Week 1 Programming Assignment

August 8, 2021

1 Programming assignment

1.1 Transfer learning

1.1.1 Instructions

In this notebook, you will create a neural network model to classify images of cats and dogs, using transfer learning: you will use part of a pre-trained image classifier model (trained on ImageNet) as a feature extractor, and train additional new layers to perform the cats and dogs classification task.

Some code cells are provided you in the notebook. You should avoid editing provided code, and make sure to execute the cells in order to avoid unexpected errors. Some cells begin with the line:

`#### GRADED CELL ####`

Don't move or edit this first line - this is what the automatic grader looks for to recognise graded cells. These cells require you to write your own code to complete them, and are automatically graded when you submit the notebook. Don't edit the function name or signature provided in these cells, otherwise the automatic grader might not function properly. Inside these graded cells, you can use any functions or classes that are imported below, but make sure you don't use any variables that are outside the scope of the function.

1.1.2 How to submit

Complete all the tasks you are asked for in the worksheet. When you have finished and are happy with your code, press the **Submit Assignment** button at the top of this notebook.

1.1.3 Let's get started!

We'll start running some imports, and loading the dataset. Do not edit the existing imports in the following cell. If you would like to make further Tensorflow imports, you should add them here.

In [24]:
```
#### PACKAGE IMPORTS ####

# Run this cell first to import all required packages. Do not make any imports elsewh

import tensorflow as tf
from tensorflow.keras.models import  Sequential, Model
import numpy as np
```

```
import os
import pandas as pd
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns

# If you would like to make further imports from Tensorflow, add them here
from tensorflow.keras.layers import Flatten, Conv2D, AveragePooling2D, MaxPooling2D,
from tensorflow.keras.models import load_model
```

The Dogs vs Cats dataset In this assignment, you will use the Dogs vs Cats dataset, which was used for a 2013 Kaggle competition. It consists of 25000 images containing either a cat or a dog. We will only use a subset of 600 images and labels. The dataset is a subset of a much larger dataset of 3 million photos that were originally used as a CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart), referred to as "Asirra" or Animal Species Image Recognition for Restricting Access.

- J. Elson, J. Douceur, J. Howell, and J. Saul. "Asirra: A CAPTCHA that Exploits Interest-Aligned Manual Image Categorization." Proceedings of 14th ACM Conference on Computer and Communications Security (CCS), October 2007.

Your goal is to train a classifier model using part of a pre-trained image classifier, using the principle of transfer learning.

Load and preprocess the data

```
In [4]: images_train = np.load('data/images_train.npy') / 255.
        images_valid = np.load('data/images_valid.npy') / 255.
        images_test = np.load('data/images_test.npy') / 255.

        labels_train = np.load('data/labels_train.npy')
        labels_valid = np.load('data/labels_valid.npy')
        labels_test = np.load('data/labels_test.npy')
```

```
In [5]: print("{} training data examples".format(images_train.shape[0]))
        print("{} validation data examples".format(images_valid.shape[0]))
        print("{} test data examples".format(images_test.shape[0]))
```

```
600 training data examples
300 validation data examples
300 test data examples
```

Display sample images and labels from the training set

```
In [6]: # Display a few images and labels
```

```
class_names = np.array(['Dog', 'Cat'])

plt.figure(figsize=(15,10))
inx = np.random.choice(images_train.shape[0], 15, replace=False)
for n, i in enumerate(inx):
    ax = plt.subplot(3,5,n+1)
    plt.imshow(images_train[i])
    plt.title(class_names[labels_train[i]])
    plt.axis('off')
```

Create a benchmark model We will first train a CNN classifier model as a benchmark model before implementing the transfer learning approach. Using the functional API, build the benchmark model according to the following specifications:

- The model should use the `input_shape` in the function argument to set the shape in the Input layer.
- The first and second hidden layers should be Conv2D layers with 32 filters, 3x3 kernel size and ReLU activation.
- The third hidden layer should be a MaxPooling2D layer with a 2x2 window size.
- The fourth and fifth hidden layers should be Conv2D layers with 64 filters, 3x3 kernel size and ReLU activation.
- The sixth hidden layer should be a MaxPooling2D layer with a 2x2 window size.
- The seventh and eighth hidden layers should be Conv2D layers with 128 filters, 3x3 kernel size and ReLU activation.

- The ninth hidden layer should be a MaxPooling2D layer with a 2x2 window size.
- This should be followed by a Flatten layer, and a Dense layer with 128 units and ReLU activation
- The final layer should be a Dense layer with a single neuron and sigmoid activation.
- All of the Conv2D layers should use `'SAME'` padding.

In total, the network should have 13 layers (including the `Input` layer).

The model should then be compiled with the RMSProp optimiser with learning rate 0.001, binary cross entropy loss and and binary accuracy metric.

In [34]:
```
#### GRADED CELL ####

# Complete the following function.
# Make sure to not change the function name or arguments.

def get_benchmark_model(input_shape):
    """
    This function should build and compile a CNN model according to the above specifi
    using the functional API. The function takes input_shape as an argument, which sh
    used to specify the shape in the Input layer.
    Your function should return the model.
    """
    inputs = Input(shape=input_shape, name='input')
    h = Conv2D(32, 3, activation='relu', name='conv2d_1', padding='SAME')(inputs)
    h = Conv2D(32, 3, activation='relu', name='conv2d_2', padding='SAME')(h)
    h = MaxPooling2D(2, name='max_pool_1')(h)
    h = Conv2D(64, 3, activation='relu', name='conv2d_3', padding='SAME')(h)
    h = Conv2D(64, 3, activation='relu', name='conv2d_4', padding='SAME')(h)
    h = MaxPooling2D(2, name='max_pool_2')(h)
    h = Conv2D(128, 3, activation='relu', name='conv2d_5', padding='SAME')(h)
    h = Conv2D(128, 3, activation='relu', name='conv2d_6', padding='SAME')(h)
    h = MaxPooling2D(2, name='max_pool_3')(h)
    h = Flatten()(h)
    h = Dense(128, activation='relu', name='dense_1')(h)
    outputs = Dense(1, activation='sigmoid', name='output')(h)
    model = Model(inputs=inputs, outputs=outputs)
    model.compile(optimizer=tf.keras.optimizers.RMSprop(0.001),
              loss = 'binary_crossentropy',
              metrics = ['acc'])
    return(model)
```

In [8]:
```
# Build and compile the benchmark model, and display the model summary

benchmark_model = get_benchmark_model(images_train[0].shape)

benchmark_model.summary()
```

```
Model: "model"
_________________________________________________________________
```

```
Layer (type)                 Output Shape              Param #
=================================================================
input (InputLayer)           [(None, 160, 160, 3)]     0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 160, 160, 32)      896
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 160, 160, 32)      9248
_________________________________________________________________
max_pool_1 (MaxPooling2D)    (None, 80, 80, 32)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 80, 80, 64)        18496
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 80, 80, 64)        36928
_________________________________________________________________
max_pool_2 (MaxPooling2D)    (None, 40, 40, 64)        0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 40, 40, 128)       73856
_________________________________________________________________
conv2d_6 (Conv2D)            (None, 40, 40, 128)       147584
_________________________________________________________________
max_pool_3 (MaxPooling2D)    (None, 20, 20, 128)       0
_________________________________________________________________
flatten (Flatten)            (None, 51200)             0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               6553728
_________________________________________________________________
output (Dense)               (None, 1)                 129
=================================================================
Total params: 6,840,865
Trainable params: 6,840,865
Non-trainable params: 0
_________________________________________________________________
```

Train the CNN benchmark model We will train the benchmark CNN model using an `EarlyStopping` callback. Feel free to increase the training time if you wish.

```
In [9]: # Fit the benchmark model and save its training history
        earlystopping = tf.keras.callbacks.EarlyStopping(patience=2)
        history_benchmark = benchmark_model.fit(images_train, labels_train, epochs=10, batch_si
                                                validation_data=(images_valid, labels_valid),
                                                callbacks=[earlystopping])

Train on 600 samples, validate on 300 samples
Epoch 1/10
600/600 [==============================] - 389s 648ms/sample - loss: 1.7339 - acc: 0.5233 - val
Epoch 2/10
600/600 [==============================] - 400s 666ms/sample - loss: 0.6967 - acc: 0.4817 - val
```

