

---

# MLP Regression Kaggle Data Set Processing

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

# Used for Confusion Matrix
from sklearn import metrics
import seaborn as sns

np.set_printoptions(precision=2, suppress=True)

from sklearn.datasets import fetch_openml
#dataset = fetch_openml("mnist_784")

# Used for Splitting Training and Test Sets
from sklearn.model_selection import train_test_split

%matplotlib inline
from tensorflow import keras

from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import LinearRegression

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Activation, Dropout
from tensorflow.keras.utils import to_categorical, plot_model
from tensorflow.keras.datasets import mnist

from tensorflow.keras.wrappers.scikit_learn import KerasRegressor
from sklearn.model_selection import cross_val_score
```

```
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold
```

```
s_list = []
```

```
intercept_list = []
weights_list = []
```

```
#df = pd.read_csv("HR.csv")
```

```
import pandas as pd
```

```
url = 'https://drive.google.com/file/d/0B6GhBwm5vaB2ekdlZW5WZnppb28/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[2]
```

```
url = 'https://drive.google.com/file/d/1Z02aj8aDCpSzHwUYXutMgpQmr9V8lInM/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[2]
df = pd.read_csv(path)
df.head()
```

```
df.head()
```

```
from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import LabelEncoder
labelencoder = LabelEncoder()
```

```
df['n_Gender'] = labelencoder.fit_transform(df['Gender'])
df['n_JobRole'] = labelencoder.fit_transform(df['JobRole'])
df['n_Attrition'] = labelencoder.fit_transform(df['Attrition'])
df['n_BusinessTravel'] = labelencoder.fit_transform(df['BusinessTravel'])
```

```
df['n_Department'] = labelencoder.fit_transform(df['Department'])
df['n_EducationField'] = labelencoder.fit_transform(df['EducationField'])
df.head()
```

```
df.drop(['Attrition', 'MaritalStatus', 'OverTime', 'Over18', 'BusinessTravel', 'JobRole', 'Gender', 'Department', 'EducationField'],
        axis=1, inplace=True)
df.head()
```

```

p = df['n_Attrition']
#df.drop(['n_Attrition'],axis=1, inplace=True)

from sklearn.model_selection import KFold
kf = KFold(n_splits=2, random_state=None, shuffle=True)

train = df.to_numpy()
test = p.to_numpy()
#.values.ravel()

dftemp = df
#p = df.from_dict(p,orient='index',columns=['n_Attrition'])
#p.shape()

for train_index, test_index in kf.split(df):
    #print("TRAIN:", train_index, "TEST:", test_index,"\n\n")
    #print("start TRAIN:", train_index, "TEST:", test_index,"end\n\n")

    X_train, X_test = df.iloc[train_index], df.iloc[test_index]

    y_train, y_test = X_train.loc[:,['n_Attrition']], X_test.loc[:,['n_Attrition']]

    X_train.drop(['n_Attrition'],axis=1, inplace=True)
    train_img = X_train

    X_test.drop(['n_Attrition'],axis=1, inplace=True)
    test_img = X_test

    train_lbl = y_train
    test_lbl = y_test

```

```
X_train = train_img
X_test = test_img
Y_train = train_lbl
Y_test = test_lbl
```

```
X_train = np.array( X_train)
X_train.shape
```

```
Y_train= np.array( Y_train)
```

```
#---
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Activation, Dropout
from tensorflow.keras.utils import to_categorical, plot_model
from tensorflow.keras.datasets import mnist

from tensorflow import keras
```

```
model = Sequential()
model.add(Dense(32, input_shape=(31,), activation='relu'))
#model.add(Dense(128, activation='relu'))
#model.add(Dense(64, activation='relu'))
#model.add(Dense(32, activation='relu'))
model.add(Dense(16, activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(4, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
#---
```

```
#optimizer = keras.optimizers.RMSprop(0.0099)
#model.compile(loss='mean_squared_error',optimizer=optimizer)

#results = model.compile(loss='mean_squared_error', optimizer='adam')

model.compile(loss='mse', optimizer='adam', metrics=['mse', 'mae', 'mape'])
# train model
history = model.fit(X_train, Y_train, epochs=100, batch_size=len(X_train), verbose=2)
# plot metrics

print(f'{history}')

from matplotlib import pyplot
pyplot.plot(history.history['mse'])

res = model.predict(X_test)

pyplot.show()

