```
import tensorflow as tf
import tensorflow.keras as K
from tensorflow.keras.applications import ResNet50
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.resnet50 import preprocess_input, decode_predictions
import numpy as np
from sklearn.decomposition import PCA
from tensorflow.keras import datasets, layers, models
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
import os
import glob
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt


#função para acessar as pastas


def obter_caminhos_de_imagens(diretorio_raiz):
    caminhos_de_imagens = []
    for pasta_raiz, _, arquivos in os.walk(diretorio_raiz):
        for arquivo in arquivos:
            if arquivo.endswith(('.jpg', '.jpeg', '.png', '.bmp')):
                caminhos_de_imagens.append(os.path.join(pasta_raiz, arquivo))
    return caminhos_de_imagens

diretorio_raiz = 'C:\\Users\\laiss\\OneDrive\Arquivos\\analista de dados\\Redes Neurais Profundas\\animals10_small\\animals10_small'
caminhos_de_imagens = obter_caminhos_de_imagens(diretorio_raiz)


caminhos_de_imagens
```

```
['C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e030b20928e90021d85a5854ee454296eb70e3c818b413449df6c87ca3ed_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e132b60e2af51c22d2524518b7444f92e37fe5d404b0144390f8c770a3e5b7_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e831b00c2ef3093ed1584d05fb1d4e9fe777ead218ac104497f5c97faee9bdba_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e831b30a28f21c22d2524518b7444f92e37fe5d404b0144390f8c770a3e5b7_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e832b4082df5003ed1584d05fb1d4e9fe777ead218ac104497f5c97faee9bdba_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e833b40e28f01c22d2524518b7444f92e37fe5d404b0144390f8c770a1edb6_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e833b80721f2043ed1584d05fb1d4e9fe777ead218ac104497f5c97faee9bdba_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e835b4082ff2093ed1584d05fb1d4e9fe777ead218ac104497f5c97faee9bdba_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e836b80e20f4083ed1584d05fb1d4e9fe777ead218ac104497f5c97faee9bdba_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e837b00720fc053ed1584d05fb1d4e9fe777ead218ac104497f5c97faee9bdba_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e83db70d28f31c22d2524518b7444f92e37fe5d404b0144390f8c770a2e9b5_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e83db70d28f51c22d2524518b7444f92e37fe5d404b0144390f8c770a2e9b5_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\e83db70e2cf0073ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.png',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea34b20a2ef5013ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea34b4072cf0073ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea34b40a21fd013ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea34b40d21f0023ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea34b60d20f3033ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea34b7082cf41c22d2524518b7444f92e37fe5d404b0144390f8c770a2e9b5_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea34b80820fd013ed1584d05fb1d4e9fe777ead218ac104497f5c97faee8b1b8_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea34b80b2cfc093ed1584d05fb1d4e9fe777ead218ac104497f5c97faee8b1b8_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea35b50c2ffd013ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea35b50e2bf6033ed1584d05fb1d4e9fe777ead218ac104497f5c97faee8b1b8_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea35b8062df1063ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea35b80a2ef1043ed1584d05fb1d4e9fe777ead218ac104497f5c97faeebb5bb_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea36b00620f2033ed1584d05fb1d4e9fe777ead218ac104497f5c97faee9bdba_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea36b00828f6033ed1584d05fb1d4e9fe777ead218ac104497f5c97faee9bdba_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea36b0082af5053ed1584d05fb1d4e9fe777ead218ac104497f5c97faee8b1b8_640.jpg',
 'C:\\Users\\laiss\\OneDrive\\Arquivos\\analista de dados\\Redes Neurais
Profundas\\animals10_small\\animals10_small\\butterfly\\ea36b00a2af3023ed1584d05fb1d4e9fe777ead218ac104497f5c97faee8b1b8_640.jpg',
```

```
# Carregar o modelo ResNet50 pré-treinado para extrair as características das imagens na penultima camada.
model = ResNet50(weights='imagenet', include_top=False)


def load_and_preprocess_images(caminhos_de_imagens):
    images = []
    for path in caminhos_de_imagens:
        img = image.load_img(path, target_size=(224, 224))
        img_array = image.img_to_array(img)
        img_array = np.expand_dims(img_array, axis=0)
        img_array = preprocess_input(img_array)
        images.append(img_array)
    return np.vstack(images)

# Carregar e pré-processar as imagens
images = load_and_preprocess_images(caminhos_de_imagens)

# Extrair características das imagens na penúltima camada
features = model.predict(images)
```

```
157/157 ━━━━━━━━━━━━━━━━━━━━ 333s 2s/step
```

```
print(images) # verificando as matrizes
```

```
[[[[  14.060997     30.221        27.32      ]
   [  15.060997     31.221        28.32      ]
   [  19.060997     32.221        31.32      ]
   ...
   [ -33.939003     12.221001    -14.68      ]
   [ -33.939003     12.221001    -14.68      ]
   [ -34.939003     11.221001    -15.68      ]]

  [[  14.060997     30.221        27.32      ]
   [  16.060997     32.221        29.32      ]
   [  19.060997     32.221        31.32      ]
   ...
   [ -32.939003     13.221001    -13.68      ]
   [ -31.939003     11.221001    -14.68      ]
   [ -34.939003      8.221001    -17.68      ]]

  [[  14.060997     30.221        27.32      ]
   [  15.060997     31.221        28.32      ]
   [  18.060997     31.221        30.32      ]
   ...
   [ -31.939003     11.221001    -14.68      ]
   [ -32.939003     10.221001    -15.68      ]
   [ -32.939003      8.221001    -15.68      ]]

  ...

  [[ -19.939003     47.221        41.32      ]
   [ -20.939003     46.221        40.32      ]
   [ -20.939003     46.221        40.32      ]
   ...
   [  -5.939003     25.221        35.32      ]
   [  -8.939003     25.221        32.32      ]
   [  -9.939003     24.221        31.32      ]]

  [[ -22.939003     44.221        38.32      ]
   [ -22.939003     44.221        38.32      ]
   [ -22.939003     44.221        38.32      ]
   ...
   [  -6.939003     24.221        34.32      ]
   [  -8.939003     25.221        32.32      ]
   [  -8.939003     25.221        32.32      ]]

  [[ -23.939003     43.221        37.32      ]
   [ -23.939003     43.221        37.32      ]
   [ -23.939003     43.221        37.32      ]
   ...
   [  -7.939003     23.221        33.32      ]
   [  -9.939003     24.221        31.32      ]
   [  -9.939003     24.221        31.32      ]]]


 [[[-103.939      -116.779      -123.68      ]
   [-103.939      -116.779      -123.68      ]
   [-103.939      -116.779      -123.68      ]
   ...
   [-103.939      -116.779      -123.68      ]
   [-103.939      -116.779      -123.68      ]
   [-103.939      -116.779      -123.68      ]]
```

```
# Remodelar as características para duas dimensões
num_samples = features.shape[0]
features_2d = features.reshape(num_samples, -1)

# Realizar a clusterização dos vetores de características usando K-médias
kmeans = KMeans(n_clusters=10)
clusters = kmeans.fit_predict(features_2d)
```

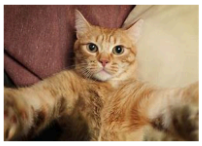
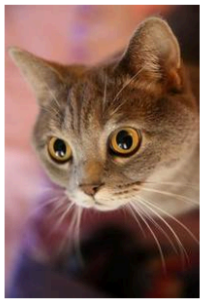
```
# Agora que temos os clusters, vamos organizar as imagens de acordo com os clusters
imagens_por_cluster = [[] for _ in range(10)]

for i, cluster in enumerate(clusters):
    imagens_por_cluster[cluster].append(caminhos_de_imagens[i])

for cluster_id, imagens_cluster in enumerate(imagens_por_cluster):
    print(f'Cluster {cluster_id}:')
    plt.figure(figsize=(15, 10))
    for i, imagem_path in enumerate(imagens_cluster[:5]):  # Exibindo apenas 5 imagens de cada cluster
        imagem = image.load_img(imagem_path)
        plt.subplot(1, 5, i+1)
        plt.imshow(imagem)
        plt.axis('off')
    plt.show()
```

Cluster 0:





Cluster 1:


Cluster 2:






Cluster 3:





Cluster 4:





Cluster 5:



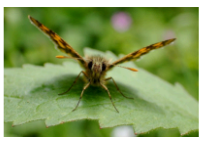


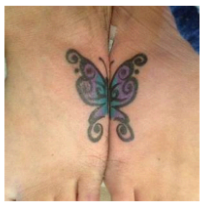
Cluster 6:





Cluster 7:






Cluster 8:

Cluster 8:

Cluster 9:

Os clusters apresentados não foram satisfatórios, porém, é notório que o modelo tentou agrupar da melhor forma que encontrou. Tivemos clusters bem definidos como o 0, 1, 3, 5 e 8.