with $\lambda = 2$.

Hint: use numpy matrix operations (e.g., `linalg.pinv` for inverse) to validate your calculus.

Solução:

Primeiro precisamos de transformar o nosso data set para que fique um polinomio de grau 3.

Data set original:

X = [0.8, 1, 1.2, 1.4, 1.6]

Data set transformado:

	bias	y1	y2	y3
x1	1.0	0.8	0.64	0.512
x2	1.0	1.0	1.0	1.0
x3	1.0	1.2	1.44	1.728
x4	1.0	1.4	1.96	2.744
x5	1.0	1.6	2.56	4.096

X = Tabela

Target:

Z = [24, 20, 10, 13, 12]

'Indicações das aulas' [@06_LinearBayesianRegression]

$$\nabla E(\mathbf{w}) = \nabla\left(\frac{1}{2}\cdot(\mathbf{z} - X\cdot\mathbf{w})^T(\mathbf{z} - X\cdot\mathbf{w}) + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}\right) = 0$$

$$-2X^T\mathbf{z} + 2X^T\cdot X\cdot\mathbf{w} + 2\lambda\cdot\mathbf{w} = 0$$

$$X^T\mathbf{z} = (X^T\cdot X + \lambda\cdot I)\cdot\mathbf{w}$$

$$(X^T\cdot X + \lambda\cdot I)^{-1}\cdot X^T\cdot\mathbf{z} = \mathbf{w}$$

TÉCNICO LISBOA

Aprendizagem 2021/22

Homework I – Group 081

Seguindo o raciocinio da aula, reparamos que já temos todos os parametros necessários para chegar ao objetivo pedido

$$\left(X^T \cdot X + \lambda \cdot I\right)^{-1} \cdot X^T \cdot \mathbf{z} = \mathbf{w}$$

O nosso X é a tabela

I é a matriz identidade

Z é o vetor target

Lambda = 2 por definição

Agora basta apenas utilizar os nossos conhecimentos de algebra linear para chegar ao vetor w

1. Calcular transposta de X

$$\begin{pmatrix} 1 & 0{,}8 & 0{,}64 & 0{,}512 \\ 1 & 1 & 1 & 1 \\ 1 & 1{,}2 & 1{,}44 & 1{,}728 \\ 1 & 1{,}4 & 1{,}96 & 2{,}744 \\ 1 & 1{,}6 & 2{,}56 & 4{,}096 \end{pmatrix}^T = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{4}{5} & 1 & \frac{6}{5} & \frac{7}{5} & \frac{8}{5} \\ \frac{16}{25} & 1 & \frac{36}{25} & \frac{49}{25} & \frac{64}{25} \\ \frac{64}{125} & 1 & \frac{216}{125} & \frac{343}{125} & \frac{512}{125} \end{pmatrix}$$

2. Multiplicar Xt por X

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{4}{5} & 1 & \frac{6}{5} & \frac{7}{5} & \frac{8}{5} \\ \frac{16}{25} & 1 & \frac{36}{25} & \frac{49}{25} & \frac{64}{25} \\ \frac{64}{125} & 1 & \frac{216}{125} & \frac{343}{125} & \frac{512}{125} \end{pmatrix} \cdot \begin{pmatrix} 1 & 0{,}8 & 0{,}64 & 0{,}512 \\ 1 & 1 & 1 & 1 \\ 1 & 1{,}2 & 1{,}44 & 1{,}728 \\ 1 & 1{,}4 & 1{,}96 & 2{,}744 \\ 1 & 1{,}6 & 2{,}56 & 4{,}096 \end{pmatrix} = \begin{pmatrix} 5 & 6 & \frac{38}{5} & \frac{252}{25} \\ 6 & \frac{38}{5} & \frac{252}{25} & \frac{8674}{625} \\ \frac{38}{5} & \frac{252}{25} & \frac{8674}{625} & \frac{492}{25} \\ \frac{252}{25} & \frac{8674}{625} & \frac{492}{25} & \frac{89234}{3125} \end{pmatrix}$$

