1. Problem Statement :

Classification of ciFAR-10 images using a convolution neural network.

2. Proposed Solution :

3. Implementation details :

A. Download Data

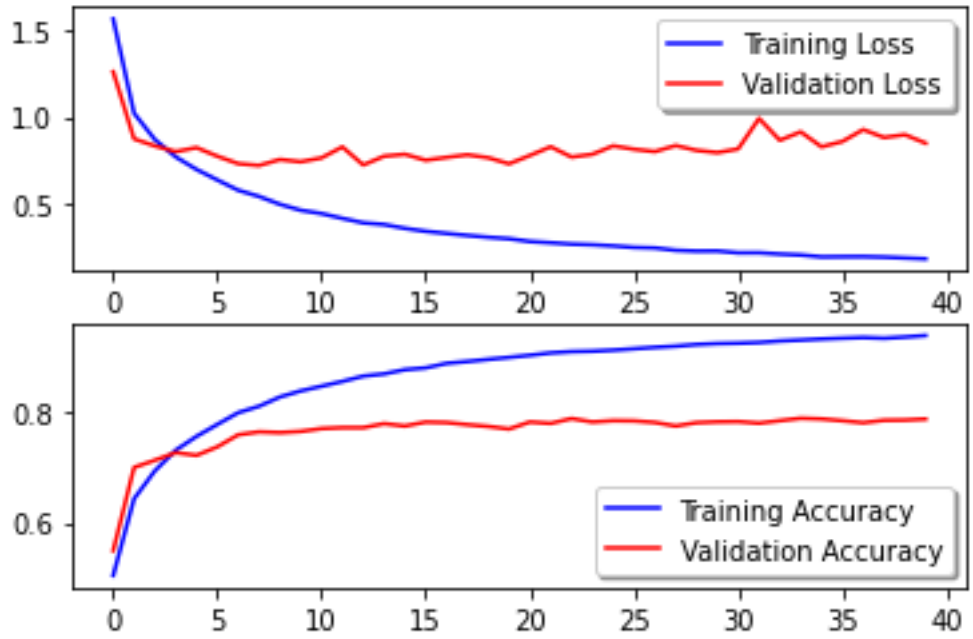
B. Create Covnet Model

Model: "sequential_2"

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 32, 32, 32)	896
max_pooling2d_3 (MaxPooling2	(None, 16, 16, 32)	0
batch_normalization_4 (Batch	(None, 16, 16, 32)	128
conv2d_4 (Conv2D)	(None, 16, 16, 128)	36992
max_pooling2d_4 (MaxPooling2	(None, 8, 8, 128)	0
batch_normalization_5 (Batch	(None, 8, 8, 128)	512
dropout_3 (Dropout)	(None, 8, 8, 128)	0
flatten_2 (Flatten)	(None, 8192)	0
dense_3 (Dense)	(None, 512)	4194816
batch_normalization_6 (Batch	(None, 512)	2048
dropout_4 (Dropout)	(None, 512)	0
dense_4 (Dense)	(None, 10)	5130

Total params: 4,240,522
Trainable params: 4,239,178
Non-trainable params: 1,344

C. Perform Hyperparameter tuning



As you can after 17 epoch validation set starts to overfit so we choose epoch value as 17 and retrain the model with new hyperparamter (epoch = 18)

On Test set :
 Accuracy : 0.765900
 Loss :   0.843699

D: Adding Inception Block to our network block:

