

## Medical Plants Detection

## Import Packages

```
In [ ]: import numpy as np
import os
import sklearn.metrics as metrics
import seaborn as sns
import sklearn.utils as sklearn_utils
import matplotlib.pyplot as plt
import cv2
import tensorflow as tf
import PIL.ImageOps as ImageOps
import PIL.Image as Image
import tqdm

In [ ]: class_names = ['Arive-Dantu',
                  'Basile',
                  'Crape_Jasmine',
                  'Curry',
                  'Brushtick',
                  'Guava',
                  'Indian_Beech',
                  'Jackfruit',
                  'Jawala_Cherry-Gasagape',
                  'Jasmine',
                  'Karanda',
                  'Lemon',
                  'Mango',
                  'Neem',
                  'Oleander',
                  'Peepal',
                  'Pomegranate',
                  'Rose_apple',
                  'Sandalwood',
                  'Tulsi']

class_names_label = {class_name: i for i, class_name in enumerate(class_names)}

nb_classes = len(class_names)

IMAGE_SIZE = (224, 224)
```

## Loading the Data

```
In [ ]: pre_process_img_path =
image = cv2.imread(img_path)
image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
image = cv2.resize(image, IMAGE_SIZE)
image = image

In [ ]: def load_data():
    datasets = ['train', 'test']
    output = []
    for dataset in datasets:
        images = []
        labels = []
        print('Loading {}').format(dataset)
        for folder in os.listdir(dataset):
            label = class_names_label[folder]
            for file in tqdm.os.listdir(os.path.join(dataset, folder)):
                img_path = os.path.join(os.path.join(dataset, folder), file)
                image = pre_process_img_path
                images.append(image)
                labels.append(label)
            images = np.array(images, dtype = 'float32')
            labels = np.array(labels, dtype = 'int32')
            output.append([images, labels])
        return output

In [ ]: (train_images, train_labels), (test_images, test_labels) = load_data()
```

## Checking the dataset

```
In [ ]: n_train = train_labels.shape[0]
n_test = test_labels.shape[0]

print ("Number of training examples: {}".format(n_train))
print ("Number of testing examples: {}".format(n_test))
print ("Each image is of size: {}".format(IMAGE_SIZE))
```

## Scaling the Data

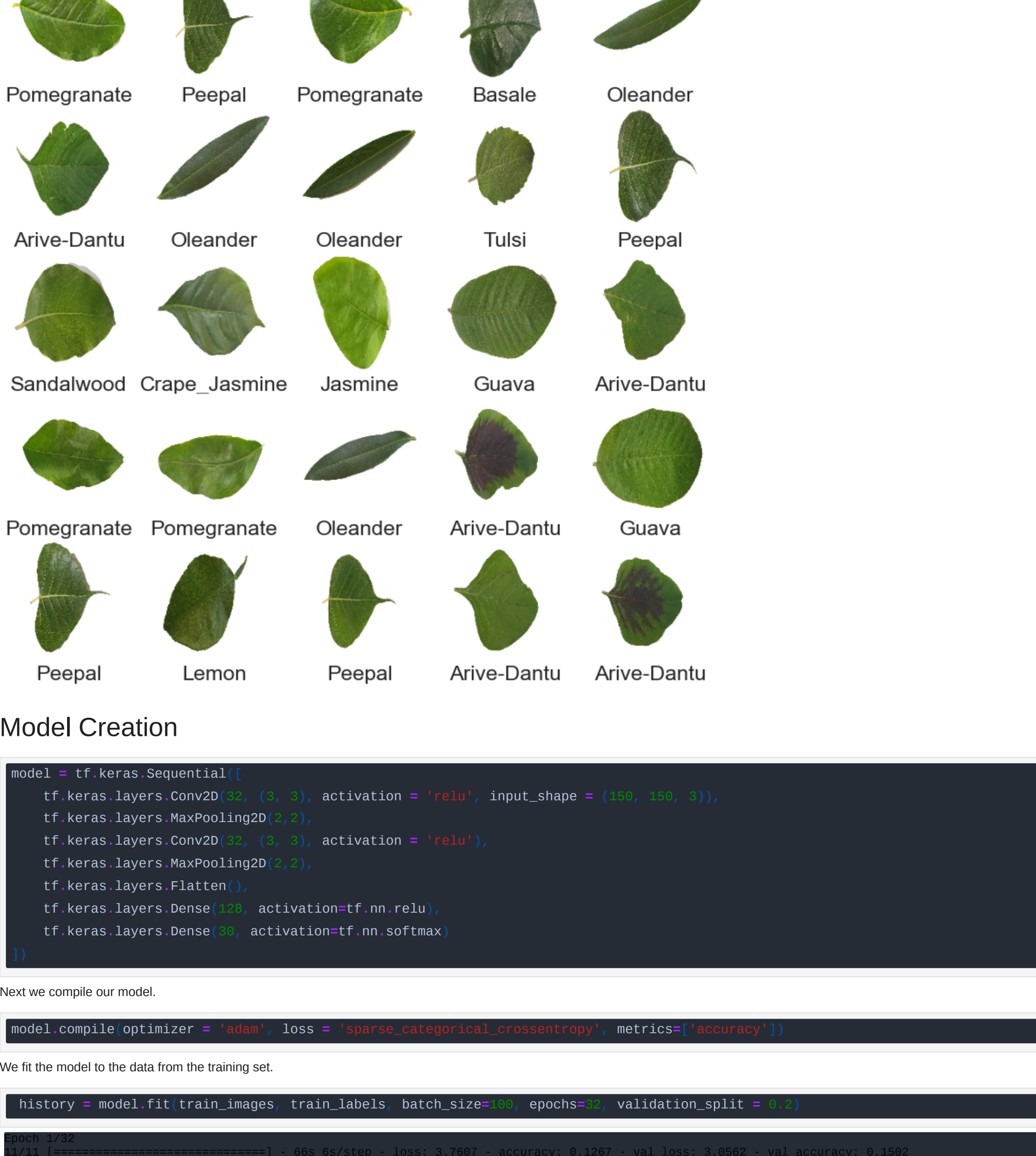
```
In [ ]: train_images = train_images / 255.0
test_images = test_images / 255.0
```

## Visualize the data

```
In [ ]: def display_examples(class_names, images, labels):
    fig = plt.figure(figsize=(8, 8))
    fig.suptitle('Some examples of images of the dataset', fontsize=8)
    for i in range(5):
        plt.subplot(5, 1, i+1)
        plt.xticks([])
        plt.yticks([])
        plt.grid(False)
        plt.imshow(images[i], cmap=plt.cm.binary)
        plt.xlabel(class_names[labels[i]])
    plt.show()

In [ ]: def display_random_image(class_names, images, labels):
    index = np.random.randint(images.shape[0])
    plt.figure()
    plt.imshow(images[index])
    plt.xticks([])
    plt.yticks([])
    plt.grid(False)
    plt.title('Image #{}: {}'.format(index, class_names[labels[index]]))
    plt.show()

In [ ]: display_examples(class_names, train_images, train_labels)
```



## Model Creation

```
In [ ]: model = tf.keras.Sequential([
    tf.keras.layers.Conv2D(32, (3, 3), activation = 'relu', input_shape = (224, 224, 3)),
    tf.keras.layers.MaxPooling2D((2, 2)),
    tf.keras.layers.Conv2D(64, (3, 3), activation = 'relu'),
    tf.keras.layers.MaxPooling2D((2, 2)),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(100, activation=tf.nn.relu),
    tf.keras.layers.Dense(10, activation=tf.nn.softmax)
])
```

Next we compile our model.

```
In [ ]: model.compile(optimizer = 'adam', loss = 'sparse_categorical_crossentropy', metrics = 'accuracy')
```

We fit the model to the data from the training set.

```
In [ ]: history = model.fit(train_images, train_labels, batch_size=32, epochs=30, validation_split = 0.1)

Epoch 1/30: 100%|#####| - 68s 5s/step - loss: 2.7887 - accuracy: 0.1287 - val_loss: 3.8562 - val_accuracy: 0.1592
Epoch 2/30: 100%|#####| - 68s 5s/step - loss: 2.6587 - accuracy: 0.2371 - val_loss: 2.2242 - val_accuracy: 0.4542
Epoch 3/30: 100%|#####| - 68s 5s/step - loss: 1.6995 - accuracy: 0.4739 - val_loss: 1.4129 - val_accuracy: 0.5861
Epoch 4/30: 100%|#####| - 67s 5s/step - loss: 0.9955 - accuracy: 0.7834 - val_loss: 1.0115 - val_accuracy: 0.6523
Epoch 5/30: 100%|#####| - 68s 5s/step - loss: 0.7119 - accuracy: 0.7879 - val_loss: 1.8225 - val_accuracy: 0.6638
Epoch 6/30: 100%|#####| - 68s 5s/step - loss: 0.5082 - accuracy: 0.8393 - val_loss: 0.7173 - val_accuracy: 0.7675
Epoch 7/30: 100%|#####| - 68s 5s/step - loss: 0.3331 - accuracy: 0.8903 - val_loss: 0.7488 - val_accuracy: 0.7675
Epoch 8/30: 100%|#####| - 62s 5s/step - loss: 0.2213 - accuracy: 0.9394 - val_loss: 0.7244 - val_accuracy: 0.7892
Epoch 9/30: 100%|#####| - 68s 5s/step - loss: 0.1813 - accuracy: 0.9513 - val_loss: 0.7051 - val_accuracy: 0.7875
Epoch 10/30: 100%|#####| - 62s 5s/step - loss: 0.1338 - accuracy: 0.9688 - val_loss: 0.6348 - val_accuracy: 0.8059
Epoch 11/30: 100%|#####| - 64s 5s/step - loss: 0.0955 - accuracy: 0.9770 - val_loss: 0.6513 - val_accuracy: 0.7849
Epoch 12/30: 100%|#####| - 64s 5s/step - loss: 0.0955 - accuracy: 0.9936 - val_loss: 0.6921 - val_accuracy: 0.8168
Epoch 13/30: 100%|#####| - 63s 5s/step - loss: 0.0393 - accuracy: 0.9972 - val_loss: 0.6672 - val_accuracy: 0.8095
Epoch 14/30: 100%|#####| - 65s 5s/step - loss: 0.0247 - accuracy: 1.0000 - val_loss: 0.6864 - val_accuracy: 0.7675
Epoch 15/30: 100%|#####| - 68s 5s/step - loss: 0.0176 - accuracy: 1.0000 - val_loss: 0.7999 - val_accuracy: 0.7675
Epoch 16/30: 100%|#####| - 68s 5s/step - loss: 0.0118 - accuracy: 1.0000 - val_loss: 0.8902 - val_accuracy: 0.8205
Epoch 17/30: 100%|#####| - 67s 5s/step - loss: 0.0085 - accuracy: 1.0000 - val_loss: 0.6966 - val_accuracy: 0.8132
Epoch 18/30: 100%|#####| - 68s 5s/step - loss: 0.0069 - accuracy: 1.0000 - val_loss: 0.7172 - val_accuracy: 0.8278
Epoch 19/30: 100%|#####| - 68s 5s/step - loss: 0.0059 - accuracy: 1.0000 - val_loss: 0.7395 - val_accuracy: 0.8095
Epoch 20/30: 100%|#####| - 63s 5s/step - loss: 0.0043 - accuracy: 1.0000 - val_loss: 0.7340 - val_accuracy: 0.8395
Epoch 21/30: 100%|#####| - 65s 5s/step - loss: 0.0034 - accuracy: 1.0000 - val_loss: 0.7467 - val_accuracy: 0.8095
Epoch 22/30: 100%|#####| - 65s 5s/step - loss: 0.0036 - accuracy: 1.0000 - val_loss: 0.7429 - val_accuracy: 0.8132
Epoch 23/30: 100%|#####| - 68s 5s/step - loss: 0.0026 - accuracy: 1.0000 - val_loss: 0.7695 - val_accuracy: 0.8278
Epoch 24/30: 100%|#####| - 68s 5s/step - loss: 0.0023 - accuracy: 1.0000 - val_loss: 0.7739 - val_accuracy: 0.8095
Epoch 25/30: 100%|#####| - 68s 5s/step - loss: 0.0021 - accuracy: 1.0000 - val_loss: 0.7717 - val_accuracy: 0.8095
Epoch 26/30: 100%|#####| - 65s 5s/step - loss: 0.0028 - accuracy: 1.0000 - val_loss: 0.7782 - val_accuracy: 0.8132
Epoch 27/30: 100%|#####| - 78s 5s/step - loss: 0.0019 - accuracy: 1.0000 - val_loss: 0.7765 - val_accuracy: 0.8095
Epoch 28/30: 100%|#####| - 67s 5s/step - loss: 0.0017 - accuracy: 1.0000 - val_loss: 0.7850 - val_accuracy: 0.8095
Epoch 29/30: 100%|#####| - 67s 5s/step - loss: 0.0014 - accuracy: 1.0000 - val_loss: 0.7843 - val_accuracy: 0.8095
Epoch 30/30: 100%|#####| - 67s 5s/step - loss: 0.0014 - accuracy: 1.0000 - val_loss: 0.7843 - val_accuracy: 0.8095
```

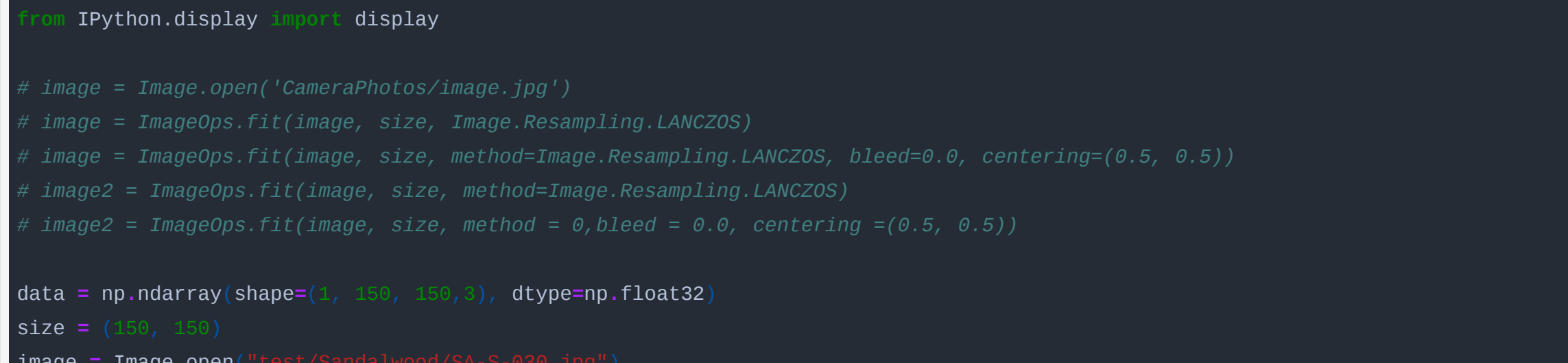
```
In [ ]: def plot_accuracy_loss(history):
    """
    We plot the accuracy and the loss during the training of the nn.
    """
    fig = plt.figure(figsize=(8, 8))

    # Plot accuracy
    plt.subplot(2, 1)
    plt.plot(history.history['accuracy'], 'bo--', label = "accuracy")
    plt.plot(history.history['val_accuracy'], 'ro--', label = "val_accuracy")
    plt.title('train_acc vs val_acc')
    plt.xlabel("epochs")
    plt.legend()

    # Plot loss function
    plt.subplot(2, 1)
    plt.plot(history.history['loss'], 'bo--', label = "loss")
    plt.plot(history.history['val_loss'], 'ro--', label = "val_loss")
    plt.title('train_loss vs val_loss')
    plt.xlabel("epochs")
    plt.legend()

    plt.show()
```

```
In [ ]: plot_accuracy_loss(history)
```



## Now we evaluate the model performance on the test set

```
In [ ]: test_loss = model.evaluate(test_images, test_labels)

In [ ]: predictions = model.predict(test_images)
pred_labels = np.argmax(predictions, axis = -1)

display_random_image(class_names, test_images, pred_labels)
```



```
In [ ]: from PIL import Image
import IPython.display as display

# image = Image.open('CameraPhotos/image.jpg')
# image = ImageOps.fit(image, size, Image.Resampling.LANCZOS)
# image2 = ImageOps.fit(image, size, method=Image.Resampling.LANCZOS, bleed=0.0, centering=(0.5, 0.5))
# image2 = ImageOps.fit(image, size, method=Image.Resampling.LANCZOS, bleed=0.0, centering=(0.5, 0.5))

data = np.ndarray(shape=(1, 224, 224, 3), dtype=np.float32)
size = 0
image = Image.open('test/Sandalwood/Sa-3-838.jpg')
image2 = image.resize(size, Image.Resampling.LANCZOS)
# image.show()
image_array = np.asarray(image2)
normalized_image_array = image_array.astype(np.float32) / 255.0
data[0] = normalized_image_array
prediction = model.predict(data)

pred_label = np.argmax(prediction, axis = -1)
class_prediction = class_names[pred_label[0]]
```

```
In [ ]: class_prediction
Out [ ]:
```

## Now we list the medicinal properties of the plant detected

```
In [ ]: class_prediction == 'Arive-Dantu':
    print('Arive-Dantu: Also known as Amarnath, this plant can be used as a food to eat when on diet or looking for weight loss as it is rich in fiber, extremely low in calories, have traces of fats and absolutely no cholesterol. It is used to help cure ulcers, diarrhea, swelling of mouth or throat and high cholesterol. It also has chemicals that act as antioxidants.')

class_prediction == 'Basile':
    print('Basile: Basile has an anti-inflammatory activity and wound healing ability. It can be helpful as a first aid, and the leaves of this plant can be crushed and applied to burns, scalds and wounds to help in healing of the wounds.')

class_prediction == 'Betel':
    print('Betel: The leaves of Betel possess immense therapeutic potential, and are often used in helping to cure mood swings and even depression. They are also quite an effective way to improve digestive health as they effectively neutralize pH imbalances in the stomach. The leaves are also full of many anti-microbial agents that combat the bacteria in your mouth.')

class_prediction == 'Crape_Jasmine':
    print('Crape_Jasmine: Jasmine is used in the curing of liver diseases, such as hepatitis, and in abdominal pain caused due to intense diarrhea, or dysentery. The smell of Jasmine flowers can be used to improve mood, reduce stress levels, and also to reduce food cravings. Jasmine can also be used to help in fighting skin diseases and speed up the process of wound healing.')

class_prediction == 'Curry':
    print('Curry: Curry leaves have immense nutritional value with low calories, and they help you fight nutritional deficiency of Vitamin A, Vitamin B, Vitamin C, Vitamin E, calcium and iron. It aids in digestion and helps in the treatment of morning sickness, nausea, and diarrhea. The leaves of this plant have properties that help in lowering blood cholesterol levels. It can also be used to promote hair growth and decrease the side effects of chemotherapy and radiotherapy.')

class_prediction == 'Brushtick':
    print('Brushtick: Brushtick contains high amounts of Vitamin C and antioxidants, which help you to build up your immune system and fight against common infections such as common cold and flu. Brushtick compounds in this plant help to relieve you from thickening of the arteries and lessens the chance of developing high blood pressure. As due to a high amount of calcium, Brushtick helps in developing strong and healthy bones.')

class_prediction == 'Fenugreek':
    print('Fenugreek: Fenugreek is commonly known as Methi in Indian households, Fenugreek is a plant with many medical abilities. It is said that Fenugreek can aid in metabolic condition such as diabetes and in regulating the blood sugar. Fenugreek has also been found to be as effective as antacid medications for heartburn. Due to its high nutritional value and less calories, it is also a food item to help prevent obesity.')

class_prediction == 'Guava':
    print('Guava: Aside from bearing a delicious taste, the fruit of the Guava tree is a rich source of Vitamin C and antioxidants. It is especially effective against preventing infections such as gastrointestinal infections, respiratory infections, oral/dental infections and skin infections. It can also aid in the treatment of Hypertension, Fever, Pain, Liver and Kidney diseases.')

class_prediction == 'Hibiscus':
    print('Hibiscus: The tea of the hibiscus flowers is quite prevalent and are used mainly to lower blood pressure and prevent hypertension. It is also used to relieve dry coughs. Some studies suggest that the tea has an effect in relieving from fever, diabetes, gallbladder attacks and even cancer. The roots of this plant can also be used to prepare a tonic.')

class_prediction == 'Indian_Beech':
    print('Indian_Beech: Popularly known as Karanja in India, the Indian Beech is a medicinal herb used mainly for skin disorders. Karanja oil is applied to the skin to manage boils, rashes and eczema as well as heal wounds as it has antimicrobial properties. The oil can also be useful in arthritis due to its anti-inflammatory activities.')

class_prediction == 'Indian_Mustard':
    print('Mustard: Mustard and its oil is widely used for the relief of joint pain, swelling, fever, coughs and colds. The mustard oil can be used as a massage oil, skin serum and for hair treatment. The oil can also be consumed, and as it is high in monounsaturated fatty acids, Mustard oil turns out to be a healthy choice for your heart.')

class_prediction == 'Jackfruit':
    print('Jackfruit: Jackfruits are full with Carotenoids, the yellow pigments that give jackfruit its characteristic colour. It is high in Vitamin A, which helps in preventing heart diseases and eye problems such as cataracts and macular degeneration and provides you with an excellent eyesight.')

class_prediction == 'Jawala_Cherry-Gasagape':
    print('Jawala_Cherry: The Jawala_Cherry plant have Anti-Diabetic properties which can potential cure type 2 diabetes. Jawala_Cherry tea contains rich amounts of nitric oxide, which relaxes blood vessels, reducing the chance of hypertension. Other than that, it can help to relieve pain, prevent infections, boost immunity and promote digestive health.')

class_prediction == 'Jamm':
    print('Jamm: The fruit extract of the Jamm plant is used in treating the common cold, cough and flu. The bark of this tree contain components like tannins and carbohydrates that can be used to fight dysentery. Jamm juice is used for treating sore throat problems and is also effective in the enlargement of the spleen')

class_prediction == 'Jasmine':
    print('Jasmine: Jasmine is used in the curing of liver diseases, such as hepatitis, and in abdominal pain caused due to intense diarrhea, or dysentery. The smell of Jasmine flowers can be used to improve mood, reduce stress levels, and also to reduce food cravings. Jasmine can also be used to help in fighting skin diseases and speed up the process of wound healing.')

class_prediction == 'Karanda':
    print('Karanda: Karanda is especially used in treating problems regarding digestion. It is used to cure worm infestation, gastritis, dermatitis, splenomegaly and indigestion. It is also useful for respiratory infections such as cough, cold, asthma, and even tuberculosis.')

class_prediction == 'Lemon':
    print('Lemon: Lemons are an excellent source of Vitamin C and fiber, and therefore, it lowers the risk factors leading to liver diseases. Lemons are also known to prevent Kidney Stones as they have Citric acid that helps in preventing Kidney Stones. Lemon, with Vitamin C and Citric acid helps in the absorption of iron.')

class_prediction == 'Mango':
    print('Mango: Known as King of Fruits by many, Mango is also packed with many medicinal properties. Mangoes have various vitamins, such as Vitamin C, E, A, and minerals such as Potassium and Magnesium. Mangoes are also rich in antioxidants, which can reduce the chances of Cancer. Mangoes are also known to promote digestive health and heart health too.')

class_prediction == 'Mexican_Mint':
    print('Mexican_Mint: Mexican Mint is a traditional remedy used to treat a variety of conditions. The leaves are a major part used for medicinal purposes. Mexican mint helps in curing respiratory illness, such as cold, sore throat, congestions, rummy nose, and also help in natural skincare.')

class_prediction == 'Mint':
    print('Mint: Mint is used usually in our daily lives to keep bad mouth odour at bay, but besides that, this plant also help in a variety of other functions such as relieving indigestion, and upset stomach, and can also improve Irritable Bowel Syndrome (IBS). Mint is also full of nutrients such as Vitamin A, Iron, Manganese, Potassium and Fiber.')

class_prediction == 'Neem':
    print('Neem: Prevalent in traditional remedies from a long time, Neem is considered as a boon for Mankind. It helps to cure many skin diseases such as Acne, fungal infections, dandruff, leprosy, and also nourishes and detoxifies the skin. It also boosts your immunity and act as an Insect and Mosquito Repellent. It helps to reduce joint pain as well and prevents Gastrointestinal diseases')

class_prediction == 'Oleander':
    print('Oleander: The use of this plant should be done extremely carefully, and never without the supervision of a doctor, as it can be a deadly poison. Despite the danger, oleander seeds and leaves are used with medicine. Oleander is used for heart conditions, asthma, epilepsy, cancer, leprosy, malaria, ringworm, indigestion, and venereal disease.')

class_prediction == 'Parijata':
    print('Parijata: Parijata plant is used for varying purposes. It shows anti-inflammatory and antipyretic (fever-reducing) properties which help in managing pain and fever. It is also used as a laxative, in rheumatism, skin ailments, and as a sedative. It is also said to provide relief from the symptoms of cough and cold. Drinking fresh Parijata leaves juice with honey helps to reduce the symptoms of fever.')

class_prediction == 'Peepal':
    print('Peepal: The bark of the Peepal tree, rich in vitamin K, is an effective complexion corrector and preserver. It also helps in various ailments such as strengthening blood capillaries, maintaining lipid levels, healing skin bruises faster, increasing skin resilience, treating pigmentation issues, wrinkles, dark circles, lightening surgery marks, scars, and stretch marks.')

class_prediction == 'Pomegranate':
    print('Pomegranate: Pomegranate has a variety of medical benefits. It is rich in antioxidants, which reduce inflammation, protect cells from damage and eventually lower the chances of Cancer. It is also a great source of Vitamin C and an immunity booster. Pomegranate has also shown to stall the progress of Alzheimer disease and protect memory.')

class_prediction == 'Raama':
    print('Raama: The Raama plant or its oil helps to reduce bone and joint pain and reduce the symptoms of rheumatoid arthritis. It can also be used to cure cough and cold, release mucus in the respiratory system and clear them, eventually facilitates easy breathing. Raama can also be applied to wounds to aid them in healing.')

class_prediction == 'Rose_apple':
    print('Rose_apple: Rose apples's seed and leaves are used for treating asthma and fever. Rose apples improve brain health and increase cognitive abilities. They are also effective against epilepsy, scabies, and inflammation in joints. They contain active and volatile compounds that have been connected with having anti-microbial and Anti-Fungal effects.')

class_prediction == 'Roshungh':
    print('Roshungh fig: Roshungh fig is noted for its big and round leaves. Leaves are crushed and the paste is applied on the wound. They are also used in diarrhea and dysentery.')

class_prediction == 'Sandalwood':
    print('Sandalwood: Sandalwood is used for treating the common cold, cough, bronchitis, fever, and sore mouth and throat. It is also used to treat urinary tract infections (UTIs), liver disease, gallbladder problems, heartstroke, gonorrhea, headache, and conditions of the heart and blood vessels (cardiovascular disease).')

class_prediction == 'Tulsi':
    print('Tulsi: Tulsi plant has the potential to cure a lot of ailments, and is used a lot in traditional remedies. Tulsi can help cure fever, to treat skin problems like acne, blackheads and premature ageing, to treat insect bites. Tulsi is also used to treat heart disease and fever, and respiratory problems.')

In [ ]: model.save('model_ser_16')
```

## Now we save our trained model so that we can use it