#print(f' predict {res}')
```

/usr/local/lib/python3.6/dist-packages/pandas/core/frame.py:4174: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#r](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#r)  
errors=errors,

Epoch 1/100

1/1 - 0s - loss: 0.2500 - mse: 0.2500 - mae: 0.5000 - mape: 423129248.0000

Epoch 2/100

1/1 - 0s - loss: 0.2498 - mse: 0.2498 - mae: 0.4998 - mape: 422917728.0000

Epoch 3/100

1/1 - 0s - loss: 0.2497 - mse: 0.2497 - mae: 0.4997 - mape: 422706176.0000

Epoch 4/100

1/1 - 0s - loss: 0.2495 - mse: 0.2495 - mae: 0.4995 - mape: 422494592.0000

Epoch 5/100

1/1 - 0s - loss: 0.2493 - mse: 0.2493 - mae: 0.4993 - mape: 422283040.0000

Epoch 6/100

1/1 - 0s - loss: 0.2491 - mse: 0.2491 - mae: 0.4991 - mape: 422071488.0000

Epoch 7/100

1/1 - 0s - loss: 0.2490 - mse: 0.2490 - mae: 0.4990 - mape: 421859904.0000

Epoch 8/100

1/1 - 0s - loss: 0.2488 - mse: 0.2488 - mae: 0.4988 - mape: 421648512.0000

Epoch 9/100

1/1 - 0s - loss: 0.2486 - mse: 0.2486 - mae: 0.4986 - mape: 421436960.0000

Epoch 10/100

1/1 - 0s - loss: 0.2484 - mse: 0.2484 - mae: 0.4984 - mape: 421225600.0000

Epoch 11/100

1/1 - 0s - loss: 0.2483 - mse: 0.2483 - mae: 0.4983 - mape: 421014048.0000

Epoch 12/100

1/1 - 0s - loss: 0.2481 - mse: 0.2481 - mae: 0.4981 - mape: 420802784.0000

Epoch 13/100

1/1 - 0s - loss: 0.2479 - mse: 0.2479 - mae: 0.4979 - mape: 420591360.0000

Epoch 14/100

1/1 - 0s - loss: 0.2478 - mse: 0.2478 - mae: 0.4978 - mape: 420380032.0000

Epoch 15/100

1/1 - 0s - loss: 0.2476 - mse: 0.2476 - mae: 0.4976 - mape: 420168768.0000

Epoch 16/100

1/1 - 0s - loss: 0.2474 - mse: 0.2474 - mae: 0.4974 - mape: 419957440.0000

Epoch 17/100

1/1 - 0s - loss: 0.2472 - mse: 0.2472 - mae: 0.4972 - mape: 419746208.0000

Epoch 18/100

1/1 - 0s - loss: 0.2471 - mse: 0.2471 - mae: 0.4971 - mape: 419535136.0000

Epoch 19/100

1/1 - 0s - loss: 0.2469 - mse: 0.2469 - mae: 0.4969 - mape: 419323936.0000

Epoch 20/100  
1/1 - 0s - loss: 0.2467 - mse: 0.2467 - mae: 0.4967 - mape: 419112800.0000  
Epoch 21/100  
1/1 - 0s - loss: 0.2466 - mse: 0.2466 - mae: 0.4965 - mape: 418901792.0000  
Epoch 22/100  
1/1 - 0s - loss: 0.2464 - mse: 0.2464 - mae: 0.4964 - mape: 418690880.0000  
Epoch 23/100  
1/1 - 0s - loss: 0.2462 - mse: 0.2462 - mae: 0.4962 - mape: 418479904.0000  
Epoch 24/100  
1/1 - 0s - loss: 0.2461 - mse: 0.2461 - mae: 0.4960 - mape: 418268928.0000  
Epoch 25/100  
1/1 - 0s - loss: 0.2459 - mse: 0.2459 - mae: 0.4959 - mape: 418058208.0000  
Epoch 26/100  
1/1 - 0s - loss: 0.2457 - mse: 0.2457 - mae: 0.4957 - mape: 417847360.0000  
Epoch 27/100  
1/1 - 0s - loss: 0.2455 - mse: 0.2455 - mae: 0.4955 - mape: 417636576.0000  
Epoch 28/100  
1/1 - 0s - loss: 0.2454 - mse: 0.2454 - mae: 0.4953 - mape: 417426080.0000  
Epoch 29/100  
1/1 - 0s - loss: 0.2452 - mse: 0.2452 - mae: 0.4952 - mape: 417215360.0000  
Epoch 30/100  
1/1 - 0s - loss: 0.2450 - mse: 0.2450 - mae: 0.4950 - mape: 417004928.0000  
Epoch 31/100  
1/1 - 0s - loss: 0.2449 - mse: 0.2449 - mae: 0.4948 - mape: 416794368.0000  
Epoch 32/100  
1/1 - 0s - loss: 0.2447 - mse: 0.2447 - mae: 0.4946 - mape: 416584000.0000  
Epoch 33/100  
1/1 - 0s - loss: 0.2445 - mse: 0.2445 - mae: 0.4945 - mape: 416373664.0000  
Epoch 34/100  
1/1 - 0s - loss: 0.2444 - mse: 0.2444 - mae: 0.4943 - mape: 416163456.0000  
Epoch 35/100  
1/1 - 0s - loss: 0.2442 - mse: 0.2442 - mae: 0.4941 - mape: 415953248.0000  
Epoch 36/100  
1/1 - 0s - loss: 0.2440 - mse: 0.2440 - mae: 0.4940 - mape: 415743168.0000  
Epoch 37/100  
1/1 - 0s - loss: 0.2439 - mse: 0.2439 - mae: 0.4938 - mape: 415533088.0000  
Epoch 38/100  
1/1 - 0s - loss: 0.2437 - mse: 0.2437 - mae: 0.4936 - mape: 415323072.0000  