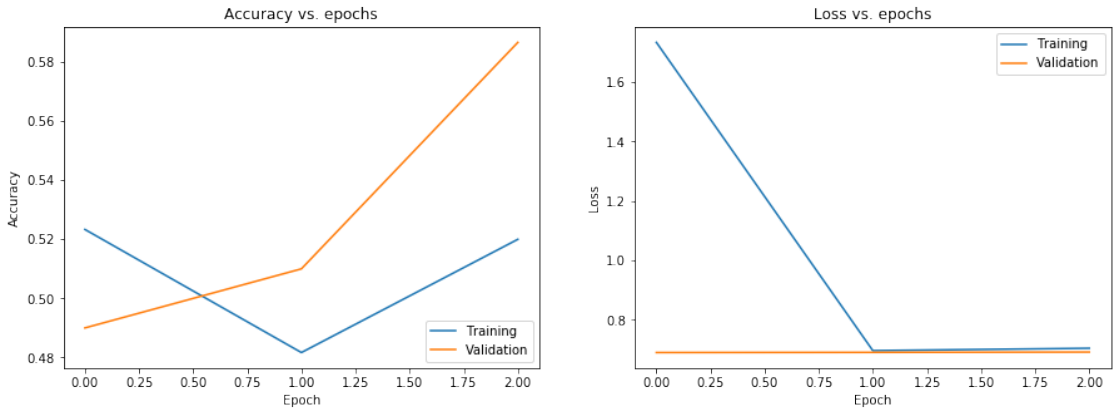
```
Epoch 3/10
600/600 [==============================] - 393s 654ms/sample - loss: 0.7048 - acc: 0.5200 - val
```

Plot the learning curves

```
In [11]: # Run this cell to plot accuracy vs epoch and loss vs epoch
```

```
plt.figure(figsize=(15,5))
plt.subplot(121)
try:
    plt.plot(history_benchmark.history['accuracy'])
    plt.plot(history_benchmark.history['val_accuracy'])
except KeyError:
    plt.plot(history_benchmark.history['acc'])
    plt.plot(history_benchmark.history['val_acc'])
plt.title('Accuracy vs. epochs')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Training', 'Validation'], loc='lower right')

plt.subplot(122)
plt.plot(history_benchmark.history['loss'])
plt.plot(history_benchmark.history['val_loss'])
plt.title('Loss vs. epochs')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Training', 'Validation'], loc='upper right')
plt.show()
```



Evaluate the benchmark model

```
In [12]: # Evaluate the benchmark model on the test set

         benchmark_test_loss, benchmark_test_acc = benchmark_model.evaluate(images_test, labels
         print("Test loss: {}".format(benchmark_test_loss))
         print("Test accuracy: {}".format(benchmark_test_acc))

Test loss: 0.6919893328348795
Test accuracy: 0.5666666626930237
```

Load the pretrained image classifier model You will now begin to build our image classifier using transfer learning. You will use the pre-trained MobileNet V2 model, available to download from Keras Applications. However, we have already downloaded the pretrained model for you, and it is available at the location `./models/MobileNetV2.h5`.

```
In [13]: #### GRADED CELL ####

         # Complete the following function.
         # Make sure to not change the function name or arguments.

         def load_pretrained_MobileNetV2(path):
             """
             This function takes a path as an argument, and uses it to
             load the full MobileNetV2 pretrained model from the path.
             Your function should return the loaded model.
             """
             loaded_model = load_model(path)

             return(loaded_model)


In [14]: # Call the function loading the pretrained model and display its summary

         base_model = load_pretrained_MobileNetV2('models/MobileNetV2.h5')
         base_model.summary()

WARNING:tensorflow:No training configuration found in save file: the model was *not* compiled.
Model: "mobilenetv2_1.00_160"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_6 (InputLayer)            [(None, 160, 160, 3) 0
__________________________________________________________________________________________________
Conv1_pad (ZeroPadding2D)       (None, 161, 161, 3)  0           input_6[0][0]
__________________________________________________________________________________________________
Conv1 (Conv2D)                  (None, 80, 80, 32)   864         Conv1_pad[0][0]
__________________________________________________________________________________________________
bn_Conv1 (BatchNormalization)   (None, 80, 80, 32)   128         Conv1[0][0]
```

