

```
# unzipping dataset
from google.colab import drive
drive.mount('/content/gdrive')
!unzip gdrive/MyDrive/FaceRecognition/archive2.zip

inflatng: __MACOSX/archive2/UTKFace/.70_0_0_20170111203945060.jpg.chip.jpg
inflatng: archive2/UTKFace/7_1_0_20170109205356415.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.7_1_0_20170109205356415.jpg.chip.jpg
inflatng: archive2/UTKFace/6_0_4_20161221202547897.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.6_0_4_20161221202547897.jpg.chip.jpg
inflatng: archive2/UTKFace/1_1_0_20170109194432489.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.1_1_0_20170109194432489.jpg.chip.jpg
inflatng: archive2/UTKFace/8_1_0_20170109200918322.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.8_1_0_20170109200918322.jpg.chip.jpg
inflatng: archive2/UTKFace/30_0_0_20170104201841050.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.30_0_0_20170104201841050.jpg.chip.jpg
inflatng: archive2/UTKFace/29_0_3_20170104220737116.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.29_0_3_20170104220737116.jpg.chip.jpg
inflatng: archive2/UTKFace/24_1_3_20170104231911026.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.24_1_3_20170104231911026.jpg.chip.jpg
inflatng: archive2/UTKFace/26_1_2_20170116184131117.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.26_1_2_20170116184131117.jpg.chip.jpg
inflatng: archive2/UTKFace/27_1_0_20170116173136538.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.27_1_0_20170116173136538.jpg.chip.jpg
inflatng: archive2/UTKFace/54_1_0_20170117191523029.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.54_1_0_20170117191523029.jpg.chip.jpg
inflatng: archive2/UTKFace/39_1_2_20170103182754641.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.39_1_2_20170103182754641.jpg.chip.jpg
inflatng: archive2/UTKFace/35_1_0_20170117154643574.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.35_1_0_20170117154643574.jpg.chip.jpg
inflatng: archive2/UTKFace/85_1_0_20170110181946121.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.85_1_0_20170110181946121.jpg.chip.jpg
inflatng: archive2/UTKFace/53_0_4_20170109012743281.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.53_0_4_20170109012743281.jpg.chip.jpg
inflatng: archive2/UTKFace/26_0_3_20170117154627472.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.26_0_3_20170117154627472.jpg.chip.jpg
inflatng: archive2/UTKFace/25_1_3_20170104222459455.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.25_1_3_20170104222459455.jpg.chip.jpg
inflatng: archive2/UTKFace/57_0_0_20170112203923347.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.57_0_0_20170112203923347.jpg.chip.jpg
inflatng: archive2/UTKFace/26_1_2_20170116184831737.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.26_1_2_20170116184831737.jpg.chip.jpg
inflatng: archive2/UTKFace/14_1_0_20170109203625481.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.14_1_0_20170109203625481.jpg.chip.jpg
inflatng: archive2/UTKFace/33_0_1_20170114034227427.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.33_0_1_20170114034227427.jpg.chip.jpg
inflatng: archive2/UTKFace/54_0_0_20170116193518166.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.54_0_0_20170116193518166.jpg.chip.jpg
inflatng: archive2/UTKFace/30_1_0_20170117181416245.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.30_1_0_20170117181416245.jpg.chip.jpg
inflatng: archive2/UTKFace/89_0_2_20170112220351776.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.89_0_2_20170112220351776.jpg.chip.jpg
inflatng: archive2/UTKFace/51_0_0_20170117160748775.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.51_0_0_20170117160748775.jpg.chip.jpg
inflatng: archive2/UTKFace/24_1_0_20170116222405565.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.24_1_0_20170116222405565.jpg.chip.jpg
inflatng: archive2/UTKFace/8_1_0_20170109202318713.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.8_1_0_20170109202318713.jpg.chip.jpg
inflatng: archive2/UTKFace/9_0_0_20170110225030430.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.9_0_0_20170110225030430.jpg.chip.jpg
inflatng: archive2/UTKFace/36_0_1_20170113182511607.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.36_0_1_20170113182511607.jpg.chip.jpg
inflatng: archive2/UTKFace/69_1_0_20170110141835631.jpg.chip.jpg
inflatng: __MACOSX/archive2/UTKFace/.69_1_0_20170110141835631.jpg.chip.jpg
```

```
# import modules
import random
import keras
from tensorflow.keras import layers
from keras.models import Sequential
import tensorflow as tf
import tensorflow_datasets as tfds
from keras.layers.core import Dense, Activation, Dropout, Flatten
from keras.layers.convolutional import Convolution2D, MaxPooling2D
from keras.optimizers import SGD, RMSprop, adam
from keras.utils import np_utils
from sklearn.tree import DecisionTreeClassifier # Import Decision Tree Classifier
import matplotlib.pyplot as plt
import numpy as np
import os
import cv2
```