3. Identidade * 2

$$\begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix}$$

4. Resultado do passo dois somado com passo Identidade*2

$$\begin{pmatrix} 5 & 6 & \frac{38}{5} & \frac{252}{25} \\ 6 & \frac{38}{5} & \frac{252}{25} & \frac{8674}{625} \\ \frac{38}{5} & \frac{252}{25} & \frac{8674}{625} & \frac{492}{25} \\ \frac{252}{25} & \frac{8674}{625} & \frac{492}{25} & \frac{89234}{3125} \end{pmatrix} + \begin{pmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 7 & 6 & \frac{38}{5} & \frac{252}{25} \\ 6 & \frac{48}{5} & \frac{252}{25} & \frac{8674}{625} \\ \frac{38}{5} & \frac{252}{25} & \frac{9924}{625} & \frac{492}{25} \\ \frac{252}{25} & \frac{8674}{625} & \frac{492}{25} & \frac{95484}{3125} \end{pmatrix}$$

5. Inversa do resultado do passo 4.

$$\begin{pmatrix} 7 & 6 & \frac{38}{5} & \frac{252}{25} \\ 6 & \frac{48}{5} & \frac{252}{25} & \frac{8674}{625} \\ \frac{38}{5} & \frac{252}{25} & \frac{9924}{625} & \frac{492}{25} \\ \frac{252}{25} & \frac{8674}{625} & \frac{492}{25} & \frac{95484}{3125} \end{pmatrix}^{(-1)} = \begin{pmatrix} \frac{4314983511}{12628448902} & \frac{-1533420825}{12628448902} & \frac{-1891799875}{25256897804} & \frac{-58882500}{6314224451} \\ \frac{-1533420825}{12628448902} & \frac{4915090875}{12628448902} & \frac{-2441765625}{25256897804} & \frac{-940266875}{12628448902} \\ \frac{-1891799875}{25256897804} & \frac{-2441765625}{25256897804} & \frac{18820323125}{50513795608} & \frac{-2163890625}{12628448902} \\ \frac{-58882500}{6314224451} & \frac{-940266875}{12628448902} & \frac{-2163890625}{12628448902} & \frac{1136484375}{6314224451} \end{pmatrix}$$

6. Multiplicar resultado de 5 por Xt

$$\begin{pmatrix} \frac{4314983511}{12628448902} & \frac{-1533420825}{12628448902} & \frac{-1891799875}{25256897804} & \frac{-58882500}{6314224451} \\ \frac{-1533420825}{12628448902} & \frac{4915090875}{12628448902} & \frac{-2441765625}{25256897804} & \frac{-940266875}{12628448902} \\ \frac{-1891799875}{25256897804} & \frac{-2441765625}{25256897804} & \frac{18820323125}{50513795608} & \frac{-2163890625}{12628448902} \\ \frac{-58882500}{6314224451} & \frac{-940266875}{12628448902} & \frac{-2163890625}{12628448902} & \frac{1136484375}{6314224451} \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ \frac{4}{5} & 1 & \frac{6}{5} & \frac{7}{5} & \frac{8}{5} \\ \frac{16}{25} & 1 & \frac{36}{25} & \frac{49}{25} & \frac{64}{25} \\ \frac{64}{125} & 1 & \frac{216}{125} & \frac{343}{125} & \frac{512}{125} \end{pmatrix} = \begin{pmatrix} \frac{2422575211}{12628448902} & \frac{3435795497}{25256897804} & \frac{909284691}{12628448902} & \frac{-17833363}{25256897804} & \frac{-1042359089}{12628448902} \\ \frac{1135870235}{12628448902} & \frac{2441040725}{25256897804} & \frac{981835815}{12628448902} & \frac{749367565}{25256897804} & \frac{-646068545}{12628448902} \\ \frac{-38532975}{25256897804} & \frac{1497629625}{50513795608} & \frac{1250308025}{25256897804} & \frac{2516426325}{50513795608} & \frac{564796725}{25256897804} \\ \frac{-545554250}{6314224451} & \frac{-474476875}{6314224451} & \frac{-217198875}{6314224451} & \frac{280831000}{6314224451} & \frac{1074164000}{6314224451} \end{pmatrix}$$

