

## ▼ CNN on CIFR Assignment:

1. Please visit this link to access the state-of-art DenseNet code for reference - DenseNet - cifar10 notebook link
2. You need to create a copy of this and "retrain" this model to achieve 90+ test accuracy.
3. You cannot use Dense Layers (also called fully connected layers), or DropOut.
4. You MUST use Image Augmentation Techniques.
5. You cannot use an already trained model as a beginning points, you have to initilize as your own
6. You cannot run the program for more than 300 Epochs, and it should be clear from your log, that you have only used 300 Epochs
7. You cannot use test images for training the model.
8. You cannot change the general architecture of DenseNet (which means you must use Dense Block, Transition and Output blocks as mentioned in the code)
9. You are free to change Convolution types (e.g. from 3x3 normal convolution to Depthwise Separable, etc)
10. You cannot have more than 1 Million parameters in total
11. You are free to move the code from Keras to Tensorflow, Pytorch, MXNET etc.

Warning: you are connected to a GPU runtime, but not utilizing the GPU. [Change to a standard runtime](#)



need of training the model from first if you lost at any epoch while training. You can directly load that model and Train from that epoch.

```
# import keras
# from keras.datasets import cifar10
# from keras.models import Model, Sequential
# from keras.layers import Dense, Dropout, Flatten, Input, AveragePooling2D, merge, Activation
# from keras.layers import Conv2D, MaxPooling2D, BatchNormalization
# from keras.layers import Concatenate
# from keras.optimizers import Adam
from tensorflow.keras import models, layers
from tensorflow.keras.models import Model
from tensorflow.keras.layers import BatchNormalization, Activation, Flatten
from tensorflow.keras.optimizers import Adam

# this part will prevent tensorflow to allocate all the available GPU Memory
# backend
import tensorflow as tf

# Hyperparameters
batch_size = 128
num_classes = 10
epochs = 10
```

```
epochs = 10
l = 40
num_filter = 12
compression = 0.5
dropout_rate = 0.2
```

```
# Load CIFAR10 Data
```

```
(X_train, y_train), (X_test, y_test) = tf.keras.datasets.cifar10.load_data()
img_height, img_width, channel = X_train.shape[1], X_train.shape[2], X_train.shape[3]
```

```
# convert to one hot encoding
```

```
y_train = tf.keras.utils.to_categorical(y_train, num_classes)
y_test = tf.keras.utils.to_categorical(y_test, num_classes)
```

```
↳ Downloading data from https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz
170500096/170498071 [=====] - 6s 0us/step
```

```
X_train.shape
```

```
↳ (50000, 32, 32, 3)
```

```
X_test.shape
```

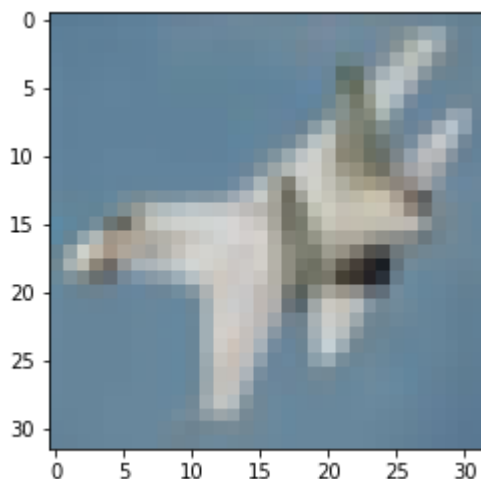
```
↳ (10000, 32, 32, 3)
```

```
from keras.preprocessing.image import ImageDataGenerator
```