Conv1_relu (ReLU)	(None, 80, 80, 32)	0	bn_Conv1[0][0]
expanded_conv_depthwise (Depthw	(None, 80, 80, 32)	288	Conv1_relu[0][0]
expanded_conv_depthwise_BN (Bat	(None, 80, 80, 32)	128	expanded_conv_depthwise[0][0]
expanded_conv_depthwise_relu (R	(None, 80, 80, 32)	0	expanded_conv_depthwise_BN[0]
expanded_conv_project (Conv2D)	(None, 80, 80, 16)	512	expanded_conv_depthwise_relu[
expanded_conv_project_BN (Batch	(None, 80, 80, 16)	64	expanded_conv_project[0][0]
block_1_expand (Conv2D)	(None, 80, 80, 96)	1536	expanded_conv_project_BN[0][0
block_1_expand_BN (BatchNormali	(None, 80, 80, 96)	384	block_1_expand[0][0]
block_1_expand_relu (ReLU)	(None, 80, 80, 96)	0	block_1_expand_BN[0][0]
block_1_pad (ZeroPadding2D)	(None, 81, 81, 96)	0	block_1_expand_relu[0][0]
block_1_depthwise (DepthwiseCon	(None, 40, 40, 96)	864	block_1_pad[0][0]
block_1_depthwise_BN (BatchNorm	(None, 40, 40, 96)	384	block_1_depthwise[0][0]
block_1_depthwise_relu (ReLU)	(None, 40, 40, 96)	0	block_1_depthwise_BN[0][0]
block_1_project (Conv2D)	(None, 40, 40, 24)	2304	block_1_depthwise_relu[0][0]
block_1_project_BN (BatchNormal	(None, 40, 40, 24)	96	block_1_project[0][0]
block_2_expand (Conv2D)	(None, 40, 40, 144)	3456	block_1_project_BN[0][0]
block_2_expand_BN (BatchNormali	(None, 40, 40, 144)	576	block_2_expand[0][0]
block_2_expand_relu (ReLU)	(None, 40, 40, 144)	0	block_2_expand_BN[0][0]
block_2_depthwise (DepthwiseCon	(None, 40, 40, 144)	1296	block_2_expand_relu[0][0]
block_2_depthwise_BN (BatchNorm	(None, 40, 40, 144)	576	block_2_depthwise[0][0]
block_2_depthwise_relu (ReLU)	(None, 40, 40, 144)	0	block_2_depthwise_BN[0][0]
block_2_project (Conv2D)	(None, 40, 40, 24)	3456	block_2_depthwise_relu[0][0]
block_2_project_BN (BatchNormal	(None, 40, 40, 24)	96	block_2_project[0][0]
block_2_add (Add)	(None, 40, 40, 24)	0	block_1_project_BN[0][0]

```
                                                                 block_2_project_BN[0][0]
__________________________________________________________________________________________________
block_3_expand (Conv2D)         (None, 40, 40, 144)  3456        block_2_add[0][0]
__________________________________________________________________________________________________
block_3_expand_BN (BatchNormali (None, 40, 40, 144)  576         block_3_expand[0][0]
__________________________________________________________________________________________________
block_3_expand_relu (ReLU)      (None, 40, 40, 144)  0           block_3_expand_BN[0][0]
__________________________________________________________________________________________________
block_3_pad (ZeroPadding2D)     (None, 41, 41, 144)  0           block_3_expand_relu[0][0]
__________________________________________________________________________________________________
block_3_depthwise (DepthwiseCon (None, 20, 20, 144)  1296        block_3_pad[0][0]
__________________________________________________________________________________________________
block_3_depthwise_BN (BatchNorm (None, 20, 20, 144)  576         block_3_depthwise[0][0]
__________________________________________________________________________________________________
block_3_depthwise_relu (ReLU)   (None, 20, 20, 144)  0           block_3_depthwise_BN[0][0]
__________________________________________________________________________________________________
block_3_project (Conv2D)        (None, 20, 20, 32)   4608        block_3_depthwise_relu[0][0]
__________________________________________________________________________________________________
block_3_project_BN (BatchNormal (None, 20, 20, 32)   128         block_3_project[0][0]
__________________________________________________________________________________________________
block_4_expand (Conv2D)         (None, 20, 20, 192)  6144        block_3_project_BN[0][0]
__________________________________________________________________________________________________
block_4_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_4_expand[0][0]
__________________________________________________________________________________________________
block_4_expand_relu (ReLU)      (None, 20, 20, 192)  0           block_4_expand_BN[0][0]
__________________________________________________________________________________________________
block_4_depthwise (DepthwiseCon (None, 20, 20, 192)  1728        block_4_expand_relu[0][0]
__________________________________________________________________________________________________
block_4_depthwise_BN (BatchNorm (None, 20, 20, 192)  768         block_4_depthwise[0][0]
__________________________________________________________________________________________________
block_4_depthwise_relu (ReLU)   (None, 20, 20, 192)  0           block_4_depthwise_BN[0][0]
__________________________________________________________________________________________________
block_4_project (Conv2D)        (None, 20, 20, 32)   6144        block_4_depthwise_relu[0][0]
__________________________________________________________________________________________________
block_4_project_BN (BatchNormal (None, 20, 20, 32)   128         block_4_project[0][0]
__________________________________________________________________________________________________
block_4_add (Add)               (None, 20, 20, 32)   0           block_3_project_BN[0][0]
                                                                 block_4_project_BN[0][0]
__________________________________________________________________________________________________
block_5_expand (Conv2D)         (None, 20, 20, 192)  6144        block_4_add[0][0]
__________________________________________________________________________________________________
block_5_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_5_expand[0][0]
__________________________________________________________________________________________________
block_5_expand_relu (ReLU)      (None, 20, 20, 192)  0           block_5_expand_BN[0][0]
__________________________________________________________________________________________________
block_5_depthwise (DepthwiseCon (None, 20, 20, 192)  1728        block_5_expand_relu[0][0]
__________________________________________________________________________________________________
block_5_depthwise_BN (BatchNorm (None, 20, 20, 192)  768         block_5_depthwise[0][0]
```

```
block_5_depthwise_relu (ReLU)   (None, 20, 20, 192)  0           block_5_depthwise_BN[0][0]
block_5_project (Conv2D)        (None, 20, 20, 32)   6144        block_5_depthwise_relu[0][0]
block_5_project_BN (BatchNormal (None, 20, 20, 32)   128         block_5_project[0][0]
block_5_add (Add)               (None, 20, 20, 32)   0           block_4_add[0][0]
                                                                 block_5_project_BN[0][0]
block_6_expand (Conv2D)         (None, 20, 20, 192)  6144        block_5_add[0][0]
block_6_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_6_expand[0][0]
block_6_expand_relu (ReLU)      (None, 20, 20, 192)  0           block_6_expand_BN[0][0]
block_6_pad (ZeroPadding2D)     (None, 21, 21, 192)  0           block_6_expand_relu[0][0]
block_6_depthwise (DepthwiseCon (None, 10, 10, 192)  1728        block_6_pad[0][0]
block_6_depthwise_BN (BatchNorm (None, 10, 10, 192)  768         block_6_depthwise[0][0]
block_6_depthwise_relu (ReLU)   (None, 10, 10, 192)  0           block_6_depthwise_BN[0][0]
block_6_project (Conv2D)        (None, 10, 10, 64)   12288       block_6_depthwise_relu[0][0]
block_6_project_BN (BatchNormal (None, 10, 10, 64)   256         block_6_project[0][0]
block_7_expand (Conv2D)         (None, 10, 10, 384)  24576       block_6_project_BN[0][0]
block_7_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_7_expand[0][0]
block_7_expand_relu (ReLU)      (None, 10, 10, 384)  0           block_7_expand_BN[0][0]
block_7_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_7_expand_relu[0][0]
block_7_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_7_depthwise[0][0]
block_7_depthwise_relu (ReLU)   (None, 10, 10, 384)  0           block_7_depthwise_BN[0][0]
block_7_project (Conv2D)        (None, 10, 10, 64)   24576       block_7_depthwise_relu[0][0]
block_7_project_BN (BatchNormal (None, 10, 10, 64)   256         block_7_project[0][0]
block_7_add (Add)               (None, 10, 10, 64)   0           block_6_project_BN[0][0]
                                                                 block_7_project_BN[0][0]
block_8_expand (Conv2D)         (None, 10, 10, 384)  24576       block_7_add[0][0]
```

```
block_8_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_8_expand[0][0]
block_8_expand_relu (ReLU)      (None, 10, 10, 384)  0           block_8_expand_BN[0][0]
block_8_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_8_expand_relu[0][0]
block_8_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_8_depthwise[0][0]
block_8_depthwise_relu (ReLU)   (None, 10, 10, 384)  0           block_8_depthwise_BN[0][0]
block_8_project (Conv2D)        (None, 10, 10, 64)   24576       block_8_depthwise_relu[0][0]
block_8_project_BN (BatchNormal (None, 10, 10, 64)   256         block_8_project[0][0]
block_8_add (Add)               (None, 10, 10, 64)   0           block_7_add[0][0]
                                                                 block_8_project_BN[0][0]
block_9_expand (Conv2D)         (None, 10, 10, 384)  24576       block_8_add[0][0]
block_9_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_9_expand[0][0]
block_9_expand_relu (ReLU)      (None, 10, 10, 384)  0           block_9_expand_BN[0][0]
block_9_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_9_expand_relu[0][0]
block_9_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_9_depthwise[0][0]
block_9_depthwise_relu (ReLU)   (None, 10, 10, 384)  0           block_9_depthwise_BN[0][0]
block_9_project (Conv2D)        (None, 10, 10, 64)   24576       block_9_depthwise_relu[0][0]
block_9_project_BN (BatchNormal (None, 10, 10, 64)   256         block_9_project[0][0]
block_9_add (Add)               (None, 10, 10, 64)   0           block_8_add[0][0]
                                                                 block_9_project_BN[0][0]
block_10_expand (Conv2D)        (None, 10, 10, 384)  24576       block_9_add[0][0]
block_10_expand_BN (BatchNormal (None, 10, 10, 384)  1536        block_10_expand[0][0]
block_10_expand_relu (ReLU)     (None, 10, 10, 384)  0           block_10_expand_BN[0][0]
block_10_depthwise (DepthwiseCo (None, 10, 10, 384)  3456        block_10_expand_relu[0][0]
block_10_depthwise_BN (BatchNor (None, 10, 10, 384)  1536        block_10_depthwise[0][0]
block_10_depthwise_relu (ReLU)  (None, 10, 10, 384)  0           block_10_depthwise_BN[0][0]
```

```
block_10_project (Conv2D)       (None, 10, 10, 96)   36864       block_10_depthwise_relu[0][0]
block_10_project_BN (BatchNorma (None, 10, 10, 96)   384         block_10_project[0][0]
block_11_expand (Conv2D)        (None, 10, 10, 576)  55296       block_10_project_BN[0][0]
block_11_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_11_expand[0][0]
block_11_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_11_expand_BN[0][0]
block_11_depthwise (DepthwiseCo (None, 10, 10, 576)  5184        block_11_expand_relu[0][0]
block_11_depthwise_BN (BatchNor (None, 10, 10, 576)  2304        block_11_depthwise[0][0]
block_11_depthwise_relu (ReLU)  (None, 10, 10, 576)  0           block_11_depthwise_BN[0][0]
block_11_project (Conv2D)       (None, 10, 10, 96)   55296       block_11_depthwise_relu[0][0]
block_11_project_BN (BatchNorma (None, 10, 10, 96)   384         block_11_project[0][0]
block_11_add (Add)              (None, 10, 10, 96)   0           block_10_project_BN[0][0]
                                                                 block_11_project_BN[0][0]
block_12_expand (Conv2D)        (None, 10, 10, 576)  55296       block_11_add[0][0]
block_12_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_12_expand[0][0]
block_12_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_12_expand_BN[0][0]
block_12_depthwise (DepthwiseCo (None, 10, 10, 576)  5184        block_12_expand_relu[0][0]
block_12_depthwise_BN (BatchNor (None, 10, 10, 576)  2304        block_12_depthwise[0][0]
block_12_depthwise_relu (ReLU)  (None, 10, 10, 576)  0           block_12_depthwise_BN[0][0]
block_12_project (Conv2D)       (None, 10, 10, 96)   55296       block_12_depthwise_relu[0][0]
block_12_project_BN (BatchNorma (None, 10, 10, 96)   384         block_12_project[0][0]
block_12_add (Add)              (None, 10, 10, 96)   0           block_11_add[0][0]
                                                                 block_12_project_BN[0][0]
block_13_expand (Conv2D)        (None, 10, 10, 576)  55296       block_12_add[0][0]
block_13_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_13_expand[0][0]
block_13_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_13_expand_BN[0][0]
```

```
block_13_pad (ZeroPadding2D)    (None, 11, 11, 576)  0           block_13_expand_relu[0][0]
block_13_depthwise (DepthwiseCo (None, 5, 5, 576)    5184        block_13_pad[0][0]
block_13_depthwise_BN (BatchNor (None, 5, 5, 576)    2304        block_13_depthwise[0][0]
block_13_depthwise_relu (ReLU)  (None, 5, 5, 576)    0           block_13_depthwise_BN[0][0]
block_13_project (Conv2D)       (None, 5, 5, 160)    92160       block_13_depthwise_relu[0][0]
block_13_project_BN (BatchNorma (None, 5, 5, 160)    640         block_13_project[0][0]
block_14_expand (Conv2D)        (None, 5, 5, 960)    153600      block_13_project_BN[0][0]
block_14_expand_BN (BatchNormal (None, 5, 5, 960)    3840        block_14_expand[0][0]
block_14_expand_relu (ReLU)     (None, 5, 5, 960)    0           block_14_expand_BN[0][0]
block_14_depthwise (DepthwiseCo (None, 5, 5, 960)    8640        block_14_expand_relu[0][0]
block_14_depthwise_BN (BatchNor (None, 5, 5, 960)    3840        block_14_depthwise[0][0]
block_14_depthwise_relu (ReLU)  (None, 5, 5, 960)    0           block_14_depthwise_BN[0][0]
block_14_project (Conv2D)       (None, 5, 5, 160)    153600      block_14_depthwise_relu[0][0]
block_14_project_BN (BatchNorma (None, 5, 5, 160)    640         block_14_project[0][0]
block_14_add (Add)              (None, 5, 5, 160)    0           block_13_project_BN[0][0]
                                                                 block_14_project_BN[0][0]
block_15_expand (Conv2D)        (None, 5, 5, 960)    153600      block_14_add[0][0]
block_15_expand_BN (BatchNormal (None, 5, 5, 960)    3840        block_15_expand[0][0]
block_15_expand_relu (ReLU)     (None, 5, 5, 960)    0           block_15_expand_BN[0][0]
block_15_depthwise (DepthwiseCo (None, 5, 5, 960)    8640        block_15_expand_relu[0][0]
block_15_depthwise_BN (BatchNor (None, 5, 5, 960)    3840        block_15_depthwise[0][0]
block_15_depthwise_relu (ReLU)  (None, 5, 5, 960)    0           block_15_depthwise_BN[0][0]
block_15_project (Conv2D)       (None, 5, 5, 160)    153600      block_15_depthwise_relu[0][0]
block_15_project_BN (BatchNorma (None, 5, 5, 160)    640         block_15_project[0][0]
```

```
block_15_add (Add)              (None, 5, 5, 160)    0           block_14_add[0][0]
                                                                 block_15_project_BN[0][0]
__________________________________________________________________________________________
block_16_expand (Conv2D)        (None, 5, 5, 960)    153600      block_15_add[0][0]
__________________________________________________________________________________________
block_16_expand_BN (BatchNormal (None, 5, 5, 960)    3840        block_16_expand[0][0]
__________________________________________________________________________________________
block_16_expand_relu (ReLU)     (None, 5, 5, 960)    0           block_16_expand_BN[0][0]
__________________________________________________________________________________________
block_16_depthwise (DepthwiseCo (None, 5, 5, 960)    8640        block_16_expand_relu[0][0]
__________________________________________________________________________________________
block_16_depthwise_BN (BatchNor (None, 5, 5, 960)    3840        block_16_depthwise[0][0]
__________________________________________________________________________________________
block_16_depthwise_relu (ReLU)  (None, 5, 5, 960)    0           block_16_depthwise_BN[0][0]
__________________________________________________________________________________________
block_16_project (Conv2D)       (None, 5, 5, 320)    307200      block_16_depthwise_relu[0][0]
__________________________________________________________________________________________
block_16_project_BN (BatchNorma (None, 5, 5, 320)    1280        block_16_project[0][0]
__________________________________________________________________________________________
Conv_1 (Conv2D)                 (None, 5, 5, 1280)   409600      block_16_project_BN[0][0]
__________________________________________________________________________________________
Conv_1_bn (BatchNormalization)  (None, 5, 5, 1280)   5120        Conv_1[0][0]
__________________________________________________________________________________________
out_relu (ReLU)                 (None, 5, 5, 1280)   0           Conv_1_bn[0][0]
__________________________________________________________________________________________
global_average_pooling2d_6 (Glo (None, 1280)         0           out_relu[0][0]
__________________________________________________________________________________________
Logits (Dense)                  (None, 1000)         1281000     global_average_pooling2d_6[0]
==========================================================================================
Total params: 3,538,984
Trainable params: 3,504,872
Non-trainable params: 34,112
__________________________________________________________________________________________
```