Epoch 39/100  
1/1 - 0s - loss: 0.2435 - mse: 0.2435 - mae: 0.4934 - mape: 415113280.0000  
Epoch 40/100  
1/1 - 0s - loss: 0.2434 - mse: 0.2434 - mae: 0.4933 - mape: 414903328.0000  
Epoch 41/100  
1/1 - 0s - loss: 0.2432 - mse: 0.2432 - mae: 0.4931 - mape: 414693632.0000

-

Epoch 42/100  
1/1 - 0s - loss: 0.2430 - mse: 0.2430 - mae: 0.4929 - mape: 414483936.0000

Epoch 43/100  
1/1 - 0s - loss: 0.2429 - mse: 0.2429 - mae: 0.4928 - mape: 414274336.0000

Epoch 44/100  
1/1 - 0s - loss: 0.2427 - mse: 0.2427 - mae: 0.4926 - mape: 414064832.0000

Epoch 45/100  
1/1 - 0s - loss: 0.2425 - mse: 0.2425 - mae: 0.4924 - mape: 413855392.0000

Epoch 46/100  
1/1 - 0s - loss: 0.2424 - mse: 0.2424 - mae: 0.4922 - mape: 413645984.0000

Epoch 47/100  
1/1 - 0s - loss: 0.2422 - mse: 0.2422 - mae: 0.4921 - mape: 413436704.0000

Epoch 48/100  
1/1 - 0s - loss: 0.2420 - mse: 0.2420 - mae: 0.4919 - mape: 413227520.0000

Epoch 49/100  
1/1 - 0s - loss: 0.2419 - mse: 0.2419 - mae: 0.4917 - mape: 413018304.0000

Epoch 50/100  
1/1 - 0s - loss: 0.2417 - mse: 0.2417 - mae: 0.4916 - mape: 412809312.0000

Epoch 51/100  
1/1 - 0s - loss: 0.2415 - mse: 0.2415 - mae: 0.4914 - mape: 412600352.0000

Epoch 52/100  
1/1 - 0s - loss: 0.2414 - mse: 0.2414 - mae: 0.4912 - mape: 412391424.0000

Epoch 53/100  
1/1 - 0s - loss: 0.2412 - mse: 0.2412 - mae: 0.4910 - mape: 412182624.0000

Epoch 54/100  
1/1 - 0s - loss: 0.2410 - mse: 0.2410 - mae: 0.4909 - mape: 411973888.0000

Epoch 55/100  
1/1 - 0s - loss: 0.2409 - mse: 0.2409 - mae: 0.4907 - mape: 411765216.0000

Epoch 56/100  
1/1 - 0s - loss: 0.2407 - mse: 0.2407 - mae: 0.4905 - mape: 411556672.0000

Epoch 57/100  
1/1 - 0s - loss: 0.2406 - mse: 0.2406 - mae: 0.4904 - mape: 411348160.0000

Epoch 58/100  
1/1 - 0s - loss: 0.2404 - mse: 0.2404 - mae: 0.4902 - mape: 411139776.0000

Epoch 59/100  
1/1 - 0s - loss: 0.2402 - mse: 0.2402 - mae: 0.4900 - mape: 410931360.0000

Epoch 60/100  
1/1 - 0s - loss: 0.2401 - mse: 0.2401 - mae: 0.4898 - mape: 410723296.0000

Epoch 61/100  
1/1 - 0s - loss: 0.2399 - mse: 0.2399 - mae: 0.4897 - mape: 410515104.0000

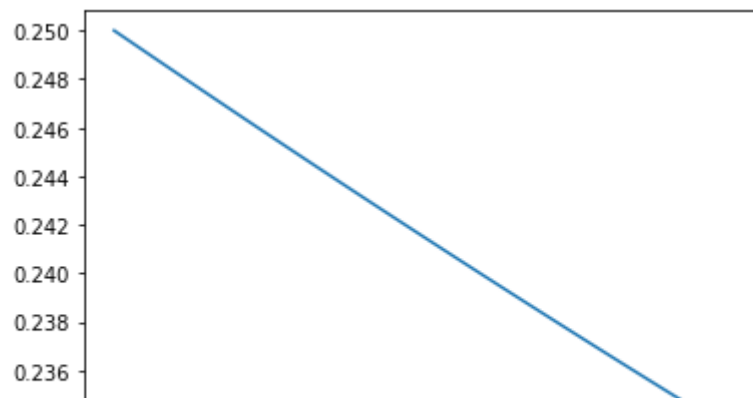
Epoch 62/100  
1/1 - 0s - loss: 0.2397 - mse: 0.2397 - mae: 0.4895 - mape: 410307040.0000

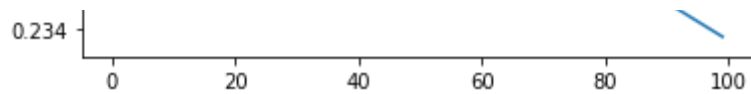
Epoch 63/100  
1/1 - 0s - loss: 0.2396 - mse: 0.2396 - mae: 0.4893 - mape: 410098040.0000



1/1 - 0s - loss: 0.2390 - mse: 0.2390 - mae: 0.4893 - mape: 410099040.0000  
Epoch 64/100  
1/1 - 0s - loss: 0.2394 - mse: 0.2394 - mae: 0.4892 - mape: 409891136.0000  
Epoch 65/100  
1/1 - 0s - loss: 0.2392 - mse: 0.2392 - mae: 0.4890 - mape: 409683328.0000  
Epoch 66/100  
1/1 - 0s - loss: 0.2391 - mse: 0.2391 - mae: 0.4888 - mape: 409475616.0000  
Epoch 67/100  
1/1 - 0s - loss: 0.2389 - mse: 0.2389 - mae: 0.4887 - mape: 409268032.0000  
Epoch 68/100  
1/1 - 0s - loss: 0.2388 - mse: 0.2388 - mae: 0.4885 - mape: 409060480.0000  
Epoch 69/100  
1/1 - 0s - loss: 0.2386 - mse: 0.2386 - mae: 0.4883 - mape: 408852992.0000  
Epoch 70/100  
1/1 - 0s - loss: 0.2384 - mse: 0.2384 - mae: 0.4881 - mape: 408645568.0000  
Epoch 71/100  
1/1 - 0s - loss: 0.2383 - mse: 0.2383 - mae: 0.4880 - mape: 408438368.0000  
Epoch 72/100  
1/1 - 0s - loss: 0.2381 - mse: 0.2381 - mae: 0.4878 - mape: 408231136.0000  
Epoch 73/100  
1/1 - 0s - loss: 0.2380 - mse: 0.2380 - mae: 0.4876 - mape: 408024064.0000  
Epoch 74/100  
1/1 - 0s - loss: 0.2378 - mse: 0.2378 - mae: 0.4875 - mape: 407817056.0000  
Epoch 75/100  
1/1 - 0s - loss: 0.2376 - mse: 0.2376 - mae: 0.4873 - mape: 407610080.0000  
Epoch 76/100  
1/1 - 0s - loss: 0.2375 - mse: 0.2375 - mae: 0.4871 - mape: 407403168.0000  
