Identificacion de Perros y Gatos

Con uso de Tensorflow

In [0]:

```
from __future__ import absolute_import, division, print_function, unicode_literals
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Conv2D, Flatten, Dropout, MaxPooling2D
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import os
import numpy as np
import matplotlib.pyplot as plt
```

In [0]:

```
#Actualizacion Tensorflow
import tensorflow.compat.v1 as tf
tf.disable_v2_behavior()
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tenso
rflow_core/python/compat/v2_compat.py:68: disable_resource_variables
(from tensorflow.python.ops.variable_scope) is deprecated and will b
e removed in a future version.
Instructions for updating:
non-resource variables are not supported in the long term
```

In [0]:

```
#Cargar datos
_URL = 'https://storage.googleapis.com/mledu-datasets/cats_and_dogs_filtered.zip'
path_to_zip = tf.keras.utils.get_file('cats_and_dogs.zip', origin=_URL, extract=True)
PATH = os.path.join(os.path.dirname(path_to_zip), 'cats_and_dogs_filtered')
```

```
Downloading data from https://storage.googleapis.com/mledu-datasets/
cats_and_dogs_filtered.zip
68608000/68606236 [==============================] - 0s 0us/step
```

In [0]:

```
train_dir = os.path.join(PATH, 'train')
validation_dir = os.path.join(PATH, 'validation')
train_cats_dir = os.path.join(train_dir, 'cats')  # directory with our training cat pictures
train_dogs_dir = os.path.join(train_dir, 'dogs')  # directory with our training dog pictures
validation_cats_dir = os.path.join(validation_dir, 'cats')  # directory with our validation cat pictures
validation_dogs_dir = os.path.join(validation_dir, 'dogs')  # directory with our validation dog pictures
```

In [0]:

```
num_cats_tr = len(os.listdir(train_cats_dir))
num_dogs_tr = len(os.listdir(train_dogs_dir))

num_cats_val = len(os.listdir(validation_cats_dir))
num_dogs_val = len(os.listdir(validation_dogs_dir))

total_train = num_cats_tr + num_dogs_tr
total_val = num_cats_val + num_dogs_val

#Impresion
print('total training cat images:', num_cats_tr)
print('total training dog images:', num_dogs_tr)

print('total validation cat images:', num_cats_val)
print('total validation dog images:', num_dogs_val)
print("--")
print("Total training images:", total_train)
print("Total validation images:", total_val)
```

```
total training cat images: 1000
total training dog images: 1000
total validation cat images: 500
total validation dog images: 500
--
Total training images: 2000
Total validation images: 1000
```

In [0]:

```
#Variables
batch_size = 128
epochs = 15
IMG_HEIGHT = 150
IMG_WIDTH = 150
```

In [0]:

```
#Preparacion de datos
#Leer imágenes del disco.
#Decodifica el contenido de estas imágenes y las convierte al formato adecuado s
egún su contenido RGB.
#Conviérte las imagenes en tensores de coma flotante.
#Cambia la escala de los tensores de valores entre 0 y 255 a valores entre 0 y
 1, ya que las redes neuronales prefieren tratar con valores de entrada pequeño
s.

train_image_generator = ImageDataGenerator(rescale=1./255) # Generator for our t
raining data
validation_image_generator = ImageDataGenerator(rescale=1./255) # Generator for
 our validation data

train_data_gen = train_image_generator.flow_from_directory(batch_size=batch_size
,
                                                           directory=train_dir,
                                                           shuffle=True,
                                                           target_size=(IMG_HEIG
HT, IMG_WIDTH),
                                                           class_mode='binary')
```

```
Found 2000 images belonging to 2 classes.
```

In [0]:

```
print(train_data_gen)
```

```
<keras_preprocessing.image.directory_iterator.DirectoryIterator obje
ct at 0x7f3c99a31b38>
```

In [0]:

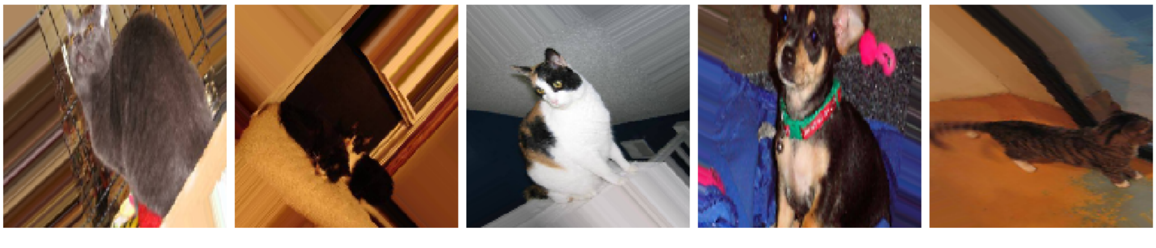
```
val_data_gen = validation_image_generator.flow_from_directory(batch_size=batch_s
ize,
                                                              directory=validati
on_dir,
                                                              target_size=(IMG_H
EIGHT, IMG_WIDTH),
                                                              class_mode='binar
y')
```

```
Found 1000 images belonging to 2 classes.
```

In [0]:

```
#Visualizacion fotos
sample_training_images, _ = next(train_data_gen)
print(sample_training_images[0])

# This function will plot images in the form of a grid with 1 row and 5 columns where images are placed in each column.
def plotImages(images_arr):
    fig, axes = plt.subplots(1, 5, figsize=(20,20))
    axes = axes.flatten()
    for img, ax in zip( images_arr, axes):
        ax.imshow(img)
        ax.axis('off')
    plt.tight_layout()
    plt.show()

plotImages(sample_training_images[:5])
```

```
[[[0.6784314  0.5294118  0.37647063]
  [0.6784314  0.5294118  0.37647063]
  [0.6784314  0.5294118  0.37647063]
  ...
  [0.4666543  0.35579664 0.24955888]
  [0.4918045  0.34893748 0.24041337]
  [0.5089021  0.34811777 0.22019558]]

 [[0.6784314  0.5294118  0.37647063]
  [0.6784314  0.5294118  0.37647063]
  [0.6784314  0.5294118  0.37647063]
  ...
  [0.49491593 0.3480889  0.23928194]
  [0.5097507  0.34896633 0.21595271]
  [0.5166098  0.35582545 0.18165702]]

 [[0.6784314  0.5294118  0.37647063]
  [0.6784314  0.5294118  0.37647063]
  [0.6784314  0.5294118  0.37647063]
  ...
  [0.51059926 0.34981492 0.21170983]
  [0.5174584  0.35667408 0.17741416]
  [0.54655224 0.36798015 0.20092882]]

 ...

 [[0.32182163 0.18064517 0.08349145]
  [0.30546552 0.1610352  0.05582058]
  [0.29632    0.14503054 0.03295679]
  ...
  [0.9478008  0.99485964 0.9490197 ]
  [0.9469179  0.96909744 0.9451921 ]
  [0.985123   0.98441964 0.9765765 ]]

 [[0.30433407 0.15905519 0.052992  ]
  [0.29518858 0.14305054 0.03012821]
  [0.31228557 0.15328841 0.04763754]
  ...
  [0.9521075  0.9965932  0.94747573]
  [0.94354576 0.9803379  0.94744015]
  [0.97388244 0.9720551  0.96421194]]

 [[0.2942539  0.14126728 0.02760233]
  [0.3148313  0.15498556 0.05046612]
  [0.3320771  0.16470589 0.06799932]
  ...
  [0.9588518  0.99771714 0.9441036 ]
  [0.9418452  0.988904   0.9490197 ]
  [0.96264195 0.95969045 0.9518473 ]]]
```



In [0]:

```
sample_training_images, _ = next(train_data_gen)

print(len(sample_training_images))
```

```
128
```