Use the pre-trained model as a feature extractor You will remove the final layer of the network and replace it with new, untrained classifier layers for our task. You will first create a new model that has the same input tensor as the MobileNetV2 model, and uses the output tensor from the layer with name `global_average_pooling2d_6` as the model output.

In [35]:
```
#### GRADED CELL ####

# Complete the following function.
# Make sure to not change the function name or arguments.

def remove_head(pretrained_model):
    """
    This function should create and return a new model, using the input and output
```

```
        tensors as specified above.
        Use the 'get_layer' method to access the correct layer of the pre-trained model.
        """
        new_model_output = pretrained_model.get_layer('global_average_pooling2d_6').output
        new_model = Model(inputs=pretrained_model.input, outputs=new_model_output)

        return(new_model)
```

In [20]:
```
# Call the function removing the classification head and display the summary

feature_extractor = remove_head(base_model)
feature_extractor.summary()
```

```
Model: "model_1"
_________________________________________________________________________________________________
Layer (type)                 Output Shape              Param #     Connected to
=================================================================================================
input_6 (InputLayer)         [(None, 160, 160, 3)      0
_________________________________________________________________________________________________
Conv1_pad (ZeroPadding2D)    (None, 161, 161, 3)       0           input_6[0][0]
_________________________________________________________________________________________________
Conv1 (Conv2D)               (None, 80, 80, 32)        864         Conv1_pad[0][0]
_________________________________________________________________________________________________
bn_Conv1 (BatchNormalization)(None, 80, 80, 32)        128         Conv1[0][0]
_________________________________________________________________________________________________
Conv1_relu (ReLU)            (None, 80, 80, 32)        0           bn_Conv1[0][0]
_________________________________________________________________________________________________
expanded_conv_depthwise (Depthw (None, 80, 80, 32)     288         Conv1_relu[0][0]
_________________________________________________________________________________________________
expanded_conv_depthwise_BN (Bat (None, 80, 80, 32)     128         expanded_conv_depthwise[0][0]
_________________________________________________________________________________________________
expanded_conv_depthwise_relu (R (None, 80, 80, 32)     0           expanded_conv_depthwise_BN[0]
_________________________________________________________________________________________________
expanded_conv_project (Conv2D)  (None, 80, 80, 16)     512         expanded_conv_depthwise_relu[
_________________________________________________________________________________________________
expanded_conv_project_BN (Batch (None, 80, 80, 16)     64          expanded_conv_project[0][0]
_________________________________________________________________________________________________
block_1_expand (Conv2D)      (None, 80, 80, 96)        1536        expanded_conv_project_BN[0][0
_________________________________________________________________________________________________
block_1_expand_BN (BatchNormali (None, 80, 80, 96)     384         block_1_expand[0][0]
_________________________________________________________________________________________________
block_1_expand_relu (ReLU)   (None, 80, 80, 96)        0           block_1_expand_BN[0][0]
_________________________________________________________________________________________________
block_1_pad (ZeroPadding2D)  (None, 81, 81, 96)        0           block_1_expand_relu[0][0]
_________________________________________________________________________________________________
block_1_depthwise (DepthwiseCon (None, 40, 40, 96)     864         block_1_pad[0][0]
_________________________________________________________________________________________________
block_1_depthwise_BN (BatchNorm (None, 40, 40, 96)     384         block_1_depthwise[0][0]
```

```
block_1_depthwise_relu (ReLU)   (None, 40, 40, 96)   0           block_1_depthwise_BN[0][0]
block_1_project (Conv2D)        (None, 40, 40, 24)   2304        block_1_depthwise_relu[0][0]
block_1_project_BN (BatchNormal (None, 40, 40, 24)   96          block_1_project[0][0]
block_2_expand (Conv2D)         (None, 40, 40, 144)  3456        block_1_project_BN[0][0]
block_2_expand_BN (BatchNormali (None, 40, 40, 144)  576         block_2_expand[0][0]
block_2_expand_relu (ReLU)      (None, 40, 40, 144)  0           block_2_expand_BN[0][0]
block_2_depthwise (DepthwiseCon (None, 40, 40, 144)  1296        block_2_expand_relu[0][0]
block_2_depthwise_BN (BatchNorm (None, 40, 40, 144)  576         block_2_depthwise[0][0]
block_2_depthwise_relu (ReLU)   (None, 40, 40, 144)  0           block_2_depthwise_BN[0][0]
block_2_project (Conv2D)        (None, 40, 40, 24)   3456        block_2_depthwise_relu[0][0]
block_2_project_BN (BatchNormal (None, 40, 40, 24)   96          block_2_project[0][0]
block_2_add (Add)               (None, 40, 40, 24)   0           block_1_project_BN[0][0]
                                                                 block_2_project_BN[0][0]
block_3_expand (Conv2D)         (None, 40, 40, 144)  3456        block_2_add[0][0]
block_3_expand_BN (BatchNormali (None, 40, 40, 144)  576         block_3_expand[0][0]
block_3_expand_relu (ReLU)      (None, 40, 40, 144)  0           block_3_expand_BN[0][0]
block_3_pad (ZeroPadding2D)     (None, 41, 41, 144)  0           block_3_expand_relu[0][0]
block_3_depthwise (DepthwiseCon (None, 20, 20, 144)  1296        block_3_pad[0][0]
block_3_depthwise_BN (BatchNorm (None, 20, 20, 144)  576         block_3_depthwise[0][0]
block_3_depthwise_relu (ReLU)   (None, 20, 20, 144)  0           block_3_depthwise_BN[0][0]
block_3_project (Conv2D)        (None, 20, 20, 32)   4608        block_3_depthwise_relu[0][0]
block_3_project_BN (BatchNormal (None, 20, 20, 32)   128         block_3_project[0][0]
block_4_expand (Conv2D)         (None, 20, 20, 192)  6144        block_3_project_BN[0][0]
block_4_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_4_expand[0][0]
```

```
block_4_expand_relu (ReLU)      (None, 20, 20, 192)  0           block_4_expand_BN[0][0]
__________________________________________________________________________________________
block_4_depthwise (DepthwiseCon (None, 20, 20, 192)  1728        block_4_expand_relu[0][0]
__________________________________________________________________________________________
block_4_depthwise_BN (BatchNorm (None, 20, 20, 192)  768         block_4_depthwise[0][0]
__________________________________________________________________________________________
block_4_depthwise_relu (ReLU)   (None, 20, 20, 192)  0           block_4_depthwise_BN[0][0]
__________________________________________________________________________________________
block_4_project (Conv2D)        (None, 20, 20, 32)   6144        block_4_depthwise_relu[0][0]
__________________________________________________________________________________________
block_4_project_BN (BatchNormal (None, 20, 20, 32)   128         block_4_project[0][0]
__________________________________________________________________________________________
block_4_add (Add)               (None, 20, 20, 32)   0           block_3_project_BN[0][0]
                                                                 block_4_project_BN[0][0]
__________________________________________________________________________________________
block_5_expand (Conv2D)         (None, 20, 20, 192)  6144        block_4_add[0][0]
__________________________________________________________________________________________
block_5_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_5_expand[0][0]
__________________________________________________________________________________________
block_5_expand_relu (ReLU)      (None, 20, 20, 192)  0           block_5_expand_BN[0][0]
__________________________________________________________________________________________
block_5_depthwise (DepthwiseCon (None, 20, 20, 192)  1728        block_5_expand_relu[0][0]
__________________________________________________________________________________________
block_5_depthwise_BN (BatchNorm (None, 20, 20, 192)  768         block_5_depthwise[0][0]
__________________________________________________________________________________________
block_5_depthwise_relu (ReLU)   (None, 20, 20, 192)  0           block_5_depthwise_BN[0][0]
__________________________________________________________________________________________
block_5_project (Conv2D)        (None, 20, 20, 32)   6144        block_5_depthwise_relu[0][0]
__________________________________________________________________________________________
block_5_project_BN (BatchNormal (None, 20, 20, 32)   128         block_5_project[0][0]
__________________________________________________________________________________________
block_5_add (Add)               (None, 20, 20, 32)   0           block_4_add[0][0]
                                                                 block_5_project_BN[0][0]
__________________________________________________________________________________________
block_6_expand (Conv2D)         (None, 20, 20, 192)  6144        block_5_add[0][0]
__________________________________________________________________________________________
block_6_expand_BN (BatchNormali (None, 20, 20, 192)  768         block_6_expand[0][0]
__________________________________________________________________________________________
block_6_expand_relu (ReLU)      (None, 20, 20, 192)  0           block_6_expand_BN[0][0]
__________________________________________________________________________________________
block_6_pad (ZeroPadding2D)     (None, 21, 21, 192)  0           block_6_expand_relu[0][0]
__________________________________________________________________________________________
block_6_depthwise (DepthwiseCon (None, 10, 10, 192)  1728        block_6_pad[0][0]
__________________________________________________________________________________________
block_6_depthwise_BN (BatchNorm (None, 10, 10, 192)  768         block_6_depthwise[0][0]
__________________________________________________________________________________________
block_6_depthwise_relu (ReLU)   (None, 10, 10, 192)  0           block_6_depthwise_BN[0][0]
__________________________________________________________________________________________
```

```
block_6_project (Conv2D)        (None, 10, 10, 64)   12288       block_6_depthwise_relu[0][0]
--------------------------------------------------------------------------------------------------
block_6_project_BN (BatchNormal (None, 10, 10, 64)   256         block_6_project[0][0]
--------------------------------------------------------------------------------------------------
block_7_expand (Conv2D)         (None, 10, 10, 384)  24576       block_6_project_BN[0][0]
--------------------------------------------------------------------------------------------------
block_7_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_7_expand[0][0]
--------------------------------------------------------------------------------------------------
block_7_expand_relu (ReLU)      (None, 10, 10, 384)  0           block_7_expand_BN[0][0]
--------------------------------------------------------------------------------------------------
block_7_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_7_expand_relu[0][0]
--------------------------------------------------------------------------------------------------
block_7_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_7_depthwise[0][0]
--------------------------------------------------------------------------------------------------
