In [1]:

**import** tensorflow

In [2]:

**import** pandas **as** pd  
**import** numpy **as** np  
**import** matplotlib.pyplot **as** plt

In [3]:

data**=**pd**.**read\_csv(r"C:\Users\dippy\Salary\_Data.csv")

In [4]:

data**.**head()

Out[4]:

|  | **YearsExperience** | **Salary** |
| --- | --- | --- |
| **0** | 1.1 | 39343.0 |
| **1** | 1.3 | 46205.0 |
| **2** | 1.5 | 37731.0 |
| **3** | 2.0 | 43525.0 |
| **4** | 2.2 | 39891.0 |

In [5]:

data**.**shape

Out[5]:

(30, 2)

In [6]:

data**.**head()

Out[6]:

|  | **YearsExperience** | **Salary** |
| --- | --- | --- |
| **0** | 1.1 | 39343.0 |
| **1** | 1.3 | 46205.0 |
| **2** | 1.5 | 37731.0 |
| **3** | 2.0 | 43525.0 |
| **4** | 2.2 | 39891.0 |

In [7]:

x**=**data**.**iloc[:,:**-**1]**.**values  
y**=**data**.**iloc[:,**-**1]**.**values

In [8]:

x

Out[8]:

array([[ 1.1],  
 [ 1.3],  
 [ 1.5],  
 [ 2. ],  
 [ 2.2],  
 [ 2.9],  
 [ 3. ],  
 [ 3.2],  
 [ 3.2],  
 [ 3.7],  
 [ 3.9],  
 [ 4. ],  
 [ 4. ],  
 [ 4.1],  
 [ 4.5],  
 [ 4.9],  
 [ 5.1],  
 [ 5.3],  
 [ 5.9],  
 [ 6. ],  
 [ 6.8],  
 [ 7.1],  
 [ 7.9],  
 [ 8.2],  
 [ 8.7],  
 [ 9. ],  
 [ 9.5],  
 [ 9.6],  
 [10.3],  
 [10.5]])

In [9]:

y

Out[9]:

array([ 39343., 46205., 37731., 43525., 39891., 56642., 60150.,  
 54445., 64445., 57189., 63218., 55794., 56957., 57081.,  
 61111., 67938., 66029., 83088., 81363., 93940., 91738.,  
 98273., 101302., 113812., 109431., 105582., 116969., 112635.,  
 122391., 121872.])

In [10]:

**from** sklearn.preprocessing **import** MinMaxScaler

In [11]:

scaler**=**MinMaxScaler()

In [12]:

scaler**.**fit(x)

Out[12]:

MinMaxScaler()

In [13]:

x\_scaled**=**scaler**.**transform(x)

In [14]:

x\_scaled

Out[14]:

array([[0. ],  
 [0.0212766 ],  
 [0.04255319],  
 [0.09574468],  
 [0.11702128],  
 [0.19148936],  
 [0.20212766],  
 [0.22340426],  
 [0.22340426],  
 [0.27659574],  
 [0.29787234],  
 [0.30851064],  
 [0.30851064],  
 [0.31914894],  
 [0.36170213],  
 [0.40425532],  
 [0.42553191],  
 [0.44680851],  
 [0.5106383 ],  
 [0.5212766 ],  
 [0.60638298],  
 [0.63829787],  
 [0.72340426],  
 [0.75531915],  
 [0.80851064],  
 [0.84042553],  
 [0.89361702],  
 [0.90425532],  
 [0.9787234 ],  
 [1. ]])

In [15]:

**from** sklearn.model\_selection **import** train\_test\_split

In [16]:

xtrain,xtest,ytrain,ytest**=**train\_test\_split(x\_scaled,y,test\_size**=**0.3,random\_state**=**123)

In [17]:

xtrain**.**shape

Out[17]:

(21, 1)

In [18]:

xtest**.**shape

Out[18]:

(9, 1)

In [19]:

**from** tensorflow.keras.models **import** Sequential  
**from** tensorflow.keras.layers **import** Dense

In [20]:

model**=**Sequential()  
model**.**add(Dense(6,activation**=**"relu"))  
model**.**add(Dense(4,activation**=**"relu"))  
model**.**add(Dense(2,activation**=**"relu"))  
model**.**add(Dense(1))

In [21]:

model**.**compile(optimizer**=**"rmsprop",loss**=**"mse")

In [22]:

model**.**fit(xtrain,ytrain,epochs**=**21,batch\_size**=**1)