Warning: you are connected to a GPU runtime, but not utilizing the GPU. [Change to a standard runtime](#)

```
from matplotlib import pyplot
pyplot.imshow(X_test[10])
```

```
↳ <matplotlib.image.AxesImage at 0x7f5b3df9c780>
```



```
#https://github.com/moritzhambach/Image-Augmentation-in-Keras-CIFAR-10-/blob/master/CNN%20
```

```
datagen = ImageDataGenerator(
    rotation_range=15,
    horizontal_flip=True,
    width_shift_range=0.1,
    height_shift_range=0.1
    #zoom_range=0.3
```

```

    )
    datagen.fit(X_train)

# Dense Block
def denseblock(input, num_filter = 12, dropout_rate = 0.0):
    global compression
    temp = input
    for _ in range(1):
        BatchNorm = layers.BatchNormalization()(temp)
        relu = layers.Activation('relu')(BatchNorm)
        Conv2D_3_3 = layers.Conv2D(int(num_filter*compression), (3,3), use_bias=False ,padding='same')
        if dropout_rate>0:
            Conv2D_3_3 = layers.Dropout(dropout_rate)(Conv2D_3_3)
        concat = layers.Concatenate(axis=-1)([temp,Conv2D_3_3])

        temp = concat

    return temp

```

```

## transition Block
def transition(input, num_filter = 12, dropout_rate = 0.0):
    global compression
    BatchNorm = layers.BatchNormalization()(input)
    relu = layers.Activation('relu')(BatchNorm)
    Conv2D_BottleNeck = layers.Conv2D(int(num_filter*compression), (1,1), use_bias=False ,padding='same')
    if dropout_rate>0:
        Conv2D_BottleNeck = layers.Dropout(dropout_rate)(Conv2D_BottleNeck)

```

Warning: you are connected to a GPU runtime, but not utilizing the GPU. [Change to a standard runtime](#)

```

#output layer
def output_layer(input):
    global compression
    BatchNorm = layers.BatchNormalization()(input)
    relu = layers.Activation('relu')(BatchNorm)
    AvgPooling = layers.AveragePooling2D(pool_size=(2,2))(relu)
    flat = layers.Flatten()(AvgPooling)
    output = layers.Dense(num_classes, activation='softmax')(flat)
    return output

num_filter = 32
dropout_rate = 0.0
l = 12
input = layers.Input(shape=(img_height, img_width, channel,))
First_Conv2D = layers.Conv2D(num_filter, (3,3), use_bias=False ,padding='same')(input)

First_Block = denseblock(First_Conv2D, num_filter, dropout_rate)
First_Transition = transition(First_Block, num_filter, dropout_rate)

Second_Block = denseblock(First_Transition, num_filter, dropout_rate)
Second_Transition = transition(Second_Block, num_filter, dropout_rate)

Third_Block = denseblock(Second_Transition, num_filter, dropout_rate)
Third_Transition = transition(Third_Block, num_filter, dropout_rate)

```

```
Last_Block = denseblock(Third_Transition, num_filter, dropout_rate)
output = output_layer>Last_Block)

# model for DenseNet
model = Model(inputs=[input], outputs=[output])

model.summary()
```



Warning: you are connected to a GPU runtime, but not utilizing the GPU.

[Change to a standard runtime](#)



Model: "functional\_3"

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	[(None, 32, 32, 3)]	0	
conv2d_52 (Conv2D)	(None, 32, 32, 32)	864	input_2[0][0]
batch_normalization_52 (Batch Normalization)	(None, 32, 32, 32)	128	conv2d_52[0][0]
activation_52 (Activation)	(None, 32, 32, 32)	0	batch_normalization_52[0][0]
conv2d_53 (Conv2D)	(None, 32, 32, 16)	4608	activation_52[0][0]
concatenate_48 (Concatenate)	(None, 32, 32, 48)	0	conv2d_52[0][0] conv2d_53[0][0]
batch_normalization_53 (Batch Normalization)	(None, 32, 32, 48)	192	concatenate_48[0][0]
activation_53 (Activation)	(None, 32, 32, 48)	0	batch_normalization_53[0][0]
conv2d_54 (Conv2D)	(None, 32, 32, 16)	6912	activation_53[0][0]
concatenate_49 (Concatenate)	(None, 32, 32, 64)	0	concatenate_48[0][0] conv2d_54[0][0]
batch_normalization_54 (Batch Normalization)	(None, 32, 32, 64)	256	concatenate_49[0][0]
activation_54 (Activation)	(None, 32, 32, 64)	0	batch_normalization_54[0][0]
conv2d_55 (Conv2D)	(None, 32, 32, 16)	11520	activation_54[0][0]
concatenate_50 (Concatenate)	(None, 32, 32, 96)	0	concatenate_49[0][0] conv2d_55[0][0]
batch_normalization_55 (Batch Normalization)	(None, 32, 32, 96)	320	concatenate_50[0][0]
activation_55 (Activation)	(None, 32, 32, 96)	0	batch_normalization_55[0][0]
conv2d_56 (Conv2D)	(None, 32, 32, 16)	11520	activation_55[0][0]
concatenate_51 (Concatenate)	(None, 32, 32, 96)	0	concatenate_50[0][0] conv2d_56[0][0]
batch_normalization_56 (Batch Normalization)	(None, 32, 32, 96)	384	concatenate_51[0][0]
activation_56 (Activation)	(None, 32, 32, 96)	0	batch_normalization_56[0][0]
conv2d_57 (Conv2D)	(None, 32, 32, 16)	13824	activation_56[0][0]
concatenate_52 (Concatenate)	(None, 32, 32, 112)	0	concatenate_51[0][0] conv2d_57[0][0]
batch_normalization_57 (Batch Normalization)	(None, 32, 32, 112)	448	concatenate_52[0][0]
activation_57 (Activation)	(None, 32, 32, 112)	0	batch_normalization_57[0][0]
conv2d_58 (Conv2D)	(None, 32, 32, 16)	16128	activation_57[0][0]
concatenate_53 (Concatenate)	(None, 32, 32, 128)	0	concatenate_52[0][0] conv2d_58[0][0]

Warning: you are connected to a GPU runtime, but not utilizing the GPU.  
runtime

[Change to a standard runtime](#)



batch_normalization_58 (BatchNo	(None, 32, 32, 128)	512	concatenate_53[0][0]
activation_58 (Activation)	(None, 32, 32, 128)	0	batch_normalization_58[0][0]
conv2d_59 (Conv2D)	(None, 32, 32, 16)	18432	activation_58[0][0]
concatenate_54 (Concatenate)	(None, 32, 32, 144)	0	concatenate_53[0][0] conv2d_59[0][0]
batch_normalization_59 (BatchNo	(None, 32, 32, 144)	576	concatenate_54[0][0]
activation_59 (Activation)	(None, 32, 32, 144)	0	batch_normalization_59[0][0]
conv2d_60 (Conv2D)	(None, 32, 32, 16)	20736	activation_59[0][0]
concatenate_55 (Concatenate)	(None, 32, 32, 160)	0	concatenate_54[0][0] conv2d_60[0][0]
batch_normalization_60 (BatchNo	(None, 32, 32, 160)	640	concatenate_55[0][0]
activation_60 (Activation)	(None, 32, 32, 160)	0	batch_normalization_60[0][0]
conv2d_61 (Conv2D)	(None, 32, 32, 16)	23040	activation_60[0][0]
concatenate_56 (Concatenate)	(None, 32, 32, 176)	0	concatenate_55[0][0] conv2d_61[0][0]
batch_normalization_61 (BatchNo	(None, 32, 32, 176)	704	concatenate_56[0][0]
activation_61 (Activation)	(None, 32, 32, 176)	0	batch_normalization_61[0][0]
conv2d_62 (Conv2D)	(None, 32, 32, 16)	25600	activation_61[0][0]
concatenate_57 (Concatenate)	(None, 32, 32, 192)	0	concatenate_56[0][0] conv2d_62[0][0]
batch_normalization_62 (BatchNo	(None, 32, 32, 192)	768	concatenate_57[0][0]
activation_62 (Activation)	(None, 32, 32, 192)	0	batch_normalization_62[0][0]
conv2d_63 (Conv2D)	(None, 32, 32, 16)	27648	activation_62[0][0]
concatenate_58 (Concatenate)	(None, 32, 32, 208)	0	concatenate_57[0][0] conv2d_63[0][0]
batch_normalization_63 (BatchNo	(None, 32, 32, 208)	832	concatenate_58[0][0]
activation_63 (Activation)	(None, 32, 32, 208)	0	batch_normalization_63[0][0]
conv2d_64 (Conv2D)	(None, 32, 32, 16)	29952	activation_63[0][0]
concatenate_59 (Concatenate)	(None, 32, 32, 224)	0	concatenate_58[0][0] conv2d_64[0][0]
batch_normalization_64 (BatchNo	(None, 32, 32, 224)	896	concatenate_59[0][0]
activation_64 (Activation)	(None, 32, 32, 224)	0	batch_normalization_64[0][0]
conv2d_65 (Conv2D)	(None, 32, 32, 16)	3584	activation_64[0][0]
average_pooling2d_4 (AveragePool	(None, 16, 16, 16)	0	conv2d_65[0][0]

Warning: you are connected to a GPU runtime, but not utilizing the GPU.  
runtime

[Change to a standard](#)



average_pooling2d_4 (AveragePool2D)	(None, 16, 16, 16)	0	conv2d_65[0][0]
batch_normalization_65 (Batch Normalization)	(None, 16, 16, 16)	64	average_pooling2d_4[0][0]
activation_65 (Activation)	(None, 16, 16, 16)	0	batch_normalization_65[0][0]
conv2d_66 (Conv2D)	(None, 16, 16, 16)	2304	activation_65[0][0]
concatenate_60 (Concatenate)	(None, 16, 16, 32)	0	average_pooling2d_4[0][0] conv2d_66[0][0]
batch_normalization_66 (Batch Normalization)	(None, 16, 16, 32)	128	concatenate_60[0][0]
activation_66 (Activation)	(None, 16, 16, 32)	0	batch_normalization_66[0][0]
conv2d_67 (Conv2D)	(None, 16, 16, 16)	4608	activation_66[0][0]
concatenate_61 (Concatenate)	(None, 16, 16, 48)	0	concatenate_60[0][0] conv2d_67[0][0]
batch_normalization_67 (Batch Normalization)	(None, 16, 16, 48)	192	concatenate_61[0][0]
activation_67 (Activation)	(None, 16, 16, 48)	0	batch_normalization_67[0][0]
conv2d_68 (Conv2D)	(None, 16, 16, 16)	6912	activation_67[0][0]
concatenate_62 (Concatenate)	(None, 16, 16, 64)	0	concatenate_61[0][0] conv2d_68[0][0]
batch_normalization_68 (Batch Normalization)	(None, 16, 16, 64)	256	concatenate_62[0][0]

Warning: you are connected to a GPU runtime, but not utilizing the GPU.  
runtime

[Change to a standard runtime](#)



concatenate_63 (Concatenate)	(None, 16, 16, 80)	0	concatenate_62[0][0] conv2d_69[0][0]
batch_normalization_69 (Batch Normalization)	(None, 16, 16, 80)	320	concatenate_63[0][0]
activation_69 (Activation)	(None, 16, 16, 80)	0	batch_normalization_69[0][0]
conv2d_70 (Conv2D)	(None, 16, 16, 16)	11520	activation_69[0][0]
concatenate_64 (Concatenate)	(None, 16, 16, 96)	0	concatenate_63[0][0] conv2d_70[0][0]
batch_normalization_70 (Batch Normalization)	(None, 16, 16, 96)	384	concatenate_64[0][0]
activation_70 (Activation)	(None, 16, 16, 96)	0	batch_normalization_70[0][0]
conv2d_71 (Conv2D)	(None, 16, 16, 16)	13824	activation_70[0][0]
concatenate_65 (Concatenate)	(None, 16, 16, 112)	0	concatenate_64[0][0] conv2d_71[0][0]
batch_normalization_71 (Batch Normalization)	(None, 16, 16, 112)	448	concatenate_65[0][0]
activation_71 (Activation)	(None, 16, 16, 112)	0	batch_normalization_71[0][0]
conv2d_72 (Conv2D)	(None, 16, 16, 16)	16128	activation_71[0][0]