block_7_depthwise_relu (ReLU)   (None, 10, 10, 384)  0           block_7_depthwise_BN[0][0]
--------------------------------------------------------------------------------------------------
block_7_project (Conv2D)        (None, 10, 10, 64)   24576       block_7_depthwise_relu[0][0]
--------------------------------------------------------------------------------------------------
block_7_project_BN (BatchNormal (None, 10, 10, 64)   256         block_7_project[0][0]
--------------------------------------------------------------------------------------------------
block_7_add (Add)               (None, 10, 10, 64)   0           block_6_project_BN[0][0]
                                                                 block_7_project_BN[0][0]
--------------------------------------------------------------------------------------------------
block_8_expand (Conv2D)         (None, 10, 10, 384)  24576       block_7_add[0][0]
--------------------------------------------------------------------------------------------------
block_8_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_8_expand[0][0]
--------------------------------------------------------------------------------------------------
block_8_expand_relu (ReLU)      (None, 10, 10, 384)  0           block_8_expand_BN[0][0]
--------------------------------------------------------------------------------------------------
block_8_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_8_expand_relu[0][0]
--------------------------------------------------------------------------------------------------
block_8_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_8_depthwise[0][0]
--------------------------------------------------------------------------------------------------
block_8_depthwise_relu (ReLU)   (None, 10, 10, 384)  0           block_8_depthwise_BN[0][0]
--------------------------------------------------------------------------------------------------
block_8_project (Conv2D)        (None, 10, 10, 64)   24576       block_8_depthwise_relu[0][0]
--------------------------------------------------------------------------------------------------
block_8_project_BN (BatchNormal (None, 10, 10, 64)   256         block_8_project[0][0]
--------------------------------------------------------------------------------------------------
block_8_add (Add)               (None, 10, 10, 64)   0           block_7_add[0][0]
                                                                 block_8_project_BN[0][0]
--------------------------------------------------------------------------------------------------
block_9_expand (Conv2D)         (None, 10, 10, 384)  24576       block_8_add[0][0]
--------------------------------------------------------------------------------------------------
block_9_expand_BN (BatchNormali (None, 10, 10, 384)  1536        block_9_expand[0][0]
--------------------------------------------------------------------------------------------------
block_9_expand_relu (ReLU)      (None, 10, 10, 384)  0           block_9_expand_BN[0][0]
--------------------------------------------------------------------------------------------------
```

```
block_9_depthwise (DepthwiseCon (None, 10, 10, 384)  3456        block_9_expand_relu[0][0]
__________________________________________________________________________________________
block_9_depthwise_BN (BatchNorm (None, 10, 10, 384)  1536        block_9_depthwise[0][0]
__________________________________________________________________________________________
block_9_depthwise_relu (ReLU)   (None, 10, 10, 384)  0           block_9_depthwise_BN[0][0]
__________________________________________________________________________________________
block_9_project (Conv2D)        (None, 10, 10, 64)   24576       block_9_depthwise_relu[0][0]
__________________________________________________________________________________________
block_9_project_BN (BatchNormal (None, 10, 10, 64)   256         block_9_project[0][0]
__________________________________________________________________________________________
block_9_add (Add)               (None, 10, 10, 64)   0           block_8_add[0][0]
                                                                 block_9_project_BN[0][0]
__________________________________________________________________________________________
block_10_expand (Conv2D)        (None, 10, 10, 384)  24576       block_9_add[0][0]
__________________________________________________________________________________________
block_10_expand_BN (BatchNormal (None, 10, 10, 384)  1536        block_10_expand[0][0]
__________________________________________________________________________________________
block_10_expand_relu (ReLU)     (None, 10, 10, 384)  0           block_10_expand_BN[0][0]
__________________________________________________________________________________________
block_10_depthwise (DepthwiseCo (None, 10, 10, 384)  3456        block_10_expand_relu[0][0]
__________________________________________________________________________________________
block_10_depthwise_BN (BatchNor (None, 10, 10, 384)  1536        block_10_depthwise[0][0]
__________________________________________________________________________________________
block_10_depthwise_relu (ReLU)  (None, 10, 10, 384)  0           block_10_depthwise_BN[0][0]
__________________________________________________________________________________________
block_10_project (Conv2D)       (None, 10, 10, 96)   36864       block_10_depthwise_relu[0][0]
__________________________________________________________________________________________
block_10_project_BN (BatchNorma (None, 10, 10, 96)   384         block_10_project[0][0]
__________________________________________________________________________________________
block_11_expand (Conv2D)        (None, 10, 10, 576)  55296       block_10_project_BN[0][0]
__________________________________________________________________________________________
block_11_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_11_expand[0][0]
__________________________________________________________________________________________
block_11_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_11_expand_BN[0][0]
__________________________________________________________________________________________
block_11_depthwise (DepthwiseCo (None, 10, 10, 576)  5184        block_11_expand_relu[0][0]
__________________________________________________________________________________________
block_11_depthwise_BN (BatchNor (None, 10, 10, 576)  2304        block_11_depthwise[0][0]
__________________________________________________________________________________________
block_11_depthwise_relu (ReLU)  (None, 10, 10, 576)  0           block_11_depthwise_BN[0][0]
__________________________________________________________________________________________
block_11_project (Conv2D)       (None, 10, 10, 96)   55296       block_11_depthwise_relu[0][0]
__________________________________________________________________________________________
block_11_project_BN (BatchNorma (None, 10, 10, 96)   384         block_11_project[0][0]
__________________________________________________________________________________________
block_11_add (Add)              (None, 10, 10, 96)   0           block_10_project_BN[0][0]
                                                                 block_11_project_BN[0][0]
__________________________________________________________________________________________
```

```
block_12_expand (Conv2D)        (None, 10, 10, 576)  55296       block_11_add[0][0]
--------------------------------------------------------------------------------------------------
block_12_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_12_expand[0][0]
--------------------------------------------------------------------------------------------------
block_12_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_12_expand_BN[0][0]
--------------------------------------------------------------------------------------------------
block_12_depthwise (DepthwiseCo (None, 10, 10, 576)  5184        block_12_expand_relu[0][0]
--------------------------------------------------------------------------------------------------
block_12_depthwise_BN (BatchNor (None, 10, 10, 576)  2304        block_12_depthwise[0][0]
--------------------------------------------------------------------------------------------------
block_12_depthwise_relu (ReLU)  (None, 10, 10, 576)  0           block_12_depthwise_BN[0][0]
--------------------------------------------------------------------------------------------------
block_12_project (Conv2D)       (None, 10, 10, 96)   55296       block_12_depthwise_relu[0][0]
--------------------------------------------------------------------------------------------------
block_12_project_BN (BatchNorma (None, 10, 10, 96)   384         block_12_project[0][0]
--------------------------------------------------------------------------------------------------
block_12_add (Add)              (None, 10, 10, 96)   0           block_11_add[0][0]
                                                                 block_12_project_BN[0][0]
--------------------------------------------------------------------------------------------------
block_13_expand (Conv2D)        (None, 10, 10, 576)  55296       block_12_add[0][0]
--------------------------------------------------------------------------------------------------
block_13_expand_BN (BatchNormal (None, 10, 10, 576)  2304        block_13_expand[0][0]
--------------------------------------------------------------------------------------------------
block_13_expand_relu (ReLU)     (None, 10, 10, 576)  0           block_13_expand_BN[0][0]
--------------------------------------------------------------------------------------------------
block_13_pad (ZeroPadding2D)    (None, 11, 11, 576)  0           block_13_expand_relu[0][0]
--------------------------------------------------------------------------------------------------
block_13_depthwise (DepthwiseCo (None, 5, 5, 576)    5184        block_13_pad[0][0]
--------------------------------------------------------------------------------------------------
block_13_depthwise_BN (BatchNor (None, 5, 5, 576)    2304        block_13_depthwise[0][0]
--------------------------------------------------------------------------------------------------
block_13_depthwise_relu (ReLU)  (None, 5, 5, 576)    0           block_13_depthwise_BN[0][0]
--------------------------------------------------------------------------------------------------
block_13_project (Conv2D)       (None, 5, 5, 160)    92160       block_13_depthwise_relu[0][0]
--------------------------------------------------------------------------------------------------
block_13_project_BN (BatchNorma (None, 5, 5, 160)    640         block_13_project[0][0]
--------------------------------------------------------------------------------------------------
block_14_expand (Conv2D)        (None, 5, 5, 960)    153600      block_13_project_BN[0][0]
--------------------------------------------------------------------------------------------------
block_14_expand_BN (BatchNormal (None, 5, 5, 960)    3840        block_14_expand[0][0]
--------------------------------------------------------------------------------------------------
block_14_expand_relu (ReLU)     (None, 5, 5, 960)    0           block_14_expand_BN[0][0]
--------------------------------------------------------------------------------------------------
block_14_depthwise (DepthwiseCo (None, 5, 5, 960)    8640        block_14_expand_relu[0][0]
--------------------------------------------------------------------------------------------------
block_14_depthwise_BN (BatchNor (None, 5, 5, 960)    3840        block_14_depthwise[0][0]
--------------------------------------------------------------------------------------------------
block_14_depthwise_relu (ReLU)  (None, 5, 5, 960)    0           block_14_depthwise_BN[0][0]
```