Epoch 1/21  
21/21 [==============================] - 0s 571us/step - loss: 6072687104.0000  
Epoch 2/21  
21/21 [==============================] - 0s 666us/step - loss: 6072671232.0000  
Epoch 3/21  
21/21 [==============================] - 0s 619us/step - loss: 6072654848.0000  
Epoch 4/21  
21/21 [==============================] - 0s 666us/step - loss: 6072636416.0000  
Epoch 5/21  
21/21 [==============================] - 0s 619us/step - loss: 6072615936.0000  
Epoch 6/21  
21/21 [==============================] - 0s 609us/step - loss: 6072593408.0000  
Epoch 7/21  
21/21 [==============================] - 0s 1ms/step - loss: 6072566784.0000  
Epoch 8/21  
21/21 [==============================] - 0s 619us/step - loss: 6072535552.0000  
Epoch 9/21  
21/21 [==============================] - 0s 619us/step - loss: 6072502784.0000  
Epoch 10/21  
21/21 [==============================] - 0s 619us/step - loss: 6072466944.0000  
Epoch 11/21  
21/21 [==============================] - 0s 522us/step - loss: 6072424448.0000  
Epoch 12/21  
21/21 [==============================] - 0s 528us/step - loss: 6072382464.0000  
Epoch 13/21  
21/21 [==============================] - 0s 571us/step - loss: 6072335872.0000  
Epoch 14/21  
21/21 [==============================] - 0s 523us/step - loss: 6072283648.0000  
Epoch 15/21  
21/21 [==============================] - 0s 619us/step - loss: 6072229888.0000  
Epoch 16/21  
21/21 [==============================] - 0s 571us/step - loss: 6072167936.0000  
Epoch 17/21  
21/21 [==============================] - 0s 523us/step - loss: 6072100864.0000  
Epoch 18/21  
21/21 [==============================] - 0s 585us/step - loss: 6072028672.0000  
Epoch 19/21  
21/21 [==============================] - 0s 619us/step - loss: 6071953920.0000  
Epoch 20/21  
21/21 [==============================] - 0s 553us/step - loss: 6071872000.0000  
Epoch 21/21  
21/21 [==============================] - 0s 571us/step - loss: 6071778304.0000

Out[22]:

<tensorflow.python.keras.callbacks.History at 0x1a32c308520>

In [23]:

model**.**summary()

Model: "sequential"  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Layer (type) Output Shape Param #   
=================================================================  
dense (Dense) (1, 6) 12   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_1 (Dense) (1, 4) 28   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_2 (Dense) (1, 2) 10   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_3 (Dense) (1, 1) 3   
=================================================================  
Total params: 53  
Trainable params: 53  
Non-trainable params: 0  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

In [24]:

model**.**history**.**history["loss"]

Out[24]:

[6072687104.0,  
 6072671232.0,  
 6072654848.0,  
 6072636416.0,  
 6072615936.0,  
 6072593408.0,  
 6072566784.0,  
 6072535552.0,  
 6072502784.0,  
 6072466944.0,  
 6072424448.0,  
 6072382464.0,  
 6072335872.0,  
 6072283648.0,  
 6072229888.0,  
 6072167936.0,  
 6072100864.0,  
 6072028672.0,  
 6071953920.0,  
 6071872000.0,  
 6071778304.0]

In [25]:

**%matplotlib**

Using matplotlib backend: Qt5Agg

In [26]:

plt**.**plot(model**.**history**.**history["loss"],marker**=**"o")  
plt**.**xlabel("epoch")  
plt**.**ylabel("loss")  
plt**.**show()

In [27]:

ypred**=**model**.**predict(xtest)

In [28]:

**from** sklearn.metrics **import** explained\_variance\_score

In [29]:

explained\_variance\_score(ytest,ypred)

Out[29]:

9.536744655025409e-05

In [30]:

result**=**pd**.**DataFrame(ypred,columns**=**["ypred"])

In [31]:

result["ytest"]**=**ytest

In [32]:

result["error"]**=**result["ytest"]**-**result["ypred"]

In [33]:

result**.**head()

Out[33]:

|  | **ypred** | **ytest** | **error** |
| --- | --- | --- | --- |
| **0** | 5.499998 | 54445.0 | 54439.500002 |
| **1** | 8.823394 | 121872.0 | 121863.176606 |
| **2** | 5.363420 | 56642.0 | 56636.636580 |
| **3** | 8.368134 | 116969.0 | 116960.631866 |
| **4** | 5.499998 | 64445.0 | 64439.500002 |

In [34]:

tensorflow**.**keras**.**models**.**save\_model(model,"reg\_model.h5")

In [35]:

*### loading the model*

In [36]:

loaded\_model**=**tensorflow**.**keras**.**models**.**load\_model("reg\_model.h5",compile**=False**)

In [37]:

user\_input**=**scaler**.**transform([[1.1]])

In [38]:

user\_input

Out[38]:

array([[0.]])

In [39]:

loaded\_model**.**predict(user\_input)

Out[39]:

array([[4.5439525]], dtype=float32)

In [ ]: