[ ]:
```
import zipfile
from google.colab import drive

drive.mount('/content/drive/')

#poc_DATASET
zip_ref = zipfile.ZipFile("/content/drive/My Drive/sg_ff_filtered_red.zip", 'r')

#ISGI_20000_200gray_DATASET
# zip_ref = zipfile.ZipFile("/content/drive/My Drive/ISGI_dataset_200g.zip", 'r')

#ISGI_20000_200rgb_DATASET
#zip_ref = zipfile.ZipFile("/content/drive/My Drive/ISGI_dataset_200rgb.zip",␣
 ↪'r')

zip_ref.extractall("/tmp/")
zip_ref.close()
```

```
Mounted at /content/drive/
```

[ ]:
```

```

[ ]:
```
import os

base_dir = '/tmp/sg_ff_filtered_red'
train_dir = os.path.join(base_dir, 'train')
validation_dir = os.path.join(base_dir, 'validation')
```

```
# Directory with our training FlickerFaces pictures
train_ff_dir = os.path.join(train_dir, 'ff')

# Directory with our training StyleGAN pictures
train_sg_dir = os.path.join(train_dir, 'sg')

# Directory with our validation FlickerFaces pictures
validation_ff_dir = os.path.join(validation_dir, 'ff')

# Directory with our validation StyleGAN pictures
validation_sg_dir = os.path.join(validation_dir, 'sg')
```

[2]:
```
#imports
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout
from tensorflow.keras.backend import clear_session
import tensorflow as tf
tf.test.gpu_device_name()
```

[2]:
```
'/device:GPU:0'
```

[ ]:
```
print('Training_FlickerFaces images total: \t', len(os.listdir(train_ff_dir)))
print('Training_StyleGAN images total: \t', len(os.listdir(train_sg_dir)))
print('Validation_FlickerFaces images total: \t', len(os.
 ↪listdir(validation_ff_dir)))
print('Validation_StyleGAN images total: \t', len(os.listdir(validation_sg_dir)))
```

```
Training_FlickerFaces images total:      1000
Training_StyleGAN images total:          1000
Validation_FlickerFaces images total:    500
Validation_StyleGAN images total:        500
```

[1]:
```
!pip install optuna
```

```
Requirement already satisfied: optuna in /usr/local/lib/python3.7/dist-packages
(2.10.0)
Requirement already satisfied: colorlog in /usr/local/lib/python3.7/dist-
packages (from optuna) (6.5.0)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages
(from optuna) (1.19.5)
Requirement already satisfied: cliff in /usr/local/lib/python3.7/dist-packages
(from optuna) (3.9.0)
Requirement already satisfied: tqdm in /usr/local/lib/python3.7/dist-packages
(from optuna) (4.62.3)
Requirement already satisfied: cmaes>=0.8.2 in /usr/local/lib/python3.7/dist-
packages (from optuna) (0.8.2)
Requirement already satisfied: sqlalchemy>=1.1.0 in
/usr/local/lib/python3.7/dist-packages (from optuna) (1.4.25)
```

```
Requirement already satisfied: PyYAML in /usr/local/lib/python3.7/dist-packages
(from optuna) (3.13)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.7/dist-
packages (from optuna) (21.0)
Requirement already satisfied: alembic in /usr/local/lib/python3.7/dist-packages
(from optuna) (1.7.4)
Requirement already satisfied: scipy!=1.4.0 in /usr/local/lib/python3.7/dist-
packages (from optuna) (1.4.1)
Requirement already satisfied: pyparsing>=2.0.2 in
/usr/local/lib/python3.7/dist-packages (from packaging>=20.0->optuna) (2.4.7)
Requirement already satisfied: greenlet!=0.4.17 in
/usr/local/lib/python3.7/dist-packages (from sqlalchemy>=1.1.0->optuna) (1.1.2)
Requirement already satisfied: importlib-metadata in
/usr/local/lib/python3.7/dist-packages (from sqlalchemy>=1.1.0->optuna) (4.8.1)
Requirement already satisfied: importlib-resources in
/usr/local/lib/python3.7/dist-packages (from alembic->optuna) (5.2.2)
Requirement already satisfied: Mako in /usr/local/lib/python3.7/dist-packages
(from alembic->optuna) (1.1.5)
Requirement already satisfied: autopage>=0.4.0 in /usr/local/lib/python3.7/dist-
packages (from cliff->optuna) (0.4.0)
Requirement already satisfied: cmd2>=1.0.0 in /usr/local/lib/python3.7/dist-
packages (from cliff->optuna) (2.2.0)
Requirement already satisfied: pbr!=2.1.0,>=2.0.0 in
/usr/local/lib/python3.7/dist-packages (from cliff->optuna) (5.6.0)
Requirement already satisfied: stevedore>=2.0.1 in
/usr/local/lib/python3.7/dist-packages (from cliff->optuna) (3.5.0)
Requirement already satisfied: PrettyTable>=0.7.2 in
/usr/local/lib/python3.7/dist-packages (from cliff->optuna) (2.2.1)
Requirement already satisfied: pyperclip>=1.6 in /usr/local/lib/python3.7/dist-
packages (from cmd2>=1.0.0->cliff->optuna) (1.8.2)
Requirement already satisfied: attrs>=16.3.0 in /usr/local/lib/python3.7/dist-
packages (from cmd2>=1.0.0->cliff->optuna) (21.2.0)
Requirement already satisfied: wcwidth>=0.1.7 in /usr/local/lib/python3.7/dist-
packages (from cmd2>=1.0.0->cliff->optuna) (0.2.5)
Requirement already satisfied: typing-extensions in
/usr/local/lib/python3.7/dist-packages (from cmd2>=1.0.0->cliff->optuna)
(3.7.4.3)
Requirement already satisfied: colorama>=0.3.7 in /usr/local/lib/python3.7/dist-
packages (from cmd2>=1.0.0->cliff->optuna) (0.4.4)
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/dist-
packages (from importlib-metadata->sqlalchemy>=1.1.0->optuna) (3.6.0)
Requirement already satisfied: MarkupSafe>=0.9.2 in
/usr/local/lib/python3.7/dist-packages (from Mako->alembic->optuna) (2.0.1)
```