block_14_project (Conv2D)	(None, 5, 5, 160)	153600	block_14_depthwise_relu[0][0]
block_14_project_BN (BatchNorma	(None, 5, 5, 160)	640	block_14_project[0][0]
block_14_add (Add)	(None, 5, 5, 160)	0	block_13_project_BN[0][0] block_14_project_BN[0][0]
block_15_expand (Conv2D)	(None, 5, 5, 960)	153600	block_14_add[0][0]
block_15_expand_BN (BatchNormal	(None, 5, 5, 960)	3840	block_15_expand[0][0]
block_15_expand_relu (ReLU)	(None, 5, 5, 960)	0	block_15_expand_BN[0][0]
block_15_depthwise (DepthwiseCo	(None, 5, 5, 960)	8640	block_15_expand_relu[0][0]
block_15_depthwise_BN (BatchNor	(None, 5, 5, 960)	3840	block_15_depthwise[0][0]
block_15_depthwise_relu (ReLU)	(None, 5, 5, 960)	0	block_15_depthwise_BN[0][0]
block_15_project (Conv2D)	(None, 5, 5, 160)	153600	block_15_depthwise_relu[0][0]
block_15_project_BN (BatchNorma	(None, 5, 5, 160)	640	block_15_project[0][0]
block_15_add (Add)	(None, 5, 5, 160)	0	block_14_add[0][0] block_15_project_BN[0][0]
block_16_expand (Conv2D)	(None, 5, 5, 960)	153600	block_15_add[0][0]
block_16_expand_BN (BatchNormal	(None, 5, 5, 960)	3840	block_16_expand[0][0]
block_16_expand_relu (ReLU)	(None, 5, 5, 960)	0	block_16_expand_BN[0][0]
block_16_depthwise (DepthwiseCo	(None, 5, 5, 960)	8640	block_16_expand_relu[0][0]
block_16_depthwise_BN (BatchNor	(None, 5, 5, 960)	3840	block_16_depthwise[0][0]
block_16_depthwise_relu (ReLU)	(None, 5, 5, 960)	0	block_16_depthwise_BN[0][0]
block_16_project (Conv2D)	(None, 5, 5, 320)	307200	block_16_depthwise_relu[0][0]
block_16_project_BN (BatchNorma	(None, 5, 5, 320)	1280	block_16_project[0][0]
Conv_1 (Conv2D)	(None, 5, 5, 1280)	409600	block_16_project_BN[0][0]
Conv_1_bn (BatchNormalization)	(None, 5, 5, 1280)	5120	Conv_1[0][0]
out_relu (ReLU)	(None, 5, 5, 1280)	0	Conv_1_bn[0][0]

```
__________________________________________________________________________________________________
global_average_pooling2d_6 (Glo (None, 1280)         0           out_relu[0][0]
==================================================================================================
Total params: 2,257,984
Trainable params: 2,223,872
Non-trainable params: 34,112
__________________________________________________________________________________________________
```