Epoch 77/100  
1/1 - 0s - loss: 0.2373 - mse: 0.2373 - mae: 0.4870 - mape: 407196480.0000  
Epoch 78/100  
1/1 - 0s - loss: 0.2372 - mse: 0.2372 - mae: 0.4868 - mape: 406989792.0000  
Epoch 79/100  
1/1 - 0s - loss: 0.2370 - mse: 0.2370 - mae: 0.4866 - mape: 406783104.0000  
Epoch 80/100  
1/1 - 0s - loss: 0.2368 - mse: 0.2368 - mae: 0.4865 - mape: 406576736.0000  
Epoch 81/100  
1/1 - 0s - loss: 0.2367 - mse: 0.2367 - mae: 0.4863 - mape: 406370240.0000  
Epoch 82/100  
1/1 - 0s - loss: 0.2365 - mse: 0.2365 - mae: 0.4861 - mape: 406163936.0000  
Epoch 83/100  
1/1 - 0s - loss: 0.2364 - mse: 0.2364 - mae: 0.4859 - mape: 405957728.0000  
Epoch 84/100  
1/1 - 0s - loss: 0.2362 - mse: 0.2362 - mae: 0.4858 - mape: 405751552.0000  
Epoch 85/100

1/1 - 0s - loss: 0.2360 - mse: 0.2360 - mae: 0.4856 - mape: 405545600.0000  
Epoch 86/100  
1/1 - 0s - loss: 0.2359 - mse: 0.2359 - mae: 0.4854 - mape: 405339520.0000  
Epoch 87/100  
1/1 - 0s - loss: 0.2357 - mse: 0.2357 - mae: 0.4853 - mape: 405133696.0000  
Epoch 88/100  
1/1 - 0s - loss: 0.2356 - mse: 0.2356 - mae: 0.4851 - mape: 404927936.0000  
Epoch 89/100  
1/1 - 0s - loss: 0.2354 - mse: 0.2354 - mae: 0.4849 - mape: 404722240.0000  
Epoch 90/100  
1/1 - 0s - loss: 0.2353 - mse: 0.2353 - mae: 0.4848 - mape: 404516672.0000  
Epoch 91/100  
1/1 - 0s - loss: 0.2351 - mse: 0.2351 - mae: 0.4846 - mape: 404311168.0000  
Epoch 92/100  
1/1 - 0s - loss: 0.2349 - mse: 0.2349 - mae: 0.4844 - mape: 404105760.0000  
Epoch 93/100  
1/1 - 0s - loss: 0.2348 - mse: 0.2348 - mae: 0.4843 - mape: 403900384.0000  
Epoch 94/100  
1/1 - 0s - loss: 0.2346 - mse: 0.2346 - mae: 0.4841 - mape: 403695200.0000  
Epoch 95/100  
1/1 - 0s - loss: 0.2345 - mse: 0.2345 - mae: 0.4839 - mape: 403490048.0000  
Epoch 96/100  
1/1 - 0s - loss: 0.2343 - mse: 0.2343 - mae: 0.4838 - mape: 403284928.0000  
Epoch 97/100  
1/1 - 0s - loss: 0.2342 - mse: 0.2342 - mae: 0.4836 - mape: 403079968.0000  
Epoch 98/100  
1/1 - 0s - loss: 0.2340 - mse: 0.2340 - mae: 0.4834 - mape: 402875104.0000  
Epoch 99/100  
1/1 - 0s - loss: 0.2338 - mse: 0.2338 - mae: 0.4833 - mape: 402670336.0000  
Epoch 100/100  
1/1 - 0s - loss: 0.2337 - mse: 0.2337 - mae: 0.4831 - mape: 402465632.0000  
<tensorflow.python.keras.callbacks.History object at 0x7f040010f4e0>





/usr/local/lib/python3.6/dist-packages/pandas/core/frame.py:4174: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#r](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#r)

errors=errors,

Epoch 1/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 2/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 3/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 4/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 5/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 6/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 7/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 8/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 9/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 10/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 11/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 12/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 13/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 14/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 15/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 16/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 17/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

Epoch 18/100

1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711

[illegible]

[illegible]

[illegible]

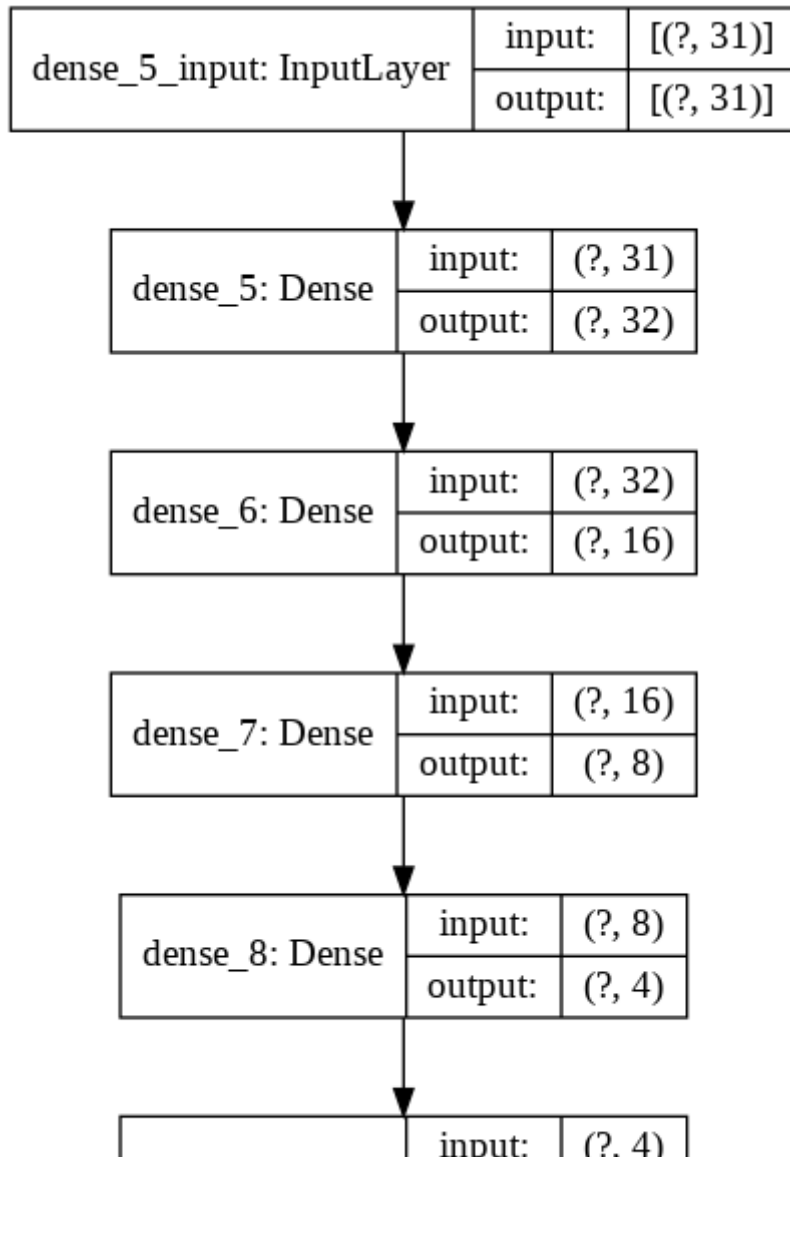
```

^
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 85/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 86/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 87/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 88/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 89/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 90/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 91/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 92/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 93/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 94/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 95/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 96/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 97/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 98/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 99/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
Epoch 100/100
1/1 - 0s - loss: 0.1687 - mse: 0.1687 - mae: 0.1687 - mape: 16.8711
<tensorflow.python.keras.callbacks.History object at 0x7f03a4209588>

```



```
plot_model(model, to_file='mlp-mnist.png', show_shapes=True)
```



```
#df1 = pd.read_csv("HR.csv")
```



```

import pandas as pd

url = 'https://drive.google.com/file/d/0B6GhBwm5vaB2ekdlZW5WZnppb28/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[-2]

url = 'https://drive.google.com/file/d/1Z02aj8aDCpSzHwUYXutMgpQmr9V8lInM/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[-2]
df1 = pd.read_csv(path)
#df.head()

df1.head()

```

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	EducationField	Employee
0	41	Yes	Travel_Rarely	1102	Sales	1	2	Life Sciences	
1	49	No	Travel_Frequently	279	Research & Development	8	1	Life Sciences	
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	Other	
3	33	No	Travel_Frequently	1392	Research & Development	3	4	Life Sciences	
4	27	No	Travel_Rarely	591	Research & Development	2	1	Medical	

## ▼ Splitting Data into Training and Test Sets

```
train_img.columns
```

```
Index(['Age', 'DailyRate', 'DistanceFromHome', 'Education', 'EmployeeCount',  
      'EmployeeNumber', 'EnvironmentSatisfaction', 'HourlyRate',  
      'JobInvolvement', 'JobLevel', 'JobSatisfaction', 'MonthlyIncome',  
      'MonthlyRate', 'NumCompaniesWorked', 'PercentSalaryHike',  
      'PerformanceRating', 'RelationshipSatisfaction', 'StandardHours',  
      'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',  
      'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole',  
      'YearsSinceLastPromotion', 'YearsWithCurrManager', 'n_Gender',  
      'n_JobRole', 'n_BusinessTravel', 'n_Department', 'n_EducationField'],  
      dtype='object')
```

```
X_train = train_img  
X_test = test_img  
Y_train = train_lbl  
Y_test = test_lbl
```

```
#print(f'Xtrain {X_train} X_test {X_test}')
```

```
print(f'x_train {X_train[0:1]}')  
print(f'y_train {Y_train[0:1]}')  
print(f'x_train {X_test[0:1]}')
```

```
x_train      Age  DailyRate  ...  n_Department  n_EducationField  
0    41      1102  ...           2              1
```

```
[1 rows x 31 columns]  
y_train      n_Attrition  
0              1
```

```
x_train    Age  DailyRate  ...  n_Department  n_EducationField
2    37      1373  ...           1             4
```

```
[1 rows x 31 columns]
```

```
print(train_img.shape)
```

```
(735, 31)
```

```
print(train_lbl.shape)
```

```
(735, 1)
```

```
print(test_img.shape)
```

```
(735, 31)
```

```
print(test_lbl.shape)
```

```
(735, 1)
```

```
s_list = []
```

```
intercept_list = []
```

```
weights_list = []
```

```
#df = pd.read_csv("HR.csv")
```

```
import pandas as pd
```

```
url = 'https://drive.google.com/file/d/0B6GhBwm5vaB2ekdlZW5WZnppb28/view?usp=sharing'
```

```
path = 'https://drive.google.com/uc?export=download&id= '+url.split('/')[ -2]
```

```

url = 'https://drive.google.com/file/d/1Z02aj8aDCpSzHwUYXutMgpQmr9V8lInM/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[2]
df = pd.read_csv(path)
df.head()

df.head()

from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import LabelEncoder
labelencoder = LabelEncoder()

df['n_Gender'] = labelencoder.fit_transform(df['Gender'])
df['n_JobRole']=labelencoder.fit_transform(df['JobRole'])
df['n_Attrition'] = labelencoder.fit_transform(df['Attrition'])
df['n_BusinessTravel'] = labelencoder.fit_transform(df['BusinessTravel'])

df['n_Department'] = labelencoder.fit_transform(df['Department'])
df['n_EducationField'] = labelencoder.fit_transform(df['EducationField'])
df.head()
df.drop(['Attrition','MaritalStatus','OverTime','Over18','BusinessTravel','JobRole','Gender','Department','EducationField'],axis=1, inplace=True)
df.head()
p = df['n_Attrition']
#df.drop(['n_Attrition'],axis=1, inplace=True)

from sklearn.model_selection import KFold
kf = KFold(n_splits=2, random_state=None, shuffle=True)

train = df.to_numpy()
test = p.to_numpy()
#.values.ravel()

dftemp = df

```

```
#p = df.from_dict(p,orient='index',columns=['n_Attrition'])
#p.shape()

for train_index, test_index in kf.split(df):
    #print("TRAIN:", train_index, "TEST:", test_index,"\n\n")
    #print("start TRAIN:", train_index, "TEST:", test_index,"end\n\n")

    X_train, X_test = df.iloc[train_index], df.iloc[test_index]

    y_train, y_test = X_train.loc[:,['n_Attrition']], X_test.loc[:,['n_Attrition']]

    X_train.drop(['n_Attrition'],axis=1, inplace=True)
    train_img = X_train

    X_test.drop(['n_Attrition'],axis=1, inplace=True)
    test_img = X_test

    train_lbl = y_train
    test_lbl = y_test

    X_train = train_img
    X_test = test_img
    Y_train = train_lbl
    Y_test = test_lbl

    X_train = np.array( X_train)
    X_train.shape

    Y_train= np.array( Y_train)