7. Multiplicar resultado de 6 por Z

$$\begin{pmatrix} \frac{2422575211}{12628448902} & \frac{3435795497}{25256897804} & \frac{909284691}{12628448902} & \frac{-17833363}{25256897804} & \frac{-1042359089}{12628448902} \\ \frac{1135870235}{12628448902} & \frac{2441040725}{25256897804} & \frac{981835815}{12628448902} & \frac{749367565}{25256897804} & \frac{-646068545}{12628448902} \\ \frac{-38532975}{25256897804} & \frac{1497629625}{50513795608} & \frac{1250308025}{25256897804} & \frac{2516426325}{50513795608} & \frac{564796725}{25256897804} \\ \frac{-545554250}{6314224451} & \frac{-474476875}{6314224451} & \frac{-217198875}{6314224451} & \frac{280831000}{6314224451} & \frac{1074164000}{6314224451} \end{pmatrix} \cdot \begin{pmatrix} 24 \\ 20 \\ 10 \\ 13 \\ 12 \end{pmatrix} = \begin{pmatrix} \frac{177936762033}{25256897804} \\ \frac{117215435345}{25256897804} \\ \frac{99377833825}{50513795608} \\ \frac{-8214057250}{6314224451} \end{pmatrix}$$

O resultado de w está em (7), que feitas as contas será aproximadamente

W ~= [7.045, 4.64, 1.97, -1.3]

2) Exercicio 2

2) [1v] Compute the training RMSE for the learnt regression model.

Solução:

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

Aprendizagem 2021/22

Homework I – Group 081

Temos a formula do RMSE e temos todos os parametros para a calcular:

1) Computar previsão do Z com os novos pesos:

$$\hat{z}(x, \mathbf{w}) = \sum_{j=0}^{3} w_j \phi_j(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 .$$

W ~= [7.045, 4.64, 1.97, -1.3]

x = 0.8:

7.45 + 0.8*4.64 + 0.64*1.97 – 0.512*1.3

Z-prev = 11.7572

(Repetir o mesmo para todos os X's)

x = 1:

7.45 + 1*4.64 + 1*1.97 – 1*1.3

Z-prev = 12.76

x = 1.2:

7.45 + 1.2*4.64 + 1.44*1.97 – 1.728*1.3

Z-prev = 13.6084

x = 1.4:

7.45 + 1.4*4.64 + 1.96*1.97 – 2.744*1.3

Z-prev =14.24

x = 1.6:

7.45 + 1.6*4.64 + 2.56*1.97 – 4.096*1.3

Z-prev = 14.5924

Ou seja:

Z = [24,20,10,13,12]

Z.prev = [11.7572, 12.76, 13.60, 14.24, 14.5925]

Aplicando a formula do RMSE ($n = 5$)

Ficamos com:

$$\text{RMSE} = \text{sqrt}(1/5 * ((z1 - z.\text{prev}1)^2 + \ldots + z5 - z.\text{prev}5)^2))$$

$$\text{RMSE} = [\ldots] = 6.687$$

3) Exercicio 3

3) [6v] Consider a multi-layer perceptron characterized by one hidden layer with 2 nodes. Using the activation function $f(x) = e^{0.1x}$ on all units, all weights initialized as 1 (including biases), and the half squared error loss, perform one batch gradient descent update (with learning rate $\eta = 0.1$) for the first three observations (0.8), (1) and (1.2).

Solução:

1) Analizar formato da rede:

1 – 2 – 1

(desconsiderando as transformações aplicadas no exercicio 1)

1ª camada (input): 1 node pois casa observação é composta apenas por uma feature (y1)

2ª camada (hidden): 2 node para seguir os requisitos do exercicio

3ª camada (output): 1 node pois as targets também são compostas apenas por uma dimensão

2) Considerações do enunciado:

- Função de ativação é $e^{0.1x}$ para todas as unidades
- TODOS os pesos e bias são inicializados a 1
- Loss function: half squared error
- Learning rate $= 0.1$
- Conseguimos prever as dimensões das matrizes dos pesos e das bias olhando para a estrutura da rede:
- $w^{[1]} = \begin{bmatrix}1\\1\end{bmatrix}$ $w^{[2]} = \begin{bmatrix}1 & 1\end{bmatrix}$ $b^{[1]} = \begin{bmatrix}1\\1\end{bmatrix}$ $b^{[2]} = [1]$

- Pipeline de ativação (esquema rápido):

a0 – NET 1 – a1 – NET2 – a2

TÉCNICO LISBOA

Aprendizagem 2021/22

Homework I – Group 081

3) Derivada da função loss: $\frac{\partial l}{\partial a^{[2]}} = \left(a^{[2]} - z\right)$