You can now construct new final classifier layers for your model. Using the Sequential API, create a new model according to the following specifications:

- The new model should begin with the feature extractor model.
- This should then be followed with a new dense layer with 32 units and ReLU activation function.
- This should be followed by a dropout layer with a rate of 0.5.
- Finally, this should be followed by a Dense layer with a single neuron and a sigmoid activation function.

In total, the network should be composed of the pretrained base model plus 3 layers.

In [22]:
```
#### GRADED CELL ####

# Complete the following function.
# Make sure to not change the function name or arguments.

def add_new_classifier_head(feature_extractor_model):
    """
    This function takes the feature extractor model as an argument, and should create
    and return a new model according to the above specification.
    """
    model = Sequential([
        feature_extractor_model,
        Dense(32, activation='relu'),
        Dropout(0.5),
        Dense(1, activation='sigmoid')
    ])
    return(model)
```

In [25]:
```
# Call the function adding a new classification head and display the summary

new_model = add_new_classifier_head(feature_extractor)
new_model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
model_1 (Model)              (None, 1280)              2257984
```

```
_________________________________________________________________
dense_1 (Dense)              (None, 32)                40992
_________________________________________________________________
dropout (Dropout)            (None, 32)                0
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 33
=================================================================
Total params: 2,299,009
Trainable params: 2,264,897
Non-trainable params: 34,112
_________________________________________________________________
```

Freeze the weights of the pretrained model You will now need to freeze the weights of the pretrained feature extractor, so that only the weights of the new layers you have added will change during the training.

You should then compile your model as before: use the RMSProp optimiser with learning rate 0.001, binary cross entropy loss and and binary accuracy metric.

In [26]:
```
#### GRADED CELL ####

# Complete the following function.
# Make sure to not change the function name or arguments.

def freeze_pretrained_weights(model):
    """
    This function should freeze the weights of the pretrained base model.
    Your function should return the model with frozen weights.
    """
    model.layers[0].trainable = False
    model.compile(optimizer=tf.keras.optimizers.RMSprop(0.001),
         loss = 'binary_crossentropy',
         metrics = ['acc'])
    return(model)
```

In [27]:
```
# Call the function freezing the pretrained weights and display the summary

frozen_new_model = freeze_pretrained_weights(new_model)
frozen_new_model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
model_1 (Model)              (None, 1280)              2257984
_________________________________________________________________
dense_1 (Dense)              (None, 32)                40992
_________________________________________________________________
```

```
dropout (Dropout)            (None, 32)                0
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 33
=================================================================
Total params: 2,299,009
Trainable params: 41,025
Non-trainable params: 2,257,984
_________________________________________________________________
```

Train the model You are now ready to train the new model on the dogs vs cats data subset. We will use an `EarlyStopping` callback with patience set to 2 epochs, as before. Feel free to increase the training time if you wish.

```
In [28]: # Train the model and save its training history

         earlystopping = tf.keras.callbacks.EarlyStopping(patience=2)
         history_frozen_new_model = frozen_new_model.fit(images_train, labels_train, epochs=10
                                                         validation_data=(images_valid, labels_
                                                         callbacks=[earlystopping])
```

```
Train on 600 samples, validate on 300 samples
Epoch 1/10
600/600 [==============================] - 163s 272ms/sample - loss: 0.4842 - acc: 0.7717 - val
Epoch 2/10
600/600 [==============================] - 155s 258ms/sample - loss: 0.2986 - acc: 0.8750 - val
Epoch 3/10
600/600 [==============================] - 154s 257ms/sample - loss: 0.2194 - acc: 0.9100 - val
```

