Assignment No: 2B

Roll No. : - 19121028

Title of the Assignment:

> Multiclass classification using Deep Neural Networks: Example: Use the OCR letter recognition dataset https://archive.ics.uci.edu/ml/datasets/letter+recognition

```
import pandas as pd

# Load the dataset
data = pd.read_csv("https://archive.ics.uci.edu/ml/machine-learning-databases/letter-recognition/letter-recognition.data", header=None)
```

```
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split

# Preprocess the dataset
X = data.iloc[:, 1:]
y = data.iloc[:, 0]

encoder = LabelEncoder()
y = encoder.fit_transform(y)

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

```
from keras.models import Sequential
from keras.layers import Dense, Dropout

# Build the DNN model
model = Sequential()
model.add(Dense(units=64, activation='relu', input_dim=X_train.shape[1]))
model.add(Dropout(0.2))
model.add(Dense(units=32, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(units=26, activation='softmax'))
```

```
# Train the model
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'])
model.fit(X_train, y_train, epochs=10, batch_size=32)
```

```
Epoch 1/10
500/500 [==============================] - 2s 2ms/step - loss: 3.1019 - accuracy: 0.1321
Epoch 2/10
500/500 [==============================] - 2s 3ms/step - loss: 1.9212 - accuracy: 0.4059
Epoch 3/10
500/500 [==============================] - 2s 3ms/step - loss: 1.5352 - accuracy: 0.5162
Epoch 4/10
500/500 [==============================] - 1s 2ms/step - loss: 1.3511 - accuracy: 0.5789
Epoch 5/10
500/500 [==============================] - 1s 2ms/step - loss: 1.2501 - accuracy: 0.6059
Epoch 6/10
500/500 [==============================] - 1s 2ms/step - loss: 1.1832 - accuracy: 0.6273
Epoch 7/10
500/500 [==============================] - 1s 2ms/step - loss: 1.1323 - accuracy: 0.6428
Epoch 8/10
500/500 [==============================] - 1s 2ms/step - loss: 1.0795 - accuracy: 0.6582
Epoch 9/10
500/500 [==============================] - 1s 3ms/step - loss: 1.0518 - accuracy: 0.6661
Epoch 10/10
500/500 [==============================] - 3s 6ms/step - loss: 1.0166 - accuracy: 0.6773
<keras.callbacks.History at 0x7f2f85b18bb0>
```

```
# Test the model
loss, accuracy = model.evaluate(X_test, y_test)
print("Test Loss:", loss)
print("Test Accuracy:", accuracy)
```

```
125/125 [==============================] - 1s 3ms/step - loss: 0.7014 - accuracy: 0.8012
Test Loss: 0.7014068365097046
Test Accuracy: 0.8012499809265137
```

```
# Make predictions
new_image = pd.DataFrame([[2, 8, 3, 7, 2, 1, 8, 9, 5, 7, 8, 8, 7, 6, 10, 8]])
prediction = model.predict(new_image)
prediction_letter = encoder.inverse_transform([prediction.argmax()])
print("Prediction:", prediction_letter[0])
```

```
1/1 [==============================] - 0s 137ms/step
Prediction: K
```