#---
```

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Activation, Dropout
from tensorflow.keras.utils import to_categorical, plot_model
from tensorflow.keras.datasets import mnist

from tensorflow import keras

model = Sequential()
model.add(Dense(16, input_shape=(31,), activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(4, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
#---

#optimizer = keras.optimizers.RMSprop(0.0099)
#model.compile(loss='mean_squared_error', optimizer=optimizer)

#results = model.compile(loss='mean_squared_error', optimizer='adam')

model.compile(loss='mse', optimizer='adam', metrics=['mse', 'mae', 'mape'])
# train model
history = model.fit(X_train, Y_train, epochs=100, batch_size=len(X_train), verbose=2)
# plot metrics

print(f'{history}')

from matplotlib import pyplot
pyplot.plot(history.history['mse'])

res = model.predict(X_test)
```

```
pyplot.show()
```

```
#print(f' predict {res}')
```

/usr/local/lib/python3.6/dist-packages/pandas/core/frame.py:4174: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#r](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#r)  
errors=errors,

Epoch 1/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 2/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 3/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 4/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 5/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 6/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 7/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000

Epoch 8/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 9/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 10/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 11/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 12/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 13/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 14/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 15/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 16/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

Epoch 17/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000

Epoch 18/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816448.0000

Epoch 19/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000

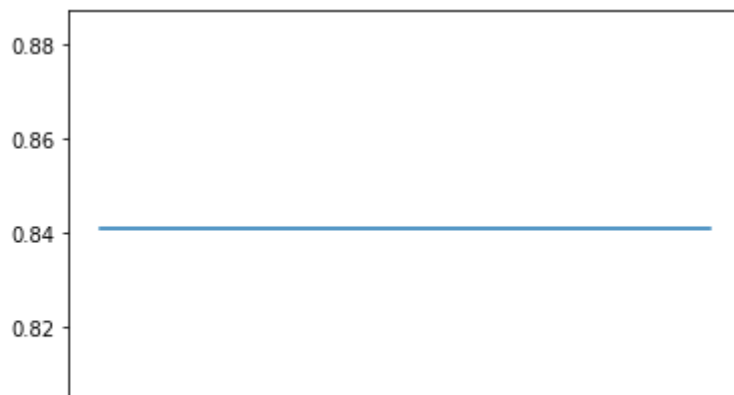


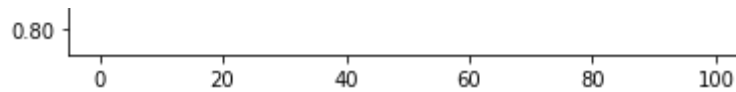
[illegible]

[illegible]

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 64/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 65/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 66/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 67/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 68/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 69/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 70/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816448.0000  
Epoch 71/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 72/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 73/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 74/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 75/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 76/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 77/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 78/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 79/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 80/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 81/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 82/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 83/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 84/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 85/100

1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 86/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 87/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 88/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 89/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 90/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 91/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 92/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 93/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 94/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 95/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 96/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 97/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 98/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816256.0000  
Epoch 99/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
Epoch 100/100  
1/1 - 0s - loss: 0.8408 - mse: 0.8408 - mae: 0.8408 - mape: 840816384.0000  
<tensorflow.python.keras.callbacks.History object at 0x7f03a403d470>





```
/usr/local/lib/python3.6/dist-packages/pandas/core/frame.py:4174: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame
```

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#r](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#r)

errors=errors,

Epoch 1/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 2/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000

Epoch 3/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 4/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 5/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 6/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000

Epoch 7/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 8/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000

Epoch 9/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 10/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 11/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 12/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 13/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000

Epoch 14/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 15/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000

Epoch 16/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 17/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000

Epoch 18/100

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000

[illegible]

1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 41/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 42/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 43/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 44/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 45/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 46/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 47/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 48/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 49/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 50/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 51/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 52/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 53/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 54/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 55/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 56/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 57/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 58/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 59/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 60/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 61/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 62/100

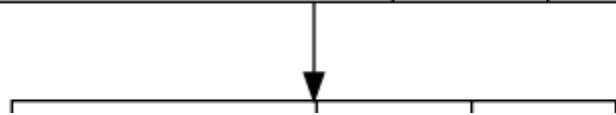
[illegible]



```
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 85/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 86/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 87/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 88/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734784.0000  
Epoch 89/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 90/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 91/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000  
Epoch 92/100  
1/1 - 0s - loss: 0.8367 - mse: 0.8367 - mae: 0.8367 - mape: 836734656.0000
```

```
plot_model(model, to_file='mlp-mnist.png', show_shapes=True)
```

dense_14_input: InputLayer	input:	[(?, 31)]
	output:	[(?, 31)]



|                      | output: | (?, 10) |



```

s_list = []

intercept_list = []
weights_list = []

#df = pd.read_csv("HR.csv")

import pandas as pd

url = 'https://drive.google.com/file/d/0B6GhBwm5vaB2ekdlZW5WZnppb28/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[-2]

url = 'https://drive.google.com/file/d/1Z02aj8aDCpSzHwUYXutMgpQmr9V8lInM/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[-2]
df = pd.read_csv(path)
df.head()

df.head()

from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import LabelEncoder
labelencoder = LabelEncoder()

df['n_Gender'] = labelencoder.fit_transform(df['Gender'])

```

```

df['n_JobRole']=labelencoder.fit_transform(df['JobRole'])
df['n_Attrition'] = labelencoder.fit_transform(df['Attrition'])
df['n_BusinessTravel'] = labelencoder.fit_transform(df['BusinessTravel'])

df['n_Department'] = labelencoder.fit_transform(df['Department'])
df['n_EducationField'] = labelencoder.fit_transform(df['EducationField'])
df.head()
df.drop(['Attrition','MaritalStatus','OverTime','Over18','BusinessTravel','JobRole','Gender','Department','EducationField'],axis=1, inplace=True)
df.head()
p = df['n_Attrition']
#df.drop(['n_Attrition'],axis=1, inplace=True)

```

```

from sklearn.model_selection import KFold
kf = KFold(n_splits=2, random_state=None, shuffle=True)