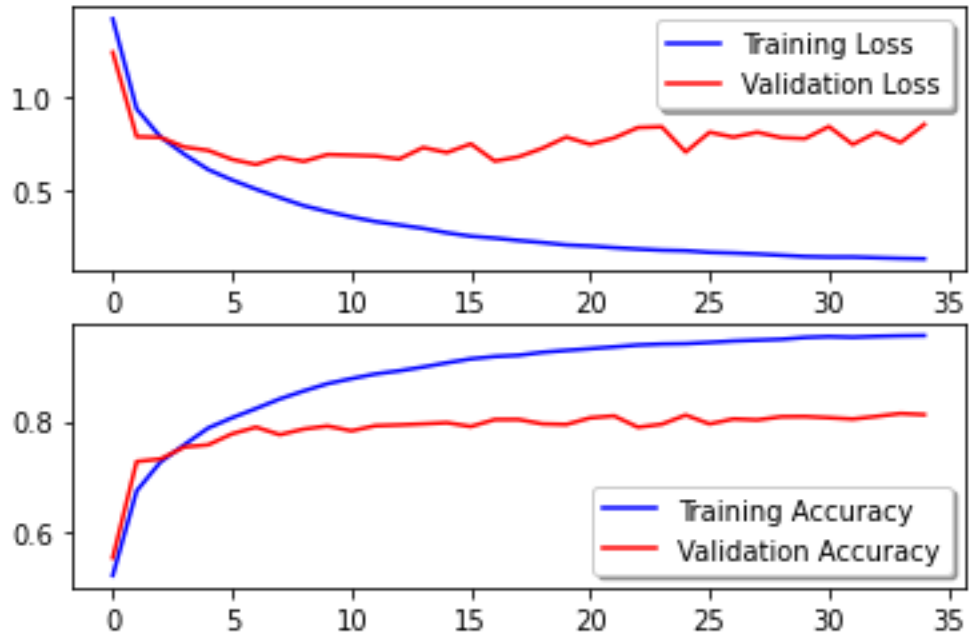
```
Model: "model_4"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_4 (InputLayer)            (None, 32, 32, 3)    0
__________________________________________________________________________________________________
conv2d_22 (Conv2D)              (None, 32, 32, 32)   896         input_4[0][0]
__________________________________________________________________________________________________
max_pooling2d_10 (MaxPooling2D) (None, 16, 16, 32)   0           conv2d_22[0][0]
__________________________________________________________________________________________________
```

```
batch_normalization_10 (BatchNo (None, 16, 16, 32)   128
max_pooling2d_10[0][0]
__________________________________________________________________________________________________
conv2d_23 (Conv2D)              (None, 16, 16, 128)  36992
batch_normalization_10[0][0]
__________________________________________________________________________________________________
max_pooling2d_11 (MaxPooling2D) (None, 8, 8, 128)    0
conv2d_23[0][0]
__________________________________________________________________________________________________
batch_normalization_11 (BatchNo (None, 8, 8, 128)    512
max_pooling2d_11[0][0]
__________________________________________________________________________________________________
conv2d_24 (Conv2D)              (None, 8, 8, 64)     8256
batch_normalization_11[0][0]
__________________________________________________________________________________________________
conv2d_26 (Conv2D)              (None, 8, 8, 64)     8256
batch_normalization_11[0][0]
__________________________________________________________________________________________________
max_pooling2d_12 (MaxPooling2D) (None, 8, 8, 128)    0
batch_normalization_11[0][0]
__________________________________________________________________________________________________
conv2d_25 (Conv2D)              (None, 8, 8, 64)     36928
conv2d_24[0][0]
__________________________________________________________________________________________________
conv2d_27 (Conv2D)              (None, 8, 8, 64)     102464
conv2d_26[0][0]
__________________________________________________________________________________________________
conv2d_28 (Conv2D)              (None, 8, 8, 64)     8256
max_pooling2d_12[0][0]
__________________________________________________________________________________________________
concatenate_4 (Concatenate)     (None, 8, 8, 192)    0
conv2d_25[0][0]

conv2d_27[0][0]

conv2d_28[0][0]
__________________________________________________________________________________________________
dropout_7 (Dropout)             (None, 8, 8, 192)    0
concatenate_4[0][0]
__________________________________________________________________________________________________
flatten_4 (Flatten)             (None, 12288)        0
dropout_7[0][0]
```

```
_____________________________________________________________________________
_____________________
dense_7 (Dense)                 (None, 512)          6291968
flatten_4[0][0]
_____________________________________________________________________________
_____________________
batch_normalization_12 (BatchNo (None, 512)          2048
dense_7[0][0]
_____________________________________________________________________________
_____________________
dropout_8 (Dropout)             (None, 512)          0
batch_normalization_12[0][0]
_____________________________________________________________________________
_____________________
dense_8 (Dense)                 (None, 10)           5130
dropout_8[0][0]
==============================================================================
======================
Total params: 6,501,834
Trainable params: 6,500,490
Non-trainable params: 1,344
```



As you can after 13 epoch validation set starts to overfit so we choose epoch value as 13 and retrain the model with new hyperparamter (epoch = 13)

- On Test set :
  Accuracy : 0.7936999797821045

Loss : 0.6680989986181259

E. Adding ResNet block to our network:

Model Architecture :

