

Ex No: 2a

Date: 02-08-2024

BUILD A SIMPLE NEURAL NETWORK WITH KERAS

AIM:

To build a simple neural network using Keras/TensorFlow.

PROCEDURE:

1. Download and load the dataset.
2. Perform analysis and preprocessing of the dataset.
3. Build a simple neural network model using Keras/TensorFlow.
4. Compile and fit the model.
5. Perform prediction with the test dataset.
6. Calculate performance metrics.

PROGRAM:

```
import numpy as np

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense

from sklearn.model_selection import train_test_split

np.random.seed(42)

dataset = np.loadtxt('pima-indians-diabetes.csv', delimiter=',')

X = dataset[:, 0:8]

y = dataset[:, 8]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = Sequential()

model.add(Dense(12, input_shape=(8,), activation='relu'))

model.add(Dense(6, activation='relu'))
```

```

model.add(Dense(8, activation='relu'))

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

model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])

model.fit(X_train, y_train, epochs=150, batch_size=10, validation_split=0.2)

_, accuracy = model.evaluate(X_test, y_test)

print('Test Accuracy: %.2f%%' % (accuracy * 100))

```

OUTPUT:

```

Epoch 146/150
50/50 [=====] - 0s 4ms/step - loss: 0.4667 - accuracy: 0.7536 - val_loss: 0.5456 - val_accuracy: 0.7
236
Epoch 147/150
50/50 [=====] - 0s 4ms/step - loss: 0.4684 - accuracy: 0.7678 - val_loss: 0.5487 - val_accuracy: 0.7
642
Epoch 148/150
50/50 [=====] - 0s 7ms/step - loss: 0.4600 - accuracy: 0.7780 - val_loss: 0.5621 - val_accuracy: 0.7
236
Epoch 149/150
50/50 [=====] - 0s 4ms/step - loss: 0.4823 - accuracy: 0.7576 - val_loss: 0.6012 - val_accuracy: 0.7
154
Epoch 150/150
50/50 [=====] - 0s 5ms/step - loss: 0.4761 - accuracy: 0.7556 - val_loss: 0.5671 - val_accuracy: 0.7
480

```

In [9]: *# Evaluate the Keras model on the testing dataset*

```

_, accuracy = model.evaluate(X_test, y_test)
print('Test Accuracy: %.2f%%' % (accuracy * 100))

```

```

5/5 [=====] - 0s 3ms/step - loss: 0.6862 - accuracy: 0.6753
Test Accuracy: 67.53%

```

RESULT:

Thus, a simple neural network using Keras/TensorFlow was built successfully.