4) Fazer o forwarding:

a0 = x (input)

Net1 = $[1\ 1]^T * x + [1\ 1]^T = [x \quad x]^T + [1 \quad 1]^T = [1 + x \quad 1 + x]^T$

a1 = $\left[e^{0.1(1+x)}\ e^{1*(1+x)}\right]^T$

Net2 = $[1 \quad 1]$ * a1 + $[1]$ = $1 + 2 * e^{0.1\,(1+x)}$

a2 = $e^{0.1*Net2}$

Aplicar forwarding para as primeiras 3 observações:

Se x = 0.8 => a2 = 1.40417

Se x = 1 => a2 = 1.41097

Se x = 2 => a2 = 1.41795

5)Calcular auxiliares de raciocinio (deltas):

1: $\frac{\partial l}{\partial a^{[2]}}$ (feito anteriormente) = $\left(a^{[2]} - z\right)$

2: $\frac{\partial a^{[2]}}{\partial NET^{[2]}} = 0.1 * a^{[2]}$

3: $\frac{\partial NET^{[2]}}{\partial a^{[1]}} = w^{[2]\ T}$

4: $\frac{\partial a^{[1]}}{\partial NET^{[1]}} = \begin{bmatrix} 0.1a^{[1]} & 0 \\ 0 & 0.1a^{[1]} \end{bmatrix}$

Delta2 = $(0.1 * a^{[2]})*(a^{[2]} - z)$

Delta1 $= \begin{bmatrix} 0.1a^{[1]} & 0 \\ 0 & 0.1a^{[1]} \end{bmatrix} * w^{[2]\ T} * delta2$

Atualizar b2:

$$\frac{\partial l}{\partial b^{[2]}} = Delta2$$

Calculando para

Loss.x1 = (0.1 * 1.40417) * (1.40417 – 24) = -3.1728

(repetir para todos os x's)

Loss.x2 = -2.6229
Loss.x3 = -1.2169

Loss.X = {soma de todas as Loss's} = -7.0126

Atualizar b2:

b2[new] = b2[old] - learningRate * Loss.X

= 1 – 0.1*(-7.0126) = 1.70126

Atualizar b1:

$$\frac{\partial l}{\partial b^{[1]}} = Delta1$$

Calculando para

Loss.x1 = $\begin{bmatrix} 0.1e^{0.18} & 0 \\ 0 & 0.1e^{0.18} \end{bmatrix} * [1\ 1]^T * delta2 = [-0.3799 - 0.3799]^T$

(repetir para todos os x's)

Loss.x2 = $[-0.3204 \quad -0.3204]^T$
Loss.x3 = $[-0.1516 \quad -0.1516]^T$

Loss.X = $[-0.8519 \quad -0.8519]^T$

Atualizar b1:

b1[new] = b1[old] - learningRate * Loss.X

= $[1 \;\; 1]^T$ – 0.1*$[-0.8519 \quad -0.8519]^T$ = $[1.08519 \quad 1.08519]^T$

Atualizar w2:

$$\frac{\partial l}{\partial w^{[2]}} = Delta2 * {a^{[1]}}^T$$

Calculando para

Loss.x1 = [-3.7985 -3.7985]
Loss.x2 = [-3.2036 -3.2036]
Loss.x3 = [-1.5163 -1.5163]

Loss.X = {soma de todas as Loss's} = [-8.5184 -8.5184]

Atualizar w2:

w2[new] = w2[old] - learningRate * Loss.X

= [1 1] – 0.1*[-8.5184 -8.5184]= [1.85184 1.85184]

Atualizar w1:

$$\frac{\partial l}{\partial w^{[2]}} = Delta1 * X^T$$

Calculando para

Loss.x1 =$[-0.30392 \quad -0.30392]^T$
Loss.x2 =$[-0.3204 \quad -0.32049]^T$
Loss.x3 =$[-0.18192 \quad -0.18192]^T$

Loss.X = {soma de todas as Loss's} = $[-0.80624 \quad -0.80624]^T$

Atualizar w1:

W1[new] = w1[old] - learningRate * Loss.X

= $[1 \quad 1]^T$ – 0.1*$[-0.80624 \quad -0.80624]^T$ = $[1.080624 \quad 1.080624]^T$