```

```

train = df.to_numpy()
test = p.to_numpy()
#.values.ravel()

```

```

dftemp = df
#p = df.from_dict(p,orient='index',columns=['n_Attrition'])
#p.shape()

```

```

for train_index, test_index in kf.split(df):
    #print("TRAIN:", train_index, "TEST:", test_index,"\n\n")
    #print("start TRAIN:", train_index, "TEST:", test_index,"end\n\n")

    X_train, X_test = df.iloc[train_index], df.iloc[test_index]

    y_train, y_test = X_train.loc[:,['n_Attrition']], X_test.loc[:,['n_Attrition']]

    X_train.drop(['n_Attrition'],axis=1, inplace=True)

```

```
train_img = X_train
```

```
X_test.drop(['n_Attrition'],axis=1, inplace=True)  
test_img = X_test
```

```
train_lbl = y_train  
test_lbl = y_test
```

```
X_train = train_img  
X_test = test_img  
Y_train = train_lbl  
Y_test = test_lbl
```

```
X_train = np.array( X_train)  
X_train.shape
```

```
Y_train= np.array( Y_train)
```

```
#---
```

```
from tensorflow.keras.models import Sequential  
from tensorflow.keras.layers import Dense, Activation, Dropout  
from tensorflow.keras.utils import to_categorical, plot_model  
from tensorflow.keras.datasets import mnist
```

```
from tensorflow import keras
```

```
model = Sequential()
```

```
model.add(Dense(10, input_shape=(28, 28), activation='tanh'))
```

```

model.add(Dense(16, input_shape=(31,), activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
#---

#optimizer = keras.optimizers.RMSprop(0.0099)
#model.compile(loss='mean_squared_error',optimizer=optimizer)

#results = model.compile(loss='mean_squared_error', optimizer='adam')

model.compile(loss='mse', optimizer='adam', metrics=['mse', 'mae', 'mape'])
# train model
history = model.fit(X_train, Y_train, epochs=100, batch_size=len(X_train), verbose=2)
# plot metrics

print(f'{history}')

from matplotlib import pyplot
pyplot.plot(history.history['mse'])

res = model.predict(X_test)

pyplot.show()

#print(f' predict {res}')
```

```
/usr/local/lib/python3.6/dist-packages/pandas/core/frame.py:4174: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame
```

See the caveats in the documentation: [Epoch 1/100](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#r<br/>errors=errors,</a></p></div><div data-bbox=)

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 2/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 3/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 4/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 5/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 6/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 7/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 8/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 9/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 10/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 11/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 12/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 13/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 14/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 15/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 16/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 17/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 18/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

Epoch 19/100

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020

[illegible]

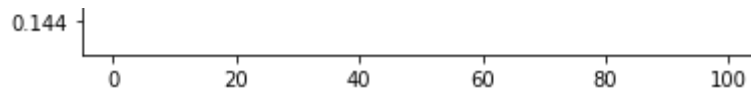
[illegible]



[illegible]

1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 86/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 87/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 88/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 89/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 90/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 91/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 92/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 93/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 94/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 95/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 96/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 97/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 98/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 99/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
Epoch 100/100  
1/1 - 0s - loss: 0.1510 - mse: 0.1510 - mae: 0.1510 - mape: 15.1020  
<tensorflow.python.keras.callbacks.History object at 0x7f039d609c18>





/usr/local/lib/python3.6/dist-packages/pandas/core/frame.py:4174: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#r](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#r)

errors=errors,

Epoch 1/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 2/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 3/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 4/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 5/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 6/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 7/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 8/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 9/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 10/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 11/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 12/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 13/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 14/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 15/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 16/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 17/100

1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

Epoch 18/100

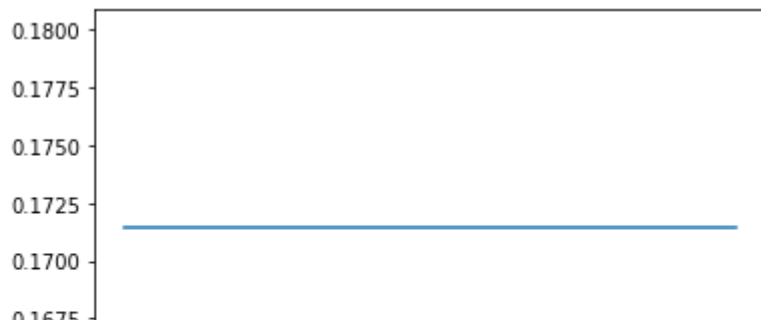
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429

```
Epoch 19/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 20/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 21/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 22/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 23/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 24/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 25/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 26/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 27/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 28/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 29/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 30/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 31/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 32/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 33/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 34/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 35/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 36/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 37/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 38/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 39/100
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429
Epoch 40/100
```

[illegible]

[illegible]

