

Machine Learning Assignment

Name :Muhammad Hassan

Reg No:FA21-BCE-051

Assignment :04

Submitted to:Dr Shoaib Azmat

Task 01 :

Code:

```
#import pandas as pd
import numpy as np
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.layers import Rescaling, Normalization, Resizing
from keras.datasets import cifar10
(x_train, y_train), (x_test, y_test) = cifar10.load_data()
inputs = keras.Input(shape=(32, 32, 3))
#x = Resizing(height=227, width=227)(inputs)
x = Rescaling(scale=1.0 / 255)(inputs)
x = layers.Conv2D(filters=32, kernel_size=(3, 3),padding='same', activation="relu")(x)
x = layers.MaxPooling2D(pool_size=(2, 2))(x)
#x = layers.BatchNormalization()(x)
x = layers.Conv2D(filters=64, kernel_size=(3, 3),padding='same', activation="relu")(x)
x = layers.MaxPooling2D(pool_size=(2, 2))(x)
#x = layers.BatchNormalization()(x)
x = layers.Conv2D(filters=128, kernel_size=(3, 3),padding='same', activation="relu")(x)
x = layers.MaxPooling2D(pool_size=(2, 2))(x)
#x = layers.BatchNormalization()(x)
x = layers.Flatten()(x)
x=layers.Dropout(0.25)(x)
outputs = layers.Dense(10, activation="softmax")(x)
x=layers.Dropout(0.5)(x)
model = keras.Model(inputs=inputs, outputs=outputs)
model.summary()
model.compile(optimizer='adam',loss='sparse_categorical_crossentropy',metrics=['accuracy'])
batch_size =75
model.fit(x_train, y_train, batch_size=batch_size, epochs=30) #Training the model
model.evaluate(x_test, y_test, batch_size=batch_size) # Testing the model
import matplotlib.pyplot as plt
predictions = model.predict(x_test)
plt.figure(figsize=(10, 10))
for i, image in enumerate(x_test[0:16,:,:]):
    plt.subplot(4, 4, i + 1)
    plt.imshow(image,cmap='gray')
    model_out=np.argmax(predictions[i])
    plt.title(f'Actual: {model_out};Predict: {y_test[i]}')
plt.axis(False)
```

Result:

Model: "model"

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	[(None, 32, 32, 3)]	0
rescaling (Rescaling)	(None, 32, 32, 3)	0
conv2d (Conv2D)	(None, 32, 32, 32)	896
max_pooling2d (MaxPooling2D)	(None, 16, 16, 32)	0
conv2d_1 (Conv2D)	(None, 16, 16, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 8, 8, 64)	0
conv2d_2 (Conv2D)	(None, 8, 8, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 4, 4, 128)	0
flatten (Flatten)	(None, 2048)	0
dropout (Dropout)	(None, 2048)	0
dense (Dense)	(None, 10)	20490

=====
Total params: 113738 (444.29 KB)
Trainable params: 113738 (444.29 KB)
Non-trainable params: 0 (0.00 Byte)
=====

Epoch 1/30
667/667 [=====] - 9s 5ms/step - loss: 1.5346 - accuracy: 0.4440
Epoch 2/30
667/667 [=====] - 3s 5ms/step - loss: 1.1281 - accuracy: 0.6027
Epoch 3/30
667/667 [=====] - 3s 4ms/step - loss: 0.9782 - accuracy: 0.6590
.
.
.
.
.
Epoch 29/30
667/667 [=====] - 3s 5ms/step - loss: 0.2832 - accuracy: 0.8980
Epoch 30/30
667/667 [=====] - 3s 4ms/step - loss: 0.2771 - accuracy: 0.9005
134/134 [=====] - 1s 3ms/step - loss: 0.8292 - accuracy: 0.7693
[0.8291670680046082, 0.7692999839782715]

Actual:3;Predict:[3]



Actual:1;Predict:[8]



Actual:8;Predict:[8]



Actual:0;Predict:[0]



Actual:6;Predict:[6]



Actual:6;Predict:[6]



Actual:1;Predict:[1]



Actual:4;Predict:[6]



Actual:3;Predict:[3]



Actual:1;Predict:[1]



Actual:0;Predict:[0]



Actual:9;Predict:[9]



Actual:5;Predict:[5]



Actual:7;Predict:[7]



Actual:9;Predict:[9]



Actual:8;Predict:[8]



Response/Comments:

After so many attempts ,At least I will reach to appropriate model which fit train data or accuracy on train data is 76% and its prediction or accuracy on test data is also 76 %.

When first time ,I try it ,my model make over fit the data then following parameters we changed.

I try to did changed many things ,such as no of filters ,epochs,batch size,padding and strides well as add normalization

All this parameters make a key role in performance

Padding save our data from lost ,strides take a step its like a alpha take too much it leave information,or too small take too much time ,epochs like a gradient decent iteration ,if it are too much then most probably it over fit the model or too small may be it under fit the mode,batch size like a segment of f full train data set

I have go towards good fitting by setting epochs and batchsize,also try to do other things but major parameters are these two