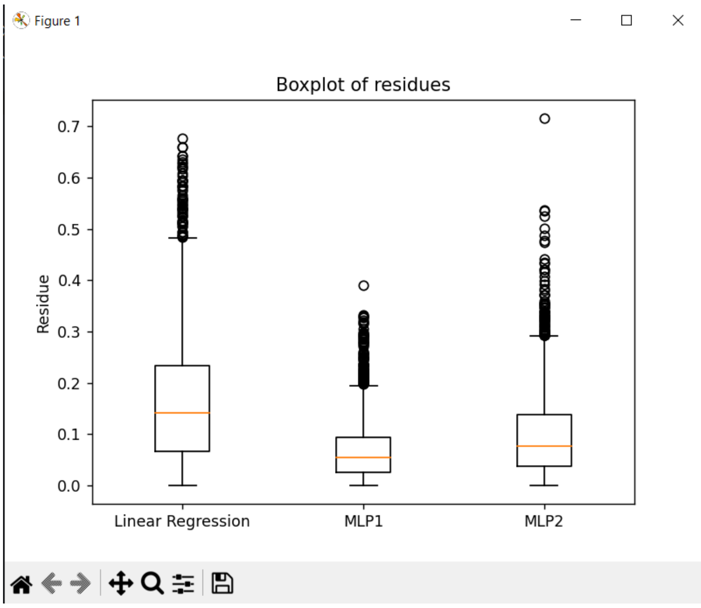
Aprendizagem 2021/22
Homework I – Group 081

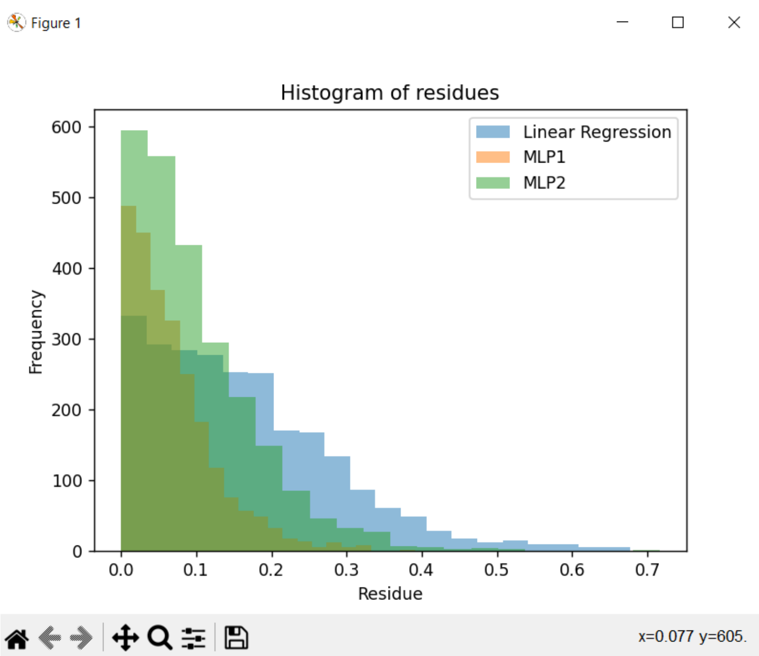
II. Programming and critical analysis

4) Answer 4

```
Linear Regression MAE:  0.162829976437694
MLP1 MAE:  0.0680414073796843
MLP2 MAE:  0.0978071820387748
```

5) Answer 5





Aprendizagem 2021/22
Homework I – Group 081

6) Answer 6

```
MLP1 converged in 452 iterations.
MLP2 converged in 77 iterations.
```

7) Answer 7

Exemplo para simplificar:

Vamos assumir que temos 100 exercicios num teste

Estudante 1: tem 90 perguntas disponiveis do teste para estudar e as outras 10 perguntas para testar o seu conhecimento, as 90 perguntas atualizam o conhecimento do estudante. Já as outras 10, servem apenas de controlo não atulizando o conhecimento do estudante.

Estudante 2: tem todas as 100 perguntas para estudar.

Resultados:

O Estudante 1 precisa de repetir o seu teste 452 vezes para que nas próximas tentativas acerte sempre 100% do teste;

O Estudante 2 precisa de repetir o seu teste apenas 77 vezes para que nas próximas tentativas acerte sempre 100% do teste;

(nota: O resultado de 100% no teste é apenas exemplificativo, na prática isto não é viável sendo um número próximo de 100% mas não atingindo este valor)

É normal que o estudante 1 precise de mais tentativas no mesmo teste para obter uma boa nota no teste integral a comparar com o estudante 2, dado que o estudante 2 tem acesso a todas as perguntas e o estudante 1 guarda 10 perguntas para saber se sabe a matéria.