[3]:

```
import optuna
```

[ ]:
```python
dropout_rate = [0] * 2

def create_model(trial):

    num_layers = trial.suggest_int("num_layers", 1, 7)
    activation = trial.suggest_categorical("activation", ["relu"])
    dropout_rate[0] = trial.suggest_uniform('dropout_rate'+str(0), 0.0, 0.5)
    dropout_rate[1] = trial.suggest_uniform('dropout_rate'+str(1), 0.0, 0.5)
    mid_units = int(trial.suggest_discrete_uniform("mid_units", 100, 300, 100))
    filters=trial.suggest_categorical("filters", [16, 32, 64, 128])
    kernel_size=trial.suggest_categorical("kernel_size", [3, 3])
    strides=trial.suggest_categorical("strides", [1, 2])

    classifier = Sequential()

    #step 1  - Convolution Layers

    classifier.add(
        Conv2D(
            filters=filters,
            kernel_size=kernel_size,
            strides=1,
            activation = activation,
            input_shape=(200, 200, 3),
        )
    )

    classifier.add(MaxPooling2D(pool_size=(2, 2)))
    for i in range(1, num_layers):
        classifier.add(
            Conv2D(
                filters=filters,
                kernel_size=kernel_size,
                strides=1,
                activation = activation,
            )
        )
    classifier.add(MaxPooling2D(pool_size=(2, 2)))
    classifier.add(Dropout(dropout_rate[0]))
    classifier.add(Flatten())
    classifier.add(Dense(units = mid_units, activation = activation))
    classifier.add(Dropout(dropout_rate[1]))
    classifier.add(Dense(units = 1, activation ='sigmoid'))

    return classifier
```

```
[ ]: #image augumentation
     from keras.preprocessing.image import ImageDataGenerator

     #Data Preparation

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

     test_datagen = ImageDataGenerator(rescale = 1./255)

     training_set = train_datagen.flow_from_directory(train_dir,
                                                      target_size = (200, 200),
                                                      batch_size = 10,
                                                      class_mode = 'binary')

     test_set = test_datagen.flow_from_directory(validation_dir,
                                                 target_size = (200, 200),
                                                 batch_size = 10,
                                                 class_mode = 'binary')
```

```
Found 2000 images belonging to 2 classes.
Found 1000 images belonging to 2 classes.
```

```
[ ]: training_set
```

```
[ ]: <keras.preprocessing.image.DirectoryIterator at 0x7f4e91a45810>
```

```
[ ]: def objective(trial):

         optimizer = trial.suggest_categorical("optimizer", ["sgd", "adam",␣
      ↪"rmsprop", "adadelta", "adagrad", "adamax"])

         classifier = create_model(trial)

         classifier.compile(optimizer = optimizer, loss = 'binary_crossentropy',␣
      ↪metrics = ['accuracy'])

         history = classifier.fit(training_set,
                                  steps_per_epoch = 100, # num_samples // batch_size
                                  epochs = 5, # entire iteration over dataset
                                  validation_data = test_set,
                                  validation_steps = 50) #https://keras.io/api/models/
      ↪model_training_apis/
```

```
    classifier.save('/drive/MyDrive/Models/trialmodel_' + str(history.
 ↪history['val_accuracy'][-1]) +".h5")


    return history.history["val_accuracy"][-1]
```