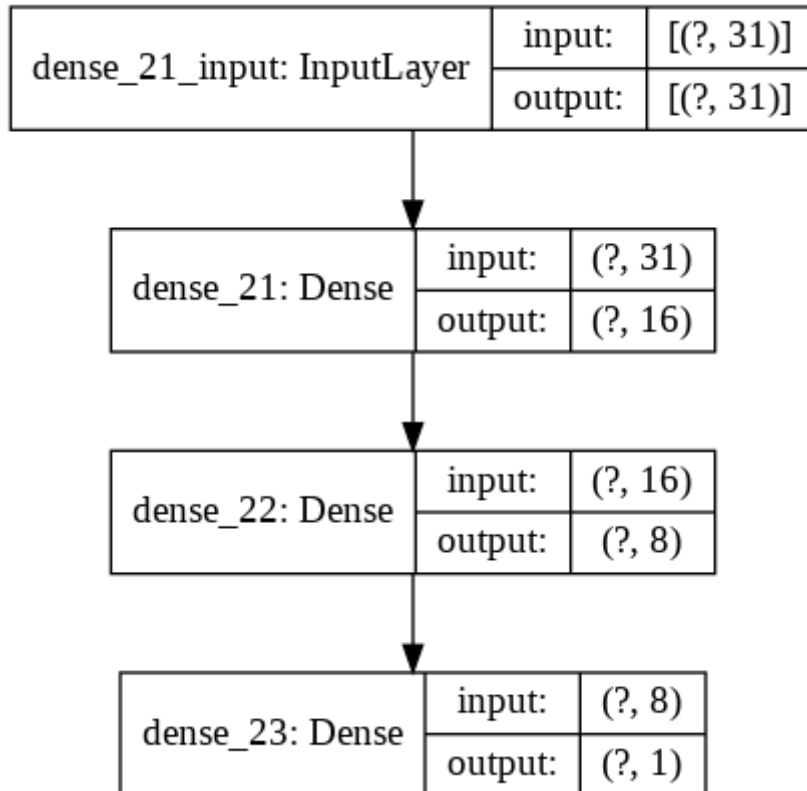
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 85/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 86/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 87/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 88/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 89/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 90/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 91/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 92/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 93/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 94/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 95/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 96/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 97/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 98/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 99/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
Epoch 100/100  
1/1 - 0s - loss: 0.1714 - mse: 0.1714 - mae: 0.1714 - mape: 17.1429  
<tensorflow.python.keras.callbacks.History object at 0x7f040007f550>



```

0.10/3 |
plot_model(model, to_file='mlp-mnist.png', show_shapes=True)

```



```

s_list = []

intercept_list = []
weights_list = []

#df = pd.read_csv("HR.csv")

import pandas as pd

url = 'https://drive.google.com/file/d/0B6GhBwm5vaB2ekdlZW5WZnppb28/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[2]

```



```

url = 'https://drive.google.com/file/d/1Z02aj8aDCpSzHwUYXutMgpQmr9V8lInM/view?usp=sharing'
path = 'https://drive.google.com/uc?export=download&id='+url.split('/')[2]
df = pd.read_csv(path)
df.head()

df.head()

from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import LabelEncoder
labelencoder = LabelEncoder()

df['n_Gender'] = labelencoder.fit_transform(df['Gender'])
df['n_JobRole']=labelencoder.fit_transform(df['JobRole'])
df['n_Attrition'] = labelencoder.fit_transform(df['Attrition'])
df['n_BusinessTravel'] = labelencoder.fit_transform(df['BusinessTravel'])

df['n_Department'] = labelencoder.fit_transform(df['Department'])
df['n_EducationField'] = labelencoder.fit_transform(df['EducationField'])
df.head()
df.drop(['Attrition','MaritalStatus','OverTime','Over18','BusinessTravel','JobRole','Gender','Department','EducationField'],axis=1, inplace=True)
df.head()
p = df['n_Attrition']
#df.drop(['n_Attrition'],axis=1, inplace=True)

from sklearn.model_selection import KFold
kf = KFold(n_splits=2, random_state=None, shuffle=True)

train = df.to_numpy()
test = p.to_numpy()
#.values.ravel()

dftemp = df
#p = df.from_dict(p.orient='index',columns=['n_Attrition'])

```

```

#p.shape()

for train_index, test_index in kf.split(df):
    #print("TRAIN:", train_index, "TEST:", test_index,"\n\n")
    #print("start TRAIN:", train_index, "TEST:", test_index,"end\n\n")

    X_train, X_test = df.iloc[train_index], df.iloc[test_index]

    y_train, y_test = X_train.loc[:,['n_Attrition']], X_test.loc[:,['n_Attrition']]

    X_train.drop(['n_Attrition'],axis=1, inplace=True)
    train_img = X_train

    X_test.drop(['n_Attrition'],axis=1, inplace=True)
    test_img = X_test

    train_lbl = y_train
    test_lbl = y_test

    X_train = train_img
    X_test = test_img
    Y_train = train_lbl
    Y_test = test_lbl

    X_train = np.array( X_train)
    X_train.shape

    Y_train= np.array( Y_train)

#---
from tensorflow.keras.models import Sequential

```

```
from tensorflow.keras.layers import Dense, Activation, Dropout
from tensorflow.keras.utils import to_categorical, plot_model
from tensorflow.keras.datasets import mnist

from tensorflow import keras

model = Sequential()
model.add(Dense(16, input_shape=(31,), activation='relu'))

model.add(Dense(1, activation='sigmoid'))
#---

#optimizer = keras.optimizers.RMSprop(0.0099)
#model.compile(loss='mean_squared_error',optimizer=optimizer)

#results = model.compile(loss='mean_squared_error', optimizer='adam')

model.compile(loss='mse', optimizer='adam', metrics=['mse', 'mae', 'mape'])
# train model
history = model.fit(X_train, Y_train, epochs=100, batch_size=len(X_train), verbose=2)
# plot metrics

print(f'{history}')

from matplotlib import pyplot
pyplot.plot(history.history['mse'])

res = model.predict(X_test)

pyplot.show()
```

```
--- ..
```

```
#print(f' predict {res}')
```

/usr/local/lib/python3.6/dist-packages/pandas/core/frame.py:4174: SettingWithCopyWarning:  
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#r](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#r)  
errors=errors,

Epoch 1/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 2/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 3/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 4/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 5/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 6/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 7/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 8/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 9/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 10/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 11/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 12/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 13/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 14/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 15/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 16/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 17/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 18/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 19/100

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

[illegible]

-

Epoch 42/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 43/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 44/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 45/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 46/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 47/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 48/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 49/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 50/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 51/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 52/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 53/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 54/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 55/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 56/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 57/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 58/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374272.0000

Epoch 59/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 60/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 61/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000

Epoch 62/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

Epoch 63/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000

1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 64/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000  
Epoch 65/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 66/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374272.0000  
Epoch 67/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000  
Epoch 68/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 69/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 70/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000  
Epoch 71/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000  
Epoch 72/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 73/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000  
Epoch 74/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000  
Epoch 75/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 76/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 77/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 78/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000  
Epoch 79/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 80/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 81/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 82/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 83/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374208.0000  
Epoch 84/100  
1/1 - 0s - loss: 0.8354 - mse: 0.8354 - mae: 0.8354 - mape: 835374080.0000  
Epoch 85/100