É importante reparar que o Estudante 2 tem uma tendência mais elevada a 'decorar' o teste que o estudante 1, se agora tivessemos um novo teste com outras perguntas, provavelmente o estudante 1 teria resultados melhores.

Algo muito semelhante acontece nos resultados obtidos no exercicio 6. O MLP1 tem early stopping, reservando 10% do training data set para fazer um controlo adicional de resultados. Já o MLP2 não tem early stopping abusando do training data set na sua totalidade. Apesar de menos iterações (menos vezes que tem que percorrer o data set (77)) o MLP2 corre sérios riscos de Overfitting (decorar) o data set. O MLP1 apesar de mais iterações (tem que percorrer o data set 452 vezes) (provavelmente consumindo mais CPU e tempo) tem uma margem segura para evitar o overfitting.

III. APPENDIX

```
from scipy.io.arff import loadarff
from sklearn import metrics
import sklearn
import sklearn.metrics
import pandas as pd
```

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import Ridge
import numpy as np
from sklearn.neural_network import MLPRegressor
import matplotlib.pyplot as plt
from sklearn.metrics import mean_absolute_error

def boxplot(y_test, y_pred):
    plt.boxplot(np.abs(y_test - y_pred))
    plt.title('Boxplot of residues')
    plt.show()

def histogram(y_test, y_pred):
    plt.hist(np.abs(y_test - y_pred))
    plt.title('Histogram of residues')
    plt.show()

if __name__ == "__main__":
    # Load the data
    data = loadarff('kin8nm.arff')
    df = pd.DataFrame(data[0])
    X = df.drop('y', axis=1)
    y = df['y']
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=0)

    # Linear Regression
    lr = Ridge(alpha=0.1)
    lr.fit(X_train, y_train)
    y_pred = lr.predict(X_test)
    y_pred_lr = y_pred

    LR_MAE = mean_absolute_error(y_test, y_pred)

    # MLP1
    mlp1 = MLPRegressor(hidden_layer_sizes=(10,10), activation='tanh', max_iter=500,
random_state=0, early_stopping=True)
    mlp1.fit(X_train, y_train)
    y_pred = mlp1.predict(X_test)
    y_pred_mlp1 = y_pred

    MLP1_MAE = mean_absolute_error(y_test, y_pred)

    # MLP2
    mlp2 = MLPRegressor(hidden_layer_sizes=(10,10), activation='tanh', max_iter=500,
random_state=0, early_stopping=False)
    mlp2.fit(X_train, y_train)
```

```
y_pred = mlp2.predict(X_test)
y_pred_mlp2 = y_pred

MLP2_MAE = mean_absolute_error(y_test, y_pred)

print("MLP1 converged in {} iterations.".format(mlp1.n_iter_))
print("MLP2 converged in {} iterations.".format(mlp2.n_iter_))

print('Linear Regression MAE: ', LR_MAE)
print('MLP1 MAE: ', MLP1_MAE)
print('MLP2 MAE: ', MLP2_MAE)

# Boxplot
fig, ax = plt.subplots()
ax.boxplot([
    np.abs(y_test - y_pred_lr),
    np.abs(y_test - y_pred_mlp1),
    np.abs(y_test - y_pred_mlp2)
])
ax.set_xticklabels(["Linear Regression", "MLP1", "MLP2"])
ax.set_ylabel("Residue")
ax.set_title("Boxplot of residues")
plt.show()

# Histogram
fig, ax = plt.subplots()
ax.hist(np.abs(y_test - y_pred_lr),
        bins=20,
        alpha=0.5,
        label="Linear Regression")
ax.hist(np.abs(y_test - y_pred_mlp1), bins=20, alpha=0.5, label="MLP1")
ax.hist(np.abs(y_test - y_pred_mlp2), bins=20, alpha=0.5, label="MLP2")
ax.set_xlabel("Residue")
ax.set_ylabel("Frequency")
ax.set_title("Histogram of residues")
ax.legend()
plt.show()
```

END