[ ]:
```
import pickle

study = optuna.create_study(direction="maximize", )

#studypik = pickle.load(open('study.pickle', 'rb'))
study.optimize(objective, n_trials = 10, timeout = 60 * 60 * 3,␣
 ↪show_progress_bar=True)
print(studypik.best_params)
print(studypik.best_value)
pickle.dump(studypik, open('study.pickle', 'wb'))
```

```
[I 2021-10-31 13:27:41,338] A new study created in memory with name:
no-name-57069c1e-1b4e-4142-9822-35a68cbcfece
/usr/local/lib/python3.7/dist-packages/optuna/progress_bar.py:47:
ExperimentalWarning:

Progress bar is experimental (supported from v1.2.0). The interface can change
in the future.


  0%|          | 0/10 [00:00<?, ?it/s]

100/100 [==============================] - 47s 171ms/step - loss: 0.7051 -
accuracy: 0.5170 - val_loss: 0.6786 - val_accuracy: 0.5620
[I 2021-10-31 13:28:29,917] Trial 0 finished with value:
0.5619999766349792 and parameters: {'optimizer': 'adagrad', 'num_layers': 1,
'activation': 'relu', 'dropout_rate0': 0.47587234574775245, 'dropout_rate1':
0.46183774326599186, 'mid_units': 300.0, 'filters': 64, 'kernel_size': 3,
'strides': 2}. Best is trial 0 with value: 0.5619999766349792.
100/100 [==============================] - 17s 159ms/step - loss: 0.7403 -
accuracy: 0.5790 - val_loss: 0.6311 - val_accuracy: 0.6700
[I 2021-10-31 13:28:51,470] Trial 1 finished with value:
0.6700000166893005 and parameters: {'optimizer': 'adam', 'num_layers': 2,
'activation': 'relu', 'dropout_rate0': 0.26769020451155284, 'dropout_rate1':
0.3386348492768438, 'mid_units': 100.0, 'filters': 32, 'kernel_size': 3,
'strides': 1}. Best is trial 1 with value: 0.6700000166893005.
100/100 [==============================] - 16s 156ms/step - loss: 0.7847 -
accuracy: 0.5600 - val_loss: 0.6496 - val_accuracy: 0.6620
[I 2021-10-31 13:29:08,209] Trial 2 finished with value:
0.6620000004768372 and parameters: {'optimizer': 'adamax', 'num_layers': 2,
'activation': 'relu', 'dropout_rate0': 0.362046754103899, 'dropout_rate1':
0.37466293295311437, 'mid_units': 100.0, 'filters': 16, 'kernel_size': 3,
'strides': 2}. Best is trial 1 with value: 0.6700000166893005.
```

```
100/100 [==============================] - 17s 159ms/step - loss: 0.6857 -
accuracy: 0.5740 - val_loss: 0.6632 - val_accuracy: 0.6320
[I 2021-10-31 13:29:25,547] Trial 3 finished with value:
0.6320000290870667 and parameters: {'optimizer': 'sgd', 'num_layers': 7,
'activation': 'relu', 'dropout_rate0': 0.08134882935920718, 'dropout_rate1':
0.41244931428641857, 'mid_units': 100.0, 'filters': 16, 'kernel_size': 3,
'strides': 1}. Best is trial 1 with value: 0.6700000166893005.
100/100 [==============================] - 17s 160ms/step - loss: 0.6980 -
accuracy: 0.5210 - val_loss: 0.6867 - val_accuracy: 0.5340
[I 2021-10-31 13:29:47,219] Trial 4 finished with value:
0.5339999794960022 and parameters: {'optimizer': 'adam', 'num_layers': 6,
'activation': 'relu', 'dropout_rate0': 0.2847882512202252, 'dropout_rate1':
0.03429756052223176, 'mid_units': 200.0, 'filters': 16, 'kernel_size': 3,
'strides': 1}. Best is trial 1 with value: 0.6700000166893005.
100/100 [==============================] - 17s 162ms/step - loss: 0.8187 -
accuracy: 0.5580 - val_loss: 0.6360 - val_accuracy: 0.6600
[I 2021-10-31 13:30:04,934] Trial 5 finished with value:
0.6600000262260437 and parameters: {'optimizer': 'adam', 'num_layers': 2,
'activation': 'relu', 'dropout_rate0': 0.11358935025708411, 'dropout_rate1':
0.3094802571565712, 'mid_units': 100.0, 'filters': 64, 'kernel_size': 3,
'strides': 1}. Best is trial 1 with value: 0.6700000166893005.
100/100 [==============================] - 21s 189ms/step - loss: 0.6956 -
accuracy: 0.4830 - val_loss: 0.6936 - val_accuracy: 0.4700
[I 2021-10-31 13:30:27,587] Trial 6 finished with value:
0.4699999988079071 and parameters: {'optimizer': 'adam', 'num_layers': 7,
'activation': 'relu', 'dropout_rate0': 0.19060102520992955, 'dropout_rate1':
0.4522466144468808, 'mid_units': 300.0, 'filters': 64, 'kernel_size': 3,
'strides': 2}. Best is trial 1 with value: 0.6700000166893005.
100/100 [==============================] - 18s 167ms/step - loss: 0.6928 -
accuracy: 0.5370 - val_loss: 0.6804 - val_accuracy: 0.6020
[I 2021-10-31 13:30:49,324] Trial 7 finished with value:
0.6019999980926514 and parameters: {'optimizer': 'adadelta', 'num_layers': 4,
'activation': 'relu', 'dropout_rate0': 0.22387331439595765, 'dropout_rate1':
0.014492194696432759, 'mid_units': 200.0, 'filters': 32, 'kernel_size': 3,
'strides': 2}. Best is trial 1 with value: 0.6700000166893005.
100/100 [==============================] - 18s 167ms/step - loss: 0.8435 -
accuracy: 0.5600 - val_loss: 0.6537 - val_accuracy: 0.6600
[I 2021-10-31 13:31:11,492] Trial 8 finished with value:
0.6600000262260437 and parameters: {'optimizer': 'rmsprop', 'num_layers': 5,
'activation': 'relu', 'dropout_rate0': 0.49238462917243947, 'dropout_rate1':
0.006749682377091726, 'mid_units': 200.0, 'filters': 32, 'kernel_size': 3,
'strides': 1}. Best is trial 1 with value: 0.6700000166893005.
100/100 [==============================] - 16s 158ms/step - loss: 0.6941 -
accuracy: 0.5090 - val_loss: 0.6923 - val_accuracy: 0.4940
[I 2021-10-31 13:31:28,309] Trial 9 finished with value:
0.49399998784065247 and parameters: {'optimizer': 'adadelta', 'num_layers': 2,
'activation': 'relu', 'dropout_rate0': 0.18028263462627542, 'dropout_rate1':
0.40114405301463785, 'mid_units': 100.0, 'filters': 32, 'kernel_size': 3,
'strides': 2}. Best is trial 1 with value: 0.6700000166893005.
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-21-f4463db52685> in <module>()
      5 #studypik = pickle.load(open('study.pickle', 'rb'))
      6 study.optimize(objective, n_trials = 10, timeout = 60 * 60 * 3,␣
 ↪show_progress_bar=True)
----> 7 print(studypik.best_params)
      8 print(studypik.best_value)
      9 pickle.dump(studypik, open('study.pickle', 'wb'))

NameError: name 'studypik' is not defined
```