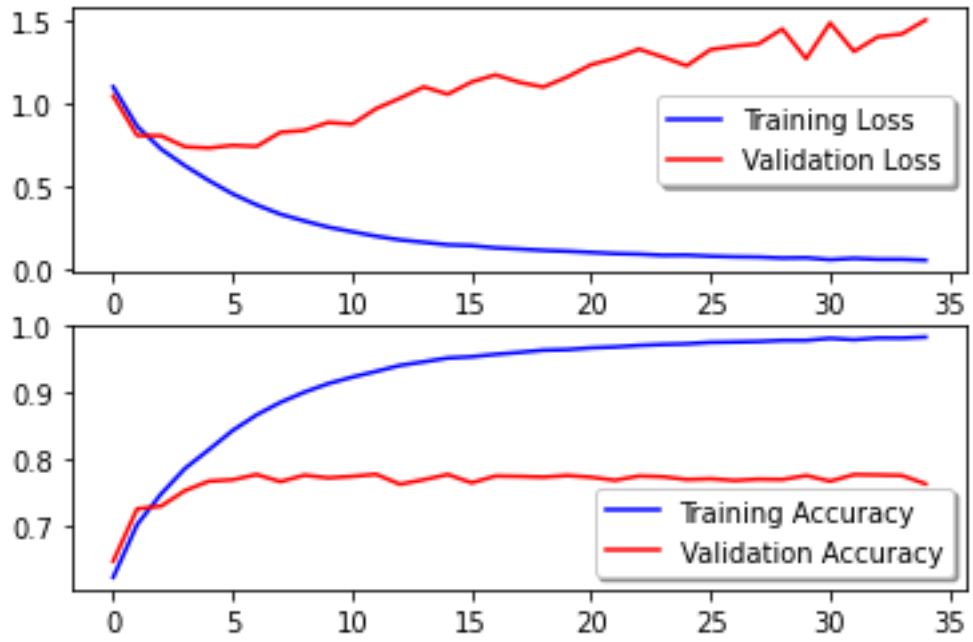
Model: "model_2"

Layer (type)	Output Shape	Param #	Connected to
input_2 (InputLayer)	(None, 32, 32, 3)	0	
conv2d_7 (Conv2D)	(None, 32, 32, 32)	896	input_2[0][0]
max_pooling2d_3 (MaxPooling2D)	(None, 16, 16, 32)	0	conv2d_7[0][0]
batch_normalization_8 (BatchNor	(None, 16, 16, 32)	128	max_pooling2d_3[0][0]
conv2d_8 (Conv2D)	(None, 16, 16, 128)	36992	batch_normalization_8[0][0]
max_pooling2d_4 (MaxPooling2D)	(None, 8, 8, 128)	0	conv2d_8[0][0]
batch_normalization_9 (BatchNor	(None, 8, 8, 128)	512	max_pooling2d_4[0][0]
conv2d_9 (Conv2D)	(None, 4, 4, 128)	147584	batch_normalization_9[0][0]
re_lu_5 (ReLU)	(None, 4, 4, 128)	0	conv2d_9[0][0]
batch_normalization_10 (BatchNo	(None, 4, 4, 128)	512	re_lu_5[0][0]

```
conv2d_10 (Conv2D)              (None, 4, 4, 128)    147584
batch_normalization_10[0][0]
__________________________________________________________________________________________________
re_lu_6 (ReLU)                  (None, 4, 4, 128)    0
conv2d_10[0][0]
__________________________________________________________________________________________________
batch_normalization_11 (BatchNo (None, 4, 4, 128)    512
re_lu_6[0][0]
__________________________________________________________________________________________________
conv2d_11 (Conv2D)              (None, 4, 4, 128)    147584
batch_normalization_11[0][0]
__________________________________________________________________________________________________
re_lu_7 (ReLU)                  (None, 4, 4, 128)    0
conv2d_11[0][0]
__________________________________________________________________________________________________
conv2d_12 (Conv2D)              (None, 4, 4, 128)    16512
batch_normalization_9[0][0]
__________________________________________________________________________________________________
batch_normalization_12 (BatchNo (None, 4, 4, 128)    512
re_lu_7[0][0]
__________________________________________________________________________________________________
add_2 (Add)                     (None, 4, 4, 128)    0
conv2d_12[0][0]

batch_normalization_12[0][0]
__________________________________________________________________________________________________
re_lu_8 (ReLU)                  (None, 4, 4, 128)    0
add_2[0][0]
__________________________________________________________________________________________________
batch_normalization_13 (BatchNo (None, 4, 4, 128)    512
re_lu_8[0][0]
__________________________________________________________________________________________________
dropout_3 (Dropout)             (None, 4, 4, 128)    0
batch_normalization_13[0][0]
__________________________________________________________________________________________________
flatten_2 (Flatten)             (None, 2048)         0
dropout_3[0][0]
__________________________________________________________________________________________________
dense_3 (Dense)                 (None, 512)          1049088
flatten_2[0][0]
__________________________________________________________________________________________________
```

```
batch_normalization_14 (BatchNo (None, 512)          2048
dense_3[0][0]
_____________________________
dropout_4 (Dropout)             (None, 512)          0
batch_normalization_14[0][0]
_____________________________
dense_4 (Dense)                 (None, 10)           5130
dropout_4[0][0]
==========================================================================
========================
Total params: 1,556,106
Trainable params: 1,553,738
Non-trainable params: 2,368
```



As you can after 6 epoch validation set starts to overfit so we choose epoch value as 6 and retrain the model with new hyperparamter (epoch = 6)

On Test set :
 Accuracy : 0.7583000063896179
 Loss :   0.7259090369224548

Conclusion :
Adding Inception block to our network helped us increase the accuracy. So adding Inception block is useful.