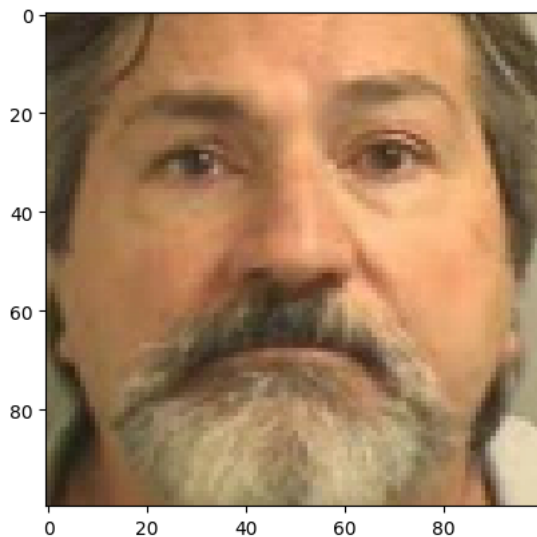
```

from PIL import Image
import matplotlib.pyplot as plt
from keras.utils import np_utils
from sklearn.model_selection import train_test_split
from keras.applications.vgg16 import VGG16
from keras.applications.vgg16 import VGG16
from keras.models import Model
from tensorflow.keras import layers
from tensorflow.keras.applications.vgg16 import preprocess_input
import tensorflow_datasets as tfds

path = "/content/archive2/UTKFace/"
amounts = [0,0,0,0,0]
training = []
count = 0
for img in os.listdir(path):
    if img != ".DS_Store":
        classification = img.split("_")
        category = classification[2]
        if category != "20170116174525125.jpg.chip.jpg" and category != "20170109142408075.jpg.chip.jpg":
            category = int(category)
            if amounts[category] < 2000:
                amounts[category] +=1
                img_array = cv2.imread(os.path.join(path, img))
                img_array = cv2.cvtColor(img_array,cv2.COLOR_RGB2BGR)
                imga = cv2.resize(img_array,dsize = (100,100), interpolation = cv2.INTER_CUBIC)
                training.append([imga, category])
# randomizes data
random.shuffle(training)
plt.imshow(training[0][0])
print(amounts)

```

```
[2000, 2000, 2000, 2000, 1692]
```



```

X = []
y = []

for features, label in training:
    X.append(features)
    y.append(label)

Xarray = np.array(X).reshape(-1, 100, 100, 3)
Xarray.shape
Xarraydiv = Xarray/ 255

Xarraytype = Xarraydiv.astype('float32')
nexty = np_utils.to_categorical(y,5)

X_train, X_test, y_train, y_test = train_test_split(Xarraytype,nexty,test_size = .2, random_state = 5)

input_shape = (100,100,3)
model6 = keras.Sequential(

```

```
[
    keras.Input(shape=input_shape),
    layers.RandomFlip("horizontal"),
    layers.Conv2D(32, kernel_size=(3, 3), activation="relu"),
    layers.MaxPooling2D(pool_size=(2, 2)),
    layers.Dropout(0.2),
    layers.Conv2D(64, kernel_size=(3, 3), activation="relu"),
    layers.MaxPooling2D(pool_size=(2, 2)),
    layers.RandomRotation(0.1),
    layers.Conv2D(128, kernel_size=(3, 3), activation="relu"),
    layers.MaxPooling2D(pool_size=(2, 2)),
    layers.Dropout(0.2),
    layers.Conv2D(256, kernel_size=(3, 3), activation="relu"),
    layers.MaxPooling2D(pool_size=(2, 2)),
    layers.Conv2D(512, kernel_size=(3, 3), activation="relu"),
    layers.MaxPooling2D(pool_size=(2, 2)),
    layers.Flatten(),
    layers.Dropout(0.5),
    layers.Dense(100, activation="sigmoid"),
    layers.Dense(5, activation="sigmoid"),
]
)
model6.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
=====		
random_flip (RandomFlip)	(None, 100, 100, 3)	0
conv2d (Conv2D)	(None, 98, 98, 32)	896
max_pooling2d (MaxPooling2D)	(None, 49, 49, 32)	0
dropout (Dropout)	(None, 49, 49, 32)	0
conv2d_1 (Conv2D)	(None, 47, 47, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 23, 23, 64)	0
random_rotation (RandomRotation)	(None, 23, 23, 64)	0
conv2d_2 (Conv2D)	(None, 21, 21, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 10, 10, 128)	0
dropout_1 (Dropout)	(None, 10, 10, 128)	0
conv2d_3 (Conv2D)	(None, 8, 8, 256)	295168
max_pooling2d_3 (MaxPooling2D)	(None, 4, 4, 256)	0
conv2d_4 (Conv2D)	(None, 2, 2, 512)	1180160
max_pooling2d_4 (MaxPooling2D)	(None, 1, 1, 512)	0
flatten (Flatten)	(None, 512)	0
dropout_2 (Dropout)	(None, 512)	0
dense (Dense)	(None, 100)	51300
dense_1 (Dense)	(None, 5)	505
=====		
Total params: 1,620,381		
Trainable params: 1,620,381		
Non-trainable params: 0		

```

batch_size = 32
epochs = 20
model6.compile(loss="categorical_crossentropy", optimizer="adam", metrics=["accuracy"])

model6.fit(X_train, y_train, batch_size=batch_size, epochs=epochs, validation_data =(X_test,y_test))

Epoch 1/20
243/243 [=====] - 9s 24ms/step - loss: 0.8421 - accuracy: 0.6770 - val_loss: 0.8718 - val_accuracy:
Epoch 2/20
243/243 [=====] - 5s 22ms/step - loss: 0.8397 - accuracy: 0.6819 - val_loss: 0.8895 - val_accuracy:
Epoch 3/20
243/243 [=====] - 5s 21ms/step - loss: 0.8254 - accuracy: 0.6871 - val_loss: 0.8250 - val_accuracy:
Epoch 4/20
243/243 [=====] - 5s 20ms/step - loss: 0.8067 - accuracy: 0.6919 - val_loss: 0.8198 - val_accuracy:
Epoch 5/20
243/243 [=====] - 5s 21ms/step - loss: 0.8096 - accuracy: 0.6957 - val_loss: 0.8755 - val_accuracy:
Epoch 6/20
243/243 [=====] - 5s 21ms/step - loss: 0.7975 - accuracy: 0.6938 - val_loss: 0.8301 - val_accuracy:
Epoch 7/20
243/243 [=====] - 5s 22ms/step - loss: 0.8045 - accuracy: 0.6944 - val_loss: 0.8280 - val_accuracy:
Epoch 8/20
243/243 [=====] - 5s 20ms/step - loss: 0.7898 - accuracy: 0.6986 - val_loss: 0.8608 - val_accuracy:
Epoch 9/20
243/243 [=====] - 5s 20ms/step - loss: 0.7818 - accuracy: 0.7053 - val_loss: 0.8096 - val_accuracy:
Epoch 10/20
243/243 [=====] - 5s 21ms/step - loss: 0.7716 - accuracy: 0.7147 - val_loss: 0.8170 - val_accuracy:
Epoch 11/20
243/243 [=====] - 5s 20ms/step - loss: 0.7759 - accuracy: 0.7066 - val_loss: 0.7924 - val_accuracy:
Epoch 12/20
243/243 [=====] - 5s 21ms/step - loss: 0.7571 - accuracy: 0.7103 - val_loss: 0.8215 - val_accuracy:
Epoch 13/20
243/243 [=====] - 5s 21ms/step - loss: 0.7606 - accuracy: 0.7153 - val_loss: 0.8434 - val_accuracy:
Epoch 14/20
243/243 [=====] - 5s 21ms/step - loss: 0.7527 - accuracy: 0.7140 - val_loss: 0.8231 - val_accuracy:
Epoch 15/20
243/243 [=====] - 5s 20ms/step - loss: 0.7456 - accuracy: 0.7177 - val_loss: 0.8443 - val_accuracy:
Epoch 16/20
243/243 [=====] - 5s 22ms/step - loss: 0.7440 - accuracy: 0.7178 - val_loss: 0.8115 - val_accuracy:
Epoch 17/20
243/243 [=====] - 5s 21ms/step - loss: 0.7341 - accuracy: 0.7232 - val_loss: 0.8724 - val_accuracy:
Epoch 18/20
243/243 [=====] - 5s 22ms/step - loss: 0.7197 - accuracy: 0.7269 - val_loss: 0.8117 - val_accuracy:
Epoch 19/20
243/243 [=====] - 5s 20ms/step - loss: 0.7328 - accuracy: 0.7294 - val_loss: 0.8283 - val_accuracy:
Epoch 20/20
243/243 [=====] - 5s 20ms/step - loss: 0.7077 - accuracy: 0.7318 - val_loss: 0.8005 - val_accuracy:
<keras.callbacks.History at 0x7f9f8410e9e0>

path = "/content/gdrive/MyDrive/RaceRecognition"
model6.save(path)

WARNING:absl:Found untraced functions such as _jit_compiled_convolution_op, _jit_compiled_convolution_op, _jit_compiled_conv

```