[ ]:
```
study = optuna.create_study(direction="maximize", )
study.optimize(objective, n_trials = 10, timeout = 60 * 60 * 3,␣
 ↪show_progress_bar=True)
print(study.best_params)
print(study.best_value)
```

```
[I 2021-10-31 13:33:28,764] A new study created in memory with name:
no-name-4f7005ab-015a-4fe3-a772-92753c5144dc
/usr/local/lib/python3.7/dist-packages/optuna/progress_bar.py:47:
ExperimentalWarning:

Progress bar is experimental (supported from v1.2.0). The interface can change
in the future.


  0%|          | 0/10 [00:00<?, ?it/s]

Epoch 1/5
100/100 [==============================] - 19s 186ms/step - loss: 0.6906 -
accuracy: 0.5480 - val_loss: 0.6855 - val_accuracy: 0.6320
Epoch 2/5
100/100 [==============================] - 18s 183ms/step - loss: 0.6852 -
accuracy: 0.5540 - val_loss: 0.6794 - val_accuracy: 0.6180
Epoch 3/5
100/100 [==============================] - 18s 182ms/step - loss: 0.6731 -
accuracy: 0.6130 - val_loss: 0.6787 - val_accuracy: 0.5360
Epoch 4/5
100/100 [==============================] - 18s 183ms/step - loss: 0.6542 -
accuracy: 0.6030 - val_loss: 0.6498 - val_accuracy: 0.6100
Epoch 5/5
100/100 [==============================] - 18s 181ms/step - loss: 0.6419 -
accuracy: 0.6370 - val_loss: 0.6408 - val_accuracy: 0.6420
[I 2021-10-31 13:35:06,407] Trial 0 finished with value:
0.6420000195503235 and parameters: {'optimizer': 'adagrad', 'num_layers': 5,
'activation': 'relu', 'dropout_rate0': 0.15440207381298776, 'dropout_rate1':
0.3946011403479582, 'mid_units': 300.0, 'filters': 64, 'kernel_size': 3,
'strides': 2}. Best is trial 0 with value: 0.6420000195503235.
Epoch 1/5
100/100 [==============================] - 26s 223ms/step - loss: 0.6917 -
accuracy: 0.5360 - val_loss: 0.6913 - val_accuracy: 0.5040
Epoch 2/5
100/100 [==============================] - 22s 223ms/step - loss: 0.6783 -
accuracy: 0.5700 - val_loss: 0.6623 - val_accuracy: 0.6460
Epoch 3/5
100/100 [==============================] - 22s 222ms/step - loss: 0.6669 -
accuracy: 0.6200 - val_loss: 0.6478 - val_accuracy: 0.6420
Epoch 4/5
100/100 [==============================] - 22s 221ms/step - loss: 0.6435 -
accuracy: 0.6240 - val_loss: 0.6571 - val_accuracy: 0.6080
Epoch 5/5
100/100 [==============================] - 22s 220ms/step - loss: 0.6391 -
accuracy: 0.6400 - val_loss: 0.6108 - val_accuracy: 0.6800
[I 2021-10-31 13:38:15,804] Trial 1 finished with value:
0.6800000071525574 and parameters: {'optimizer': 'adagrad', 'num_layers': 4,
'activation': 'relu', 'dropout_rate0': 0.44497538687245153, 'dropout_rate1':
0.403020754907426, 'mid_units': 300.0, 'filters': 128, 'kernel_size': 3,
'strides': 2}. Best is trial 1 with value: 0.6800000071525574.
Epoch 1/5
100/100 [==============================] - 34s 311ms/step - loss: 0.6928 -
accuracy: 0.5190 - val_loss: 0.6922 - val_accuracy: 0.5120
Epoch 2/5
100/100 [==============================] - 31s 308ms/step - loss: 0.6913 -
accuracy: 0.5450 - val_loss: 0.6922 - val_accuracy: 0.4860
Epoch 3/5
100/100 [==============================] - 31s 308ms/step - loss: 0.6893 -
accuracy: 0.5700 - val_loss: 0.6886 - val_accuracy: 0.5800
Epoch 4/5
100/100 [==============================] - 32s 317ms/step - loss: 0.6833 -
accuracy: 0.5850 - val_loss: 0.6765 - val_accuracy: 0.6360
Epoch 5/5
100/100 [==============================] - 31s 308ms/step - loss: 0.6683 -
accuracy: 0.5910 - val_loss: 0.6603 - val_accuracy: 0.6320
[I 2021-10-31 13:41:13,413] Trial 2 finished with value:
0.6320000290870667 and parameters: {'optimizer': 'adagrad', 'num_layers': 7,
'activation': 'relu', 'dropout_rate0': 0.10574849589901586, 'dropout_rate1':
0.41902846877861305, 'mid_units': 300.0, 'filters': 128, 'kernel_size': 3,
'strides': 2}. Best is trial 1 with value: 0.6800000071525574.
Epoch 1/5
100/100 [==============================] - 16s 159ms/step - loss: 0.6974 -