In [0]:

```
#Crear modelo
#El modelo consta de tres bloques de convolución con una capa de agrupación máxi
ma en cada uno de ellos.
#Hay una capa completamente conectada con 512 unidades con funcion de activacion
RELU.
#El modelo genera probabilidades de clase basadas en la clasificación binaria po
r la sigmoidfunción de activación.
model = Sequential([
    Conv2D(16, 3, padding='same', activation='relu', input_shape=(IMG_HEIGHT, IM
G_WIDTH ,3)),
    MaxPooling2D(),
    Conv2D(32, 3, padding='same', activation='relu'),
    MaxPooling2D(),
    Conv2D(64, 3, padding='same', activation='relu'),
    MaxPooling2D(),
    Flatten(),
    Dense(512, activation='relu'),
    Dense(1, activation='sigmoid')
])

model.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])

model.summary()
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tenso
rflow_core/python/ops/resource_variable_ops.py:1630: calling BaseRes
ourceVariable.__init__ (from tensorflow.python.ops.resource_variable
_ops) with constraint is deprecated and will be removed in a future
version.
Instructions for updating:
If using Keras pass *_constraint arguments to layers.
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tenso
rflow_core/python/ops/nn_impl.py:183: where (from tensorflow.python.
ops.array_ops) is deprecated and will be removed in a future versio
n.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 150, 150, 16)      448
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 75, 75, 16)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 75, 75, 32)        4640
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 37, 37, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 37, 37, 64)        18496
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 18, 18, 64)        0
_________________________________________________________________
flatten (Flatten)            (None, 20736)             0
_________________________________________________________________
dense (Dense)                (None, 512)               10617344
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 513
=================================================================
Total params: 10,641,441
Trainable params: 10,641,441
Non-trainable params: 0
_________________________________________________________________
```

In [0]:

```
#Entrenamiento
history = model.fit_generator(
    train_data_gen,
    steps_per_epoch=total_train // batch_size,
    epochs=epochs,
    validation_data=val_data_gen,
    validation_steps=total_val // batch_size
)
```

```
Epoch 1/15
14/15 [===========================>..] - ETA: 3s - loss: 1.0030 - ac
c: 0.4862Epoch 1/15
15/15 [==============================] - 55s 4s/step - loss: 0.9826
- acc: 0.4877 - val_loss: 0.6923 - val_acc: 0.5279
Epoch 2/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6906 - ac
c: 0.5151Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.6905
- acc: 0.5155 - val_loss: 0.6861 - val_acc: 0.6373
Epoch 3/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6808 - ac
c: 0.5516Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.6801
- acc: 0.5588 - val_loss: 0.6659 - val_acc: 0.6462
Epoch 4/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6490 - ac
c: 0.6370Epoch 1/15
15/15 [==============================] - 55s 4s/step - loss: 0.6464
- acc: 0.6410 - val_loss: 0.6168 - val_acc: 0.6708
Epoch 5/15
14/15 [===========================>..] - ETA: 3s - loss: 0.5828 - ac
c: 0.7007Epoch 1/15
15/15 [==============================] - 55s 4s/step - loss: 0.5808
- acc: 0.6982 - val_loss: 0.5863 - val_acc: 0.6875
Epoch 6/15
14/15 [===========================>..] - ETA: 3s - loss: 0.5581 - ac
c: 0.7121Epoch 1/15
15/15 [==============================] - 56s 4s/step - loss: 0.5564
- acc: 0.7120 - val_loss: 0.5795 - val_acc: 0.6942
Epoch 7/15
14/15 [===========================>..] - ETA: 3s - loss: 0.5277 - ac
c: 0.7362Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.5242
- acc: 0.7368 - val_loss: 0.5548 - val_acc: 0.7232
Epoch 8/15
14/15 [===========================>..] - ETA: 3s - loss: 0.4955 - ac
c: 0.7575Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.4905
- acc: 0.7618 - val_loss: 0.5807 - val_acc: 0.7054
Epoch 9/15
14/15 [===========================>..] - ETA: 3s - loss: 0.4496 - ac
c: 0.7829Epoch 1/15
15/15 [==============================] - 56s 4s/step - loss: 0.4482
- acc: 0.7828 - val_loss: 0.5437 - val_acc: 0.7310
Epoch 10/15
14/15 [===========================>..] - ETA: 3s - loss: 0.4220 - ac
c: 0.8102Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.4263
- acc: 0.8072 - val_loss: 0.6127 - val_acc: 0.6964
Epoch 11/15
14/15 [===========================>..] - ETA: 3s - loss: 0.4454 - ac
c: 0.7936Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.4433
- acc: 0.7933 - val_loss: 0.5683 - val_acc: 0.7109
Epoch 12/15
14/15 [===========================>..] - ETA: 3s - loss: 0.3797 - ac
c: 0.8337Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.3803
- acc: 0.8328 - val_loss: 0.6159 - val_acc: 0.6964
Epoch 13/15
```

```
14/15 [===========================>..] - ETA: 3s - loss: 0.3717 - ac
c: 0.8377Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.3674
- acc: 0.8408 - val_loss: 0.6079 - val_acc: 0.6920
Epoch 14/15
14/15 [===========================>..] - ETA: 2s - loss: 0.3282 - ac
c: 0.8710Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.3198
- acc: 0.8750 - val_loss: 0.6159 - val_acc: 0.7165
Epoch 15/15
14/15 [===========================>..] - ETA: 2s - loss: 0.2840 - ac
c: 0.8859Epoch 1/15
15/15 [==============================] - 54s 4s/step - loss: 0.2777
- acc: 0.8905 - val_loss: 0.6366 - val_acc: 0.7165
```

In [0]:

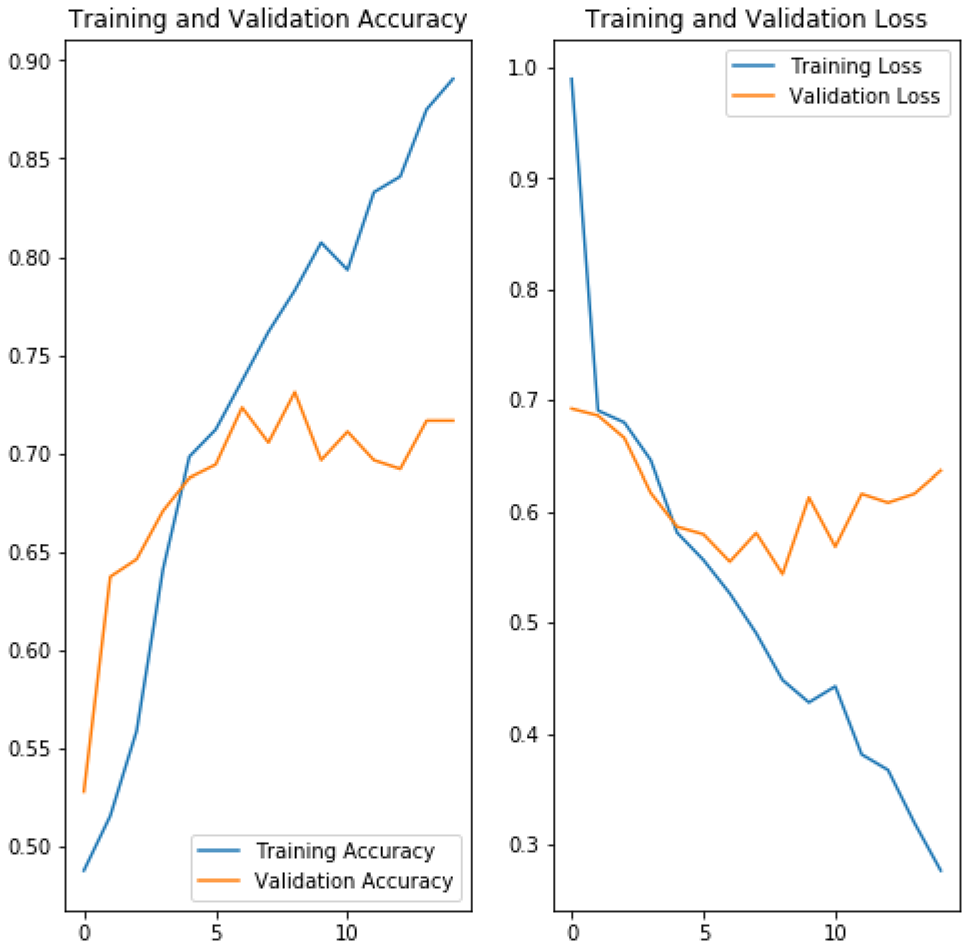
```
model.metrics_names
```

Out[0]:

```
['loss', 'acc']
```

In [0]:

```
#Visualizacion de Resultados
acc = history.history['acc']
val_acc = history.history['val_acc']

loss = history.history['loss']
val_loss = history.history['val_loss']

epochs_range = range(epochs)

plt.figure(figsize=(8, 8))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy')
plt.plot(epochs_range, val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss')
plt.plot(epochs_range, val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()
```

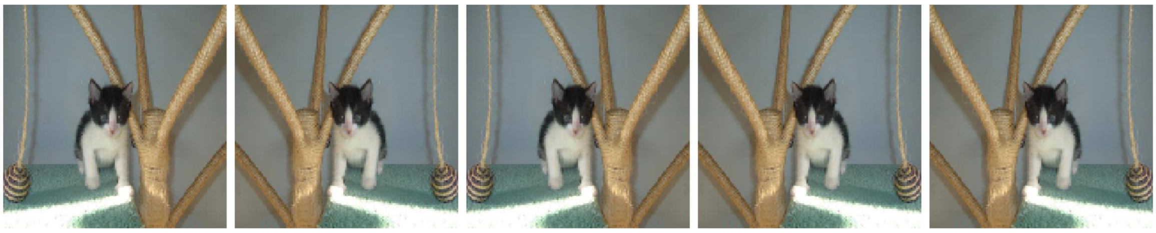


Generacion de más imágenes

In [0]:

```
#Voldeo Horizontal
image_gen = ImageDataGenerator(rescale=1./255, horizontal_flip=True)
train_data_gen = image_gen.flow_from_directory(batch_size=batch_size,
                                               directory=train_dir,
                                               shuffle=True,
                                               target_size=(IMG_HEIGHT, IMG_WIDT
H))
augmented_images = [train_data_gen[0][0][0] for i in range(5)]
# Re-use the same custom plotting function defined and used
# above to visualize the training images
plotImages(augmented_images)
```

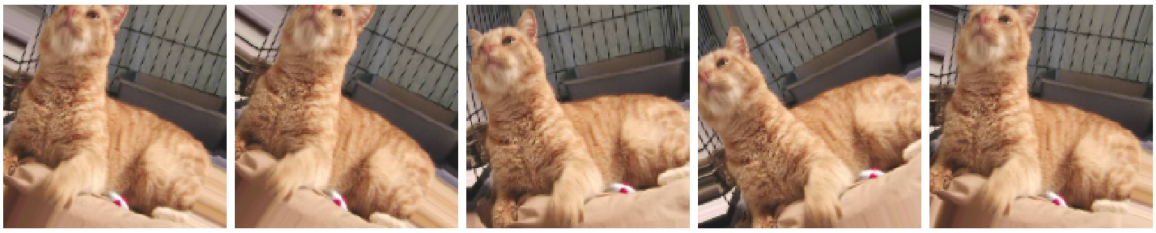
```
Found 2000 images belonging to 2 classes.
```



In [0]:

```
#Rotacion
image_gen = ImageDataGenerator(rescale=1./255, rotation_range=45)
train_data_gen = image_gen.flow_from_directory(batch_size=batch_size,
                                               directory=train_dir,
                                               shuffle=True,
                                               target_size=(IMG_HEIGHT, IMG_WIDT
H))

augmented_images = [train_data_gen[0][0][0] for i in range(5)]
plotImages(augmented_images)
```

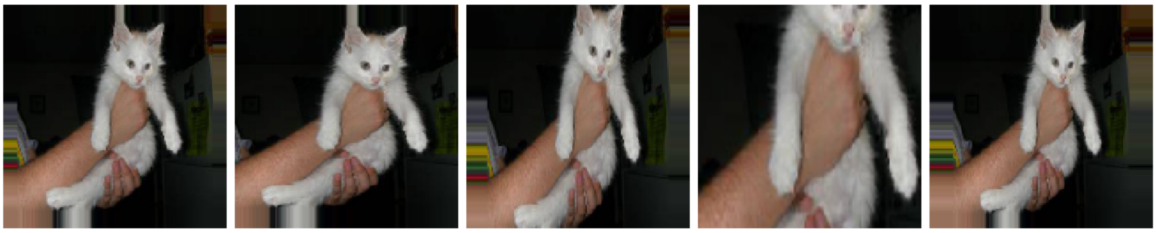
```
Found 2000 images belonging to 2 classes.
```



In [0]:

```
#Aumento de zoom
# zoom_range from 0 - 1 where 1 = 100%.
image_gen = ImageDataGenerator(rescale=1./255, zoom_range=0.5) #

train_data_gen = image_gen.flow_from_directory(batch_size=batch_size,
                                               directory=train_dir,
                                               shuffle=True,
                                               target_size=(IMG_HEIGHT, IMG_WIDT
H))

augmented_images = [train_data_gen[0][0][0] for i in range(5)]
plotImages(augmented_images)
```

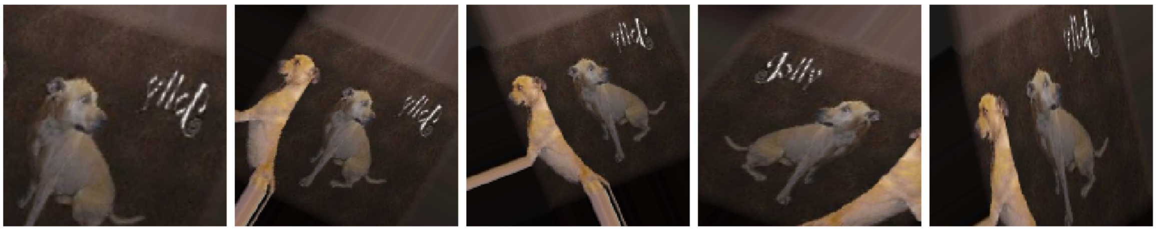
```
Found 2000 images belonging to 2 classes.
```



In [0]:

```
#Union de todas las imagenes
image_gen_train = ImageDataGenerator(
                    rescale=1./255,
                    rotation_range=45,
                    width_shift_range=.15,
                    height_shift_range=.15,
                    horizontal_flip=True,
                    zoom_range=0.5
                    )
train_data_gen = image_gen_train.flow_from_directory(batch_size=batch_size,
                                                     directory=train_dir,
                                                     shuffle=True,
                                                     target_size=(IMG_HEIGHT, IM
G_WIDTH),
                                                     class_mode='binary')

augmented_images = [train_data_gen[0][0][0] for i in range(5)]
plotImages(augmented_images)
```

```
Found 2000 images belonging to 2 classes.
```



In [0]:

```
#datos de validacion
image_gen_val = ImageDataGenerator(rescale=1./255)

val_data_gen = image_gen_val.flow_from_directory(batch_size=batch_size,
                                                 directory=validation_dir,
                                                 target_size=(IMG_HEIGHT, IMG_WI
DTH),
                                                 class_mode='binary')

print(val_data_gen)
```

```
Found 1000 images belonging to 2 classes.
<keras_preprocessing.image.directory_iterator.DirectoryIterator obje
ct at 0x7f3c994a8668>
```

In [0]:

```
#Cracion del modelo
model_new = Sequential([
    Conv2D(16, 3, padding='same', activation='relu',
           input_shape=(IMG_HEIGHT, IMG_WIDTH ,3)),
    MaxPooling2D(),
    Dropout(0.2),
    Conv2D(32, 3, padding='same', activation='relu'),
    MaxPooling2D(),
    Conv2D(64, 3, padding='same', activation='relu'),
    MaxPooling2D(),
    Dropout(0.2),
    Flatten(),
    Dense(512, activation='relu'),
    Dense(1, activation='sigmoid')
])

#Compilacion del modelo
model_new.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])

model_new.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_3 (Conv2D)            (None, 150, 150, 16)      448       
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 75, 75, 16)        0         
_________________________________________________________________
dropout (Dropout)            (None, 75, 75, 16)        0         
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 75, 75, 32)        4640      
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 37, 37, 32)        0         
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 37, 37, 64)        18496     
_________________________________________________________________
max_pooling2d_5 (MaxPooling2 (None, 18, 18, 64)        0         
_________________________________________________________________
dropout_1 (Dropout)          (None, 18, 18, 64)        0         
_________________________________________________________________
flatten_1 (Flatten)          (None, 20736)             0         
_________________________________________________________________
dense_2 (Dense)              (None, 512)               10617344  
_________________________________________________________________
dense_3 (Dense)              (None, 1)                 513       
=================================================================
Total params: 10,641,441
Trainable params: 10,641,441
Non-trainable params: 0
_________________________________________________________________
```