concatenate_66 (Concatenate)	(None, 16, 16, 128)	0	concatenate_65[0][0] conv2d_72[0][0]
batch_normalization_72 (Batch Normalization)	(None, 16, 16, 128)	512	concatenate_66[0][0]
activation_72 (Activation)	(None, 16, 16, 128)	0	batch_normalization_72[0][0]
conv2d_73 (Conv2D)	(None, 16, 16, 16)	18432	activation_72[0][0]
concatenate_67 (Concatenate)	(None, 16, 16, 144)	0	concatenate_66[0][0] conv2d_73[0][0]
batch_normalization_73 (Batch Normalization)	(None, 16, 16, 144)	576	concatenate_67[0][0]
activation_73 (Activation)	(None, 16, 16, 144)	0	batch_normalization_73[0][0]
conv2d_74 (Conv2D)	(None, 16, 16, 16)	20736	activation_73[0][0]
concatenate_68 (Concatenate)	(None, 16, 16, 160)	0	concatenate_67[0][0] conv2d_74[0][0]
batch_normalization_74 (Batch Normalization)	(None, 16, 16, 160)	640	concatenate_68[0][0]
activation_74 (Activation)	(None, 16, 16, 160)	0	batch_normalization_74[0][0]
conv2d_75 (Conv2D)	(None, 16, 16, 16)	23040	activation_74[0][0]
concatenate_69 (Concatenate)	(None, 16, 16, 176)	0	concatenate_68[0][0] conv2d_75[0][0]
batch_normalization_75 (Batch Normalization)	(None, 16, 16, 176)	704	concatenate_69[0][0]
activation_75 (Activation)	(None, 16, 16, 176)	0	batch_normalization_75[0][0]
conv2d_76 (Conv2D)	(None, 16, 16, 16)	25344	activation_75[0][0]
concatenate_70 (Concatenate)	(None, 16, 16, 192)	0	concatenate_69[0][0] conv2d_76[0][0]
batch_normalization_76 (Batch Normalization)	(None, 16, 16, 192)	768	concatenate_70[0][0]
activation_76 (Activation)	(None, 16, 16, 192)	0	batch_normalization_76[0][0]
conv2d_77 (Conv2D)	(None, 16, 16, 16)	27648	activation_76[0][0]
concatenate_71 (Concatenate)	(None, 16, 16, 208)	0	concatenate_70[0][0] conv2d_77[0][0]
batch_normalization_77 (Batch Normalization)	(None, 16, 16, 208)	832	concatenate_71[0][0]
activation_77 (Activation)	(None, 16, 16, 208)	0	batch_normalization_77[0][0]
conv2d_78 (Conv2D)	(None, 16, 16, 16)	3328	activation_77[0][0]
average_pooling2d_5 (Average Pooling)	(None, 8, 8, 16)	0	conv2d_78[0][0]
batch_normalization_78 (Batch Normalization)	(None, 8, 8, 16)	64	average_pooling2d_5[0][0]
activation_78 (Activation)	(None, 8, 8, 16)	0	batch_normalization_78[0][0]
conv2d_79 (Conv2D)	(None, 8, 8, 16)	2304	activation_78[0][0]

Warning: you are connected to a GPU runtime, but not utilizing the GPU.  
runtime

[Change to a standard runtime](#)