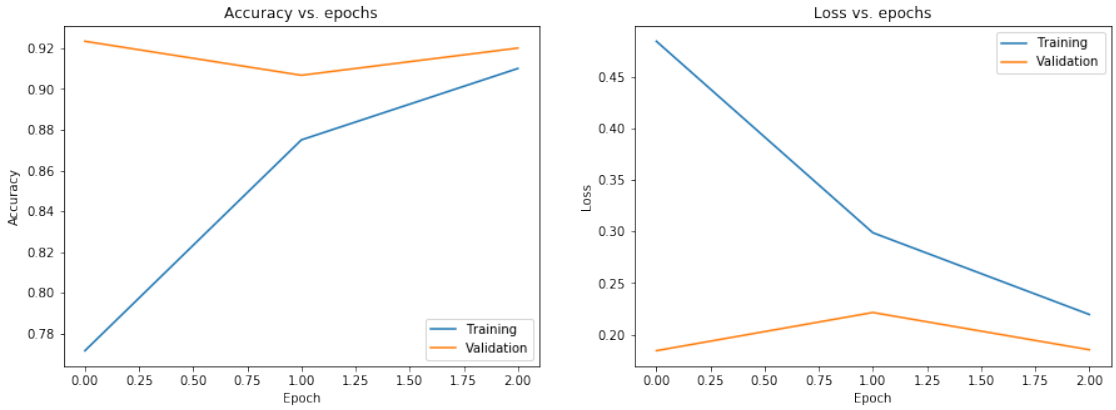
Plot the learning curves

```
In [29]: # Run this cell to plot accuracy vs epoch and loss vs epoch

         plt.figure(figsize=(15,5))
         plt.subplot(121)
         try:
             plt.plot(history_frozen_new_model.history['accuracy'])
             plt.plot(history_frozen_new_model.history['val_accuracy'])
         except KeyError:
             plt.plot(history_frozen_new_model.history['acc'])
             plt.plot(history_frozen_new_model.history['val_acc'])
         plt.title('Accuracy vs. epochs')
         plt.ylabel('Accuracy')
         plt.xlabel('Epoch')
         plt.legend(['Training', 'Validation'], loc='lower right')

         plt.subplot(122)
```

```
plt.plot(history_frozen_new_model.history['loss'])
plt.plot(history_frozen_new_model.history['val_loss'])
plt.title('Loss vs. epochs')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Training', 'Validation'], loc='upper right')
plt.show()
```



Evaluate the new model

```
In [30]: # Evaluate the benchmark model on the test set

         new_model_test_loss, new_model_test_acc = frozen_new_model.evaluate(images_test, label
         print("Test loss: {}".format(new_model_test_loss))
         print("Test accuracy: {}".format(new_model_test_acc))
```

```
Test loss: 0.18046210984388988
Test accuracy: 0.9266666769981384
```

Compare both models Finally, we will look at the comparison of training, validation and test metrics between the benchmark and transfer learning model.

```
In [31]: # Gather the benchmark and new model metrics

         benchmark_train_loss = history_benchmark.history['loss'][-1]
         benchmark_valid_loss = history_benchmark.history['val_loss'][-1]

         try:
             benchmark_train_acc = history_benchmark.history['acc'][-1]
             benchmark_valid_acc = history_benchmark.history['val_acc'][-1]
         except KeyError:
             benchmark_train_acc = history_benchmark.history['accuracy'][-1]
```

```
        benchmark_valid_acc = history_benchmark.history['val_accuracy'][-1]

    new_model_train_loss = history_frozen_new_model.history['loss'][-1]
    new_model_valid_loss = history_frozen_new_model.history['val_loss'][-1]

    try:
        new_model_train_acc = history_frozen_new_model.history['acc'][-1]
        new_model_valid_acc = history_frozen_new_model.history['val_acc'][-1]
    except KeyError:
        new_model_train_acc = history_frozen_new_model.history['accuracy'][-1]
        new_model_valid_acc = history_frozen_new_model.history['val_accuracy'][-1]
```

In [32]:
```
# Compile the metrics into a pandas DataFrame and display the table

comparison_table = pd.DataFrame([['Training loss', benchmark_train_loss, new_model_tra
                                  ['Training accuracy', benchmark_train_acc, new_model_t
                                  ['Validation loss', benchmark_valid_loss, new_model_va
                                  ['Validation accuracy', benchmark_valid_acc, new_model
                                  ['Test loss', benchmark_test_loss, new_model_test_loss
                                  ['Test accuracy', benchmark_test_acc, new_model_test_a
                                 columns=['Metric', 'Benchmark CNN', 'Transfer learning
comparison_table.index=['']*6
comparison_table
```

Out[32]:

Metric	Benchmark CNN	Transfer learning CNN
Training loss	0.704751	0.219391
Training accuracy	0.520000	0.910000
Validation loss	0.691972	0.185214
Validation accuracy	0.586667	0.920000
Test loss	0.691989	0.180462
Test accuracy	0.566667	0.926667

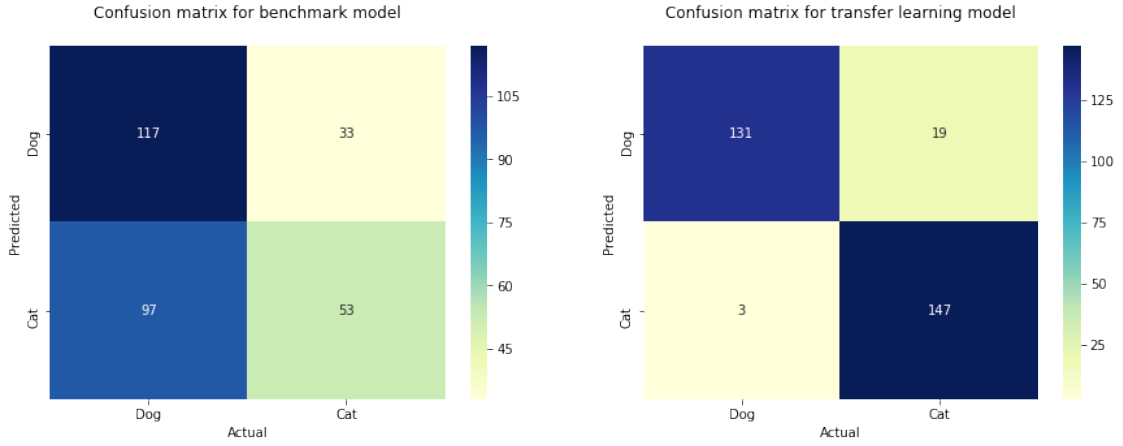
In [33]:
```
# Plot confusion matrices for benchmark and transfer learning models

plt.figure(figsize=(15, 5))

preds = benchmark_model.predict(images_test)
preds = (preds >= 0.5).astype(np.int32)
cm = confusion_matrix(labels_test, preds)
df_cm = pd.DataFrame(cm, index=['Dog', 'Cat'], columns=['Dog', 'Cat'])
plt.subplot(121)
plt.title("Confusion matrix for benchmark model\n")
sns.heatmap(df_cm, annot=True, fmt="d", cmap="YlGnBu")
plt.ylabel("Predicted")
plt.xlabel("Actual")

preds = frozen_new_model.predict(images_test)
preds = (preds >= 0.5).astype(np.int32)
```

```
cm = confusion_matrix(labels_test, preds)
df_cm = pd.DataFrame(cm, index=['Dog', 'Cat'], columns=['Dog', 'Cat'])
plt.subplot(122)
plt.title("Confusion matrix for transfer learning model\n")
sns.heatmap(df_cm, annot=True, fmt="d", cmap="YlGnBu")
plt.ylabel("Predicted")
plt.xlabel("Actual")
plt.show()
```



Congratulations for completing this programming assignment! In the next week of the course we will learn how to develop an effective data pipeline.

In [ ]: