Name - Sejal Sanas

Task -3

Implement a support vector machine (SVM) to classify images of cats and dogs from the Kaggle dataset.

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns

        import tensorflow as tf
        from tensorflow.keras import layers, models
        from tensorflow.keras.preprocessing.image import ImageDataGenerator

        import warnings
        warnings.filterwarnings('ignore')
```

```
In [2]: from tensorflow.keras.preprocessing.image import ImageDataGenerator

        datagen = ImageDataGenerator(rescale=1./255, shear_range=0.2, zoom_range=0.2, horizontal_flip=True)

        training = datagen.flow_from_directory(
                "C:/Users/Shejal Sanas/Downloads/cats_dogs_light/cats_dogs_light/train",
                target_size=(64, 64),
                batch_size=32,
                class_mode="binary"
              )
```

```
Found 0 images belonging to 0 classes.
```

```
In [3]: datagen1 = ImageDataGenerator(rescale=1./255)

        test_set = datagen1.flow_from_directory(
                r"C:/Users/Shejal Sanas/Downloads/cats_dogs_light/cats_dogs_light/test",
                target_size=(64, 64),
                batch_size=32,
                class_mode="binary"
              )
```

```
Found 0 images belonging to 0 classes.
```

```
In [4]: from tensorflow.keras.layers import Conv2D
        from tensorflow.keras.layers import Dense
        from tensorflow.keras.regularizers import l2
```

```
In [5]: cnn = tf.keras.models.Sequential()
```

```
In [6]: cnn.add(tf.keras.layers.Conv2D(filters=32,padding="same",kernel_size=3, activation='relu', strides=2,
```

```
In [7]: cnn.compile(optimizer = 'adam', loss = 'hinge', metrics = ['accuracy'])
```

```
In [8]: cnn.add(tf.keras.layers.MaxPool2D(pool_size=2, strides=2))
```

```
In [9]: cnn.add(Dense(1, kernel_regularizer=tf.keras.regularizers.l2(0.01),activation
                     ='linear'))
```

```
In [10]: cnn.add(tf.keras.layers.Flatten())
```

```
In [11]: cnn.add(Dense(1, kernel_regularizer=tf.keras.regularizers.l2(0.01),activation
                      ='linear'))
```

```
In [12]: from tensorflow.keras.models import load_model

         cnn.save('./classification.h5')
```

```
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model
(model)`. This file format is considered legacy. We recommend using instead the native Keras format,
e.g. `model.save('my_model.keras')` or `keras.saving.save_model(model, 'my_model.keras')`.
```

In [13]:
```
from tensorflow.keras.preprocessing import image
test_image = image.load_img(r"C:/Users/Shejal Sanas/Downloads/cats_dogs_light/cats_dogs_light/test/dog
test_image = image.img_to_array(test_image)
test_image=test_image/255
test_image = np.expand_dims(test_image, axis = 0)
result = cnn.predict(test_image)
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 176ms/step
```

In [14]:
```
if result[0]<0:
    print("The image classified is cat")
else:
    print("The image classified is dog")
```

```
The image classified is dog
```

In [15]:
```
from tensorflow.keras.preprocessing import image
test_image = image.load_img(r"C:/Users/Shejal Sanas/Downloads/cats_dogs_light/cats_dogs_light/train/ca
test_image = image.img_to_array(test_image)
test_image=test_image/255
test_image = np.expand_dims(test_image, axis = 0)
result = cnn.predict(test_image)
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 47ms/step
```

In [16]:
```
if result[0]<0:
    print("The image classified is dog")
else:
    print("The image classified is cat")
```

```
The image classified is cat
```