In [0]:

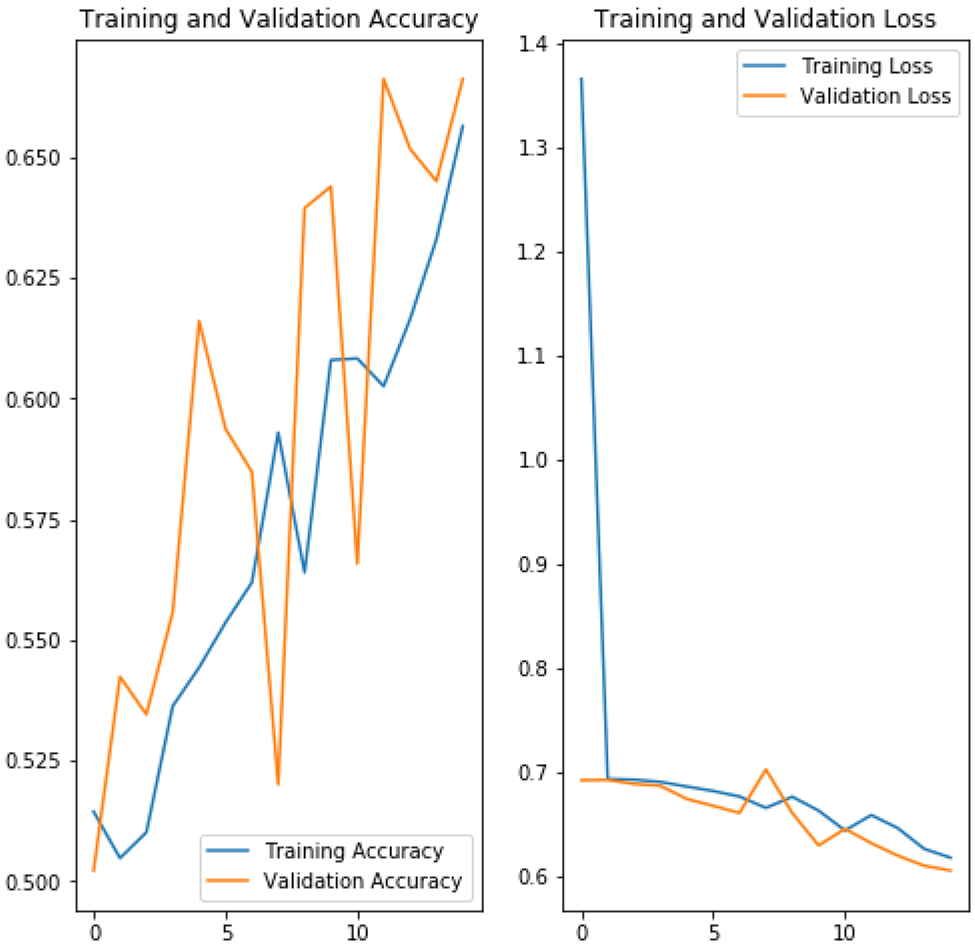
```
#Entrenamiento
history = model_new.fit_generator(
    train_data_gen,
    steps_per_epoch=total_train // batch_size,
    epochs=epochs,
    validation_data=val_data_gen,
    validation_steps=total_val // batch_size
)
```

```
Epoch 1/15
14/15 [===========================>..] - ETA: 3s - loss: 1.3952 - ac
c: 0.5138Epoch 1/15
15/15 [==============================] - 65s 4s/step - loss: 1.3494
- acc: 0.5144 - val_loss: 0.6917 - val_acc: 0.5022
Epoch 2/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6929 - ac
c: 0.5069Epoch 1/15
15/15 [==============================] - 64s 4s/step - loss: 0.6929
- acc: 0.5048 - val_loss: 0.6921 - val_acc: 0.5424
Epoch 3/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6924 - ac
c: 0.5103Epoch 1/15
15/15 [==============================] - 64s 4s/step - loss: 0.6923
- acc: 0.5101 - val_loss: 0.6881 - val_acc: 0.5346
Epoch 4/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6907 - ac
c: 0.5264Epoch 1/15
15/15 [==============================] - 66s 4s/step - loss: 0.6899
- acc: 0.5363 - val_loss: 0.6864 - val_acc: 0.5558
Epoch 5/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6862 - ac
c: 0.5430Epoch 1/15
15/15 [==============================] - 64s 4s/step - loss: 0.6859
- acc: 0.5443 - val_loss: 0.6737 - val_acc: 0.6161
Epoch 6/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6807 - ac
c: 0.5525Epoch 1/15
15/15 [==============================] - 65s 4s/step - loss: 0.6814
- acc: 0.5536 - val_loss: 0.6671 - val_acc: 0.5938
Epoch 7/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6769 - ac
c: 0.5642Epoch 1/15
15/15 [==============================] - 63s 4s/step - loss: 0.6768
- acc: 0.5620 - val_loss: 0.6603 - val_acc: 0.5848
Epoch 8/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6648 - ac
c: 0.5952Epoch 1/15
15/15 [==============================] - 64s 4s/step - loss: 0.6649
- acc: 0.5929 - val_loss: 0.7021 - val_acc: 0.5201
Epoch 9/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6735 - ac
c: 0.5737Epoch 1/15
15/15 [==============================] - 66s 4s/step - loss: 0.6759
- acc: 0.5641 - val_loss: 0.6602 - val_acc: 0.6395
Epoch 10/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6633 - ac
c: 0.6026Epoch 1/15
15/15 [==============================] - 63s 4s/step - loss: 0.6621
- acc: 0.6080 - val_loss: 0.6291 - val_acc: 0.6440
Epoch 11/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6435 - ac
c: 0.6110Epoch 1/15
15/15 [==============================] - 66s 4s/step - loss: 0.6435
- acc: 0.6083 - val_loss: 0.6450 - val_acc: 0.5658
Epoch 12/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6584 - ac
c: 0.5992Epoch 1/15
15/15 [==============================] - 63s 4s/step - loss: 0.6570
- acc: 0.6026 - val_loss: 0.6312 - val_acc: 0.6663
Epoch 13/15
```

```
14/15 [===========================>..] - ETA: 3s - loss: 0.6480 - ac
c: 0.6187Epoch 1/15
15/15 [==============================] - 64s 4s/step - loss: 0.6468
- acc: 0.6165 - val_loss: 0.6193 - val_acc: 0.6518
Epoch 14/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6248 - ac
c: 0.6365Epoch 1/15
15/15 [==============================] - 62s 4s/step - loss: 0.6254
- acc: 0.6330 - val_loss: 0.6095 - val_acc: 0.6451
Epoch 15/15
14/15 [===========================>..] - ETA: 3s - loss: 0.6193 - ac
c: 0.6548Epoch 1/15
15/15 [==============================] - 64s 4s/step - loss: 0.6177
- acc: 0.6565 - val_loss: 0.6050 - val_acc: 0.6663
```

In [0]:

```
acc = history.history['acc']
val_acc = history.history['val_acc']

loss = history.history['loss']
val_loss = history.history['val_loss']

epochs_range = range(epochs)

plt.figure(figsize=(8, 8))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy')
plt.plot(epochs_range, val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss')
plt.plot(epochs_range, val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()
```


Training and Validation Accuracy
0.650
0.625
0.600
0.575
0.550
0.525
0.500
0
5
10
Training Accuracy
Validation Accuracy
Training and Validation Loss
1.4
1.3
1.2
1.1
1.0
0.9
0.8
0.7
0.6
0
5
10
Training Loss
Validation Loss