accuracy: 0.4890 - val_loss: 0.6933 - val_accuracy: 0.5260
Epoch 2/5
100/100 [==============================] - 16s 161ms/step - loss: 0.6939 -
```

```
accuracy: 0.5240 - val_loss: 0.6920 - val_accuracy: 0.5540
Epoch 3/5
100/100 [==============================] - 16s 158ms/step - loss: 0.6959 -
accuracy: 0.5030 - val_loss: 0.6881 - val_accuracy: 0.5580
Epoch 4/5
100/100 [==============================] - 16s 157ms/step - loss: 0.6872 -
accuracy: 0.5560 - val_loss: 0.6870 - val_accuracy: 0.5680
Epoch 5/5
100/100 [==============================] - 16s 156ms/step - loss: 0.6877 -
accuracy: 0.5490 - val_loss: 0.6867 - val_accuracy: 0.5820
[I 2021-10-31 13:42:51,866] Trial 3 finished with value:
0.5820000171661377 and parameters: {'optimizer': 'adadelta', 'num_layers': 2,
'activation': 'relu', 'dropout_rate0': 0.3385236238961555, 'dropout_rate1':
0.3040985969239738, 'mid_units': 100.0, 'filters': 16, 'kernel_size': 3,
'strides': 1}. Best is trial 1 with value: 0.6800000071525574.
Epoch 1/5
100/100 [==============================] - 18s 168ms/step - loss: 0.7055 -
accuracy: 0.5030 - val_loss: 0.6708 - val_accuracy: 0.5160
Epoch 2/5
100/100 [==============================] - 16s 164ms/step - loss: 0.6135 -
accuracy: 0.6810 - val_loss: 0.5086 - val_accuracy: 0.7460
Epoch 3/5
100/100 [==============================] - 16s 163ms/step - loss: 0.5094 -
accuracy: 0.7620 - val_loss: 0.4488 - val_accuracy: 0.8080
Epoch 4/5
100/100 [==============================] - 16s 165ms/step - loss: 0.4797 -
accuracy: 0.7950 - val_loss: 0.4229 - val_accuracy: 0.8180
Epoch 5/5
100/100 [==============================] - 17s 165ms/step - loss: 0.4476 -
accuracy: 0.8120 - val_loss: 0.7007 - val_accuracy: 0.6040
[I 2021-10-31 13:44:16,156] Trial 4 finished with value:
0.6039999723434448 and parameters: {'optimizer': 'rmsprop', 'num_layers': 7,
'activation': 'relu', 'dropout_rate0': 0.49644395894413873, 'dropout_rate1':
0.1535162204490153, 'mid_units': 200.0, 'filters': 16, 'kernel_size': 3,
'strides': 1}. Best is trial 1 with value: 0.6800000071525574.
Epoch 1/5
100/100 [==============================] - 18s 174ms/step - loss: 0.6892 -
accuracy: 0.5620 - val_loss: 0.6748 - val_accuracy: 0.5600
Epoch 2/5
100/100 [==============================] - 17s 173ms/step - loss: 0.6716 -
accuracy: 0.6180 - val_loss: 0.6141 - val_accuracy: 0.6860
Epoch 3/5
100/100 [==============================] - 17s 173ms/step - loss: 0.6019 -
accuracy: 0.6880 - val_loss: 0.5594 - val_accuracy: 0.7180
Epoch 4/5
100/100 [==============================] - 17s 174ms/step - loss: 0.5672 -
accuracy: 0.7140 - val_loss: 0.5455 - val_accuracy: 0.7320
Epoch 5/5
```

```
100/100 [==============================] - 18s 176ms/step - loss: 0.5361 -
accuracy: 0.7350 - val_loss: 0.4644 - val_accuracy: 0.7800
[I 2021-10-31 13:45:56,903] Trial 5 finished with value:
0.7799999713897705 and parameters: {'optimizer': 'adamax', 'num_layers': 7,
'activation': 'relu', 'dropout_rate0': 0.1414120827493418, 'dropout_rate1':
0.4441663934981933, 'mid_units': 100.0, 'filters': 32, 'kernel_size': 3,
'strides': 2}. Best is trial 5 with value: 0.7799999713897705.
Epoch 1/5
100/100 [==============================] - 18s 172ms/step - loss: 0.6896 -
accuracy: 0.5500 - val_loss: 0.6764 - val_accuracy: 0.6560
Epoch 2/5
100/100 [==============================] - 17s 170ms/step - loss: 0.6705 -
accuracy: 0.5990 - val_loss: 0.6570 - val_accuracy: 0.6440
Epoch 3/5
100/100 [==============================] - 17s 166ms/step - loss: 0.6546 -
accuracy: 0.6200 - val_loss: 0.6586 - val_accuracy: 0.6000
Epoch 4/5
100/100 [==============================] - 17s 166ms/step - loss: 0.6275 -
accuracy: 0.6670 - val_loss: 0.5969 - val_accuracy: 0.6900
Epoch 5/5
100/100 [==============================] - 17s 167ms/step - loss: 0.6341 -
accuracy: 0.6250 - val_loss: 0.6041 - val_accuracy: 0.6820
[I 2021-10-31 13:47:37,215] Trial 6 finished with value:
0.6819999814033508 and parameters: {'optimizer': 'adagrad', 'num_layers': 4,
'activation': 'relu', 'dropout_rate0': 0.4475620563107447, 'dropout_rate1':
0.3007601379030714, 'mid_units': 300.0, 'filters': 32, 'kernel_size': 3,
'strides': 1}. Best is trial 5 with value: 0.7799999713897705.
Epoch 1/5
100/100 [==============================] - 19s 188ms/step - loss: 0.6909 -
accuracy: 0.5350 - val_loss: 0.6859 - val_accuracy: 0.5540
Epoch 2/5
100/100 [==============================] - 18s 184ms/step - loss: 0.6823 -
accuracy: 0.5710 - val_loss: 0.6666 - val_accuracy: 0.6460
Epoch 3/5
100/100 [==============================] - 19s 185ms/step - loss: 0.6710 -
accuracy: 0.5820 - val_loss: 0.6537 - val_accuracy: 0.6200
Epoch 4/5
100/100 [==============================] - 18s 181ms/step - loss: 0.6570 -
accuracy: 0.6200 - val_loss: 0.6536 - val_accuracy: 0.5960
Epoch 5/5
100/100 [==============================] - 18s 180ms/step - loss: 0.6510 -
accuracy: 0.6410 - val_loss: 0.6270 - val_accuracy: 0.6540
[I 2021-10-31 13:49:13,076] Trial 7 finished with value:
0.6539999842643738 and parameters: {'optimizer': 'adagrad', 'num_layers': 4,
'activation': 'relu', 'dropout_rate0': 0.2256719112217388, 'dropout_rate1':
0.45643380402886063, 'mid_units': 300.0, 'filters': 64, 'kernel_size': 3,
'strides': 1}. Best is trial 5 with value: 0.7799999713897705.
Epoch 1/5
```

```
100/100 [==============================] - 18s 176ms/step - loss: 0.6936 -
accuracy: 0.5080 - val_loss: 0.6952 - val_accuracy: 0.4860
Epoch 2/5
100/100 [==============================] - 18s 175ms/step - loss: 0.6927 -
accuracy: 0.5030 - val_loss: 0.6909 - val_accuracy: 0.5300
Epoch 3/5
100/100 [==============================] - 17s 173ms/step - loss: 0.6904 -
accuracy: 0.5370 - val_loss: 0.6917 - val_accuracy: 0.5180
Epoch 4/5
100/100 [==============================] - 17s 172ms/step - loss: 0.6918 -
accuracy: 0.5140 - val_loss: 0.6918 - val_accuracy: 0.4980
Epoch 5/5
100/100 [==============================] - 17s 171ms/step - loss: 0.6890 -
accuracy: 0.5540 - val_loss: 0.6913 - val_accuracy: 0.5200
[I 2021-10-31 13:50:53,707] Trial 8 finished with value:
0.5199999809265137 and parameters: {'optimizer': 'adadelta', 'num_layers': 7,
'activation': 'relu', 'dropout_rate0': 0.15795313983038067, 'dropout_rate1':
0.3708922576280311, 'mid_units': 200.0, 'filters': 32, 'kernel_size': 3,
'strides': 2}. Best is trial 5 with value: 0.7799999713897705.
Epoch 1/5
100/100 [==============================] - 19s 183ms/step - loss: 1.1202 -
accuracy: 0.5830 - val_loss: 0.5884 - val_accuracy: 0.7480
Epoch 2/5
100/100 [==============================] - 18s 177ms/step - loss: 0.5767 -
accuracy: 0.7030 - val_loss: 0.5959 - val_accuracy: 0.7120
Epoch 3/5
100/100 [==============================] - 18s 179ms/step - loss: 0.5101 -
accuracy: 0.7640 - val_loss: 0.5140 - val_accuracy: 0.8020
Epoch 4/5
100/100 [==============================] - 18s 178ms/step - loss: 0.5006 -
accuracy: 0.7680 - val_loss: 0.4608 - val_accuracy: 0.7780
Epoch 5/5
100/100 [==============================] - 18s 177ms/step - loss: 0.4943 -
accuracy: 0.7680 - val_loss: 0.5080 - val_accuracy: 0.7700
[I 2021-10-31 13:52:36,942] Trial 9 finished with value:
0.7699999809265137 and parameters: {'optimizer': 'adam', 'num_layers': 1,
'activation': 'relu', 'dropout_rate0': 0.13128267203628347, 'dropout_rate1':
0.360700887587538, 'mid_units': 200.0, 'filters': 128, 'kernel_size': 3,
'strides': 1}. Best is trial 5 with value: 0.7799999713897705.
{'optimizer': 'adamax', 'num_layers': 7, 'activation': 'relu', 'dropout_rate0':
0.1414120827493418, 'dropout_rate1': 0.4441663934981933, 'mid_units': 100.0,
'filters': 32, 'kernel_size': 3, 'strides': 2}
0.7799999713897705
```

[ ]:
```
print(study.best_params)
print(study.best_value)
```

[ ]:
```
fig = optuna.visualization.plot_optimization_history(study)
fig.show()
```

[ ]:
```
fig = optuna.visualization.plot_param_importances(study)
fig.show()
```

[ ]:
```
print(studypik.best_params)
print(studypik.best_value)
pickle.dump(studypik, open('study.pickle', 'wb'))
```

```
{'optimizer': 'adam', 'num_layers': 2, 'activation': 'linear', 'dropout_rate0':
0.19368366673208176, 'dropout_rate1': 0.3373004012393455, 'mid_units': 100.0,
'filters': 32, 'kernel_size': 3, 'strides': 2}
0.8974999785423279
```

[ ]:
```
print("Number of finished trials: {}".format(len(study.trials)))

print("Best trial:")
trial = study.best_trial

print("  Value: {}".format(trial.value))

print("  Params: ")
for key, value in trial.params.items():
  print("    {}: {}".format(key, value))
```

```
Number of finished trials: 5
Best trial:
  Value: 0.8725000023841858
  Params:
    optimizer: adam
    num_layers: 6
    activation: linear
    dropout_rate0: 0.3845052526638556
    dropout_rate1: 0.2421332530661529
    mid_units: 100.0
    filters: 16
    kernel_size: 3
    strides: 2
```

[ ]:
```
import pickle

studypik = pickle.load(open('study.pickle', 'rb'))
print(studypik.best_params)
print(studypik.best_value)
pickle.dump(studypik, open('study.pickle', 'wb'))
```

[4]:
```
!pip install pyyaml h5py
```

```
Requirement already satisfied: pyyaml in /usr/local/lib/python3.7/dist-packages
(3.13)
Requirement already satisfied: h5py in /usr/local/lib/python3.7/dist-packages
(3.1.0)
Requirement already satisfied: numpy>=1.14.5 in /usr/local/lib/python3.7/dist-
packages (from h5py) (1.19.5)
Requirement already satisfied: cached-property in /usr/local/lib/python3.7/dist-
packages (from h5py) (1.5.2)
```

[5]:
```
import os

import tensorflow as tf
from tensorflow import keras

print(tf.version.VERSION)
```

```
2.6.0
```

[6]:
```
new_model = tf.keras.models.load_model('/content/drive/MyDrive/optunam.h5')

# Check its architecture
new_model.summary()
```

```
Model: "sequential_8"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_36 (Conv2D)           (None, 198, 198, 64)      1792
_________________________________________________________________
max_pooling2d_16 (MaxPooling (None, 99, 99, 64)        0
_________________________________________________________________
conv2d_37 (Conv2D)           (None, 97, 97, 64)        36928
_________________________________________________________________
conv2d_38 (Conv2D)           (None, 95, 95, 64)        36928
_________________________________________________________________
conv2d_39 (Conv2D)           (None, 93, 93, 64)        36928
_________________________________________________________________
conv2d_40 (Conv2D)           (None, 91, 91, 64)        36928
_________________________________________________________________
conv2d_41 (Conv2D)           (None, 89, 89, 64)        36928
_________________________________________________________________
max_pooling2d_17 (MaxPooling (None, 44, 44, 64)        0
_________________________________________________________________
dropout_16 (Dropout)         (None, 44, 44, 64)        0
_________________________________________________________________
flatten_8 (Flatten)          (None, 123904)            0
_________________________________________________________________
dense_16 (Dense)             (None, 300)               37171500
_________________________________________________________________
```

```
dropout_17 (Dropout)         (None, 300)               0
_________________________________________________________________
dense_17 (Dense)             (None, 1)                 301
=================================================================
Total params: 37,358,233
Trainable params: 37,358,233
Non-trainable params: 0
_________________________________________________________________
```

[ ]:
```
import os


model = tf.keras.models.load_model("/content/trialmodel_0.9764999747276306.h5")
```

```
---------------------------------------------------------------------------
OSError                                   Traceback (most recent call last)
<ipython-input-7-7c7ae5ed1699> in <module>()
      1 import os
      2
----> 3 model = tf.keras.models.load_model("/content/trialmodel_0.97649997472763
↪h5")

/usr/local/lib/python3.7/dist-packages/keras/saving/save.py in␣
↪load_model(filepath, custom_objects, compile, options)
    199             (isinstance(filepath, h5py.File) or h5py.is_hdf5(filepath))
    200           return hdf5_format.load_model_from_hdf5(filepath, custom_object
--> 201                                                    compile)

    202
    203         filepath = path_to_string(filepath)

/usr/local/lib/python3.7/dist-packages/keras/saving/hdf5_format.py in␣
↪load_model_from_hdf5(filepath, custom_objects, compile)
    165   opened_new_file = not isinstance(filepath, h5py.File)
    166   if opened_new_file:
--> 167     f = h5py.File(filepath, mode='r')
    168   else:
    169     f = filepath

/usr/local/lib/python3.7/dist-packages/h5py/_hl/files.py in __init__(self, name,␣
↪mode, driver, libver, userblock_size, swmr, rdcc_nslots, rdcc_nbytes, rdcc_w0,
↪track_order, fs_strategy, fs_persist, fs_threshold, **kwds)
    425                                fapl,␣
↪fcpl=make_fcpl(track_order=track_order, fs_strategy=fs_strategy,
    426                                fs_persist=fs_persist,␣
↪fs_threshold=fs_threshold),
--> 427                                swmr=swmr)

    428
    429             if isinstance(libver, tuple):

/usr/local/lib/python3.7/dist-packages/h5py/_hl/files.py in make_fid(name, mode,␣
 ↪userblock_size, fapl, fcpl, swmr)
    188         if swmr and swmr_support:
    189             flags |= h5f.ACC_SWMR_READ
--> 190         fid = h5f.open(name, flags, fapl=fapl)
    191     elif mode == 'r+':
    192         fid = h5f.open(name, h5f.ACC_RDWR, fapl=fapl)

h5py/_objects.pyx in h5py._objects.with_phil.wrapper()

h5py/_objects.pyx in h5py._objects.with_phil.wrapper()

h5py/h5f.pyx in h5py.h5f.open()

OSError: Unable to open file (truncated file: eof = 14680064, sblock->base_addr =␣
 ↪0, stored_eof = 448379760)
```

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: