```
In [ ]: # # make a prediction for a new image.
        # from tensorflow.keras.preprocessing.image import load_img
        # from tensorflow.keras.preprocessing.image import img_to_array

        # images = []
        # # load and prepare the image
        # for x in range(len(df_scrub)):
        #     file_type = '.jpg'
        #     s = 'knife_images/' + str(x) + file_type
        #     # load the image
        #     img = load_img(s, target_size=(128, 128))
        #     # convert to array
        #     img = img_to_array(img)
        #     # reshape into a single sample with 3 channels
        #     img = img.reshape(1, 128, 128, 3)
        #     # center pixel data
        #     img = img.astype('float32')
        #     images.append(img)
```

```
In [ ]: # from sklearn.model_selection import train_test_split
```

```
In [38]: import numpy as np
         import pandas as pd
         import tensorflow as tf
         from tensorflow.keras.layers import Input, Dropout, Conv2D, Dense, Flatten, Globa
```

```
In [27]: from keras import models
         from keras import layers
```

```
In [ ]: X = tf.convert_to_tensor(image_list)
```

```
In [31]: y = df_CNN_regression['profit']
```

```
In [32]: X_val = X[4918:5971]
         y_val = y[4918:5971]
```

```
In [33]: X_train = X[:4918]
         y_train = y[:4918]
```

```
In [34]: X_test = X[5971:]
         y_test = y[5971:]
```

```
In [35]: display(len(X_val)/len(X))
         display(len(X_train)/len(X))
         len(X_test)/len(X)

         0.1498932384341637

         0.7000711743772242

Out[35]: 0.1500355871886121
```

In [41]:
```
model = models.Sequential()

model.add(layers.Conv2D(32, (3, 3), padding='same', activation='relu',
                        input_shape=(224 ,224,  3)))
model.add(layers.BatchNormalization())

model.add(layers.Conv2D(32, (3, 3), activation='relu', padding='same'))
model.add(layers.BatchNormalization())
model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Conv2D(64, (3, 3), activation='relu', padding='same'))
model.add(layers.BatchNormalization())

model.add(layers.Conv2D(64, (3, 3), activation='relu', padding='same'))
model.add(layers.BatchNormalization())
model.add(layers.Conv2D(64, (3, 3), activation='relu', padding='same'))
model.add(layers.BatchNormalization())
model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Conv2D(128, (3, 3), activation='relu', padding='same'))
model.add(layers.BatchNormalization())
model.add(layers.Conv2D(128, (3, 3), activation='relu', padding='same'))
model.add(layers.BatchNormalization())
model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Flatten())

model.add(Dense(512, activation='relu'))
model.add(Dropout(0.1))

model.add(Dense(256, activation='relu'))
model.add(Dense(128, activation='relu'))

model.add(Dense(1, activation='linear'))

model.compile(loss='mean_squared_error',
              optimizer='Adam',
               metrics=['mse'])
history = model.fit(X_train,
                    y_train,
                    epochs=32,
                    batch_size=300,
                    validation_data=(X_val, y_val))
```

```
Epoch 1/32
17/17 [==============================] - 678s 40s/step - loss: 10665.1348 - m
se: 10665.1348 - val_loss: 4539.1099 - val_mse: 4539.1099
Epoch 2/32
17/17 [==============================] - 669s 39s/step - loss: 2237.0037 - ms
e: 2237.0037 - val_loss: 4222.4707 - val_mse: 4222.4702
Epoch 3/32
17/17 [==============================] - 673s 40s/step - loss: 2073.3645 - ms
e: 2073.3645 - val_loss: 4098.2490 - val_mse: 4098.2490
Epoch 4/32
17/17 [==============================] - 674s 40s/step - loss: 1883.7783 - ms
e: 1883.7783 - val_loss: 4585.5220 - val_mse: 4585.5220
```

```
Epoch 5/32
17/17 [==============================] - 675s 40s/step - loss: 1682.1094 - ms
e: 1682.1095 - val_loss: 4349.6538 - val_mse: 4349.6538
Epoch 6/32
17/17 [==============================] - 665s 39s/step - loss: 1432.1898 - ms
e: 1432.1898 - val_loss: 4366.6562 - val_mse: 4366.6562
Epoch 7/32
17/17 [==============================] - 665s 39s/step - loss: 1301.9513 - ms
e: 1301.9513 - val_loss: 4368.7378 - val_mse: 4368.7378
Epoch 8/32
17/17 [==============================] - 662s 39s/step - loss: 1055.4714 - ms
e: 1055.4714 - val_loss: 3375.7917 - val_mse: 3375.7917
Epoch 9/32
17/17 [==============================] - 665s 39s/step - loss: 906.2627 - ms
e: 906.2627 - val_loss: 3313.1272 - val_mse: 3313.1272
Epoch 10/32
17/17 [==============================] - 669s 39s/step - loss: 731.3782 - ms
e: 731.3782 - val_loss: 3950.6821 - val_mse: 3950.6821
Epoch 11/32
17/17 [==============================] - 669s 39s/step - loss: 641.6349 - ms
e: 641.6349 - val_loss: 3301.6663 - val_mse: 3301.6663
Epoch 12/32
17/17 [==============================] - 665s 39s/step - loss: 543.0818 - ms
e: 543.0818 - val_loss: 4225.6763 - val_mse: 4225.6763
Epoch 13/32
17/17 [==============================] - 668s 39s/step - loss: 455.7968 - ms
e: 455.7968 - val_loss: 3543.3291 - val_mse: 3543.3291
Epoch 14/32
17/17 [==============================] - 670s 39s/step - loss: 376.9647 - ms
e: 376.9647 - val_loss: 4170.3584 - val_mse: 4170.3589
Epoch 15/32
17/17 [==============================] - 667s 39s/step - loss: 355.3262 - ms
e: 355.3262 - val_loss: 3499.7725 - val_mse: 3499.7725
Epoch 16/32
17/17 [==============================] - 669s 39s/step - loss: 306.6470 - ms
e: 306.6470 - val_loss: 3761.7344 - val_mse: 3761.7344
Epoch 17/32
17/17 [==============================] - 669s 39s/step - loss: 247.9308 - ms
e: 247.9308 - val_loss: 3444.6384 - val_mse: 3444.6384
Epoch 18/32
17/17 [==============================] - 665s 39s/step - loss: 221.5101 - ms
e: 221.5101 - val_loss: 3496.6230 - val_mse: 3496.6228
Epoch 19/32
17/17 [==============================] - 667s 39s/step - loss: 222.0941 - ms
e: 222.0941 - val_loss: 3401.8181 - val_mse: 3401.8181
Epoch 20/32
17/17 [==============================] - 672s 40s/step - loss: 217.4072 - ms
e: 217.4072 - val_loss: 3100.0210 - val_mse: 3100.0210
Epoch 21/32
17/17 [==============================] - 674s 40s/step - loss: 198.0668 - ms
e: 198.0668 - val_loss: 3122.1951 - val_mse: 3122.1951
Epoch 22/32
17/17 [==============================] - 679s 40s/step - loss: 169.3572 - ms
e: 169.3572 - val_loss: 3262.3933 - val_mse: 3262.3933
Epoch 23/32
17/17 [==============================] - 675s 40s/step - loss: 168.7509 - ms
e: 168.7509 - val_loss: 3138.6255 - val_mse: 3138.6255
```

```
Epoch 24/32
17/17 [==============================] - 675s 40s/step - loss: 142.4550 - ms
e: 142.4550 - val_loss: 3043.8423 - val_mse: 3043.8418
Epoch 25/32
17/17 [==============================] - 673s 40s/step - loss: 141.9805 - ms
e: 141.9805 - val_loss: 2916.8665 - val_mse: 2916.8665
Epoch 26/32
17/17 [==============================] - 669s 39s/step - loss: 125.4312 - ms
e: 125.4312 - val_loss: 2831.0752 - val_mse: 2831.0752
Epoch 27/32
17/17 [==============================] - 672s 40s/step - loss: 121.0943 - ms
e: 121.0943 - val_loss: 2719.6580 - val_mse: 2719.6580
Epoch 28/32
17/17 [==============================] - 673s 40s/step - loss: 115.6269 - ms
e: 115.6269 - val_loss: 2692.2061 - val_mse: 2692.2061
Epoch 29/32
17/17 [==============================] - 667s 39s/step - loss: 113.0639 - ms
e: 113.0639 - val_loss: 2631.1128 - val_mse: 2631.1130
Epoch 30/32
17/17 [==============================] - 667s 39s/step - loss: 105.9257 - ms
e: 105.9257 - val_loss: 2482.4878 - val_mse: 2482.4878
Epoch 31/32
17/17 [==============================] - 665s 39s/step - loss: 107.3539 - ms
e: 107.3539 - val_loss: 2501.4958 - val_mse: 2501.4958
Epoch 32/32
17/17 [==============================] - 663s 39s/step - loss: 98.5172 - mse:
98.5172 - val_loss: 2524.4724 - val_mse: 2524.4724
```

In [42]:
```
results_train = model.evaluate(X_test, y_test)
```

```
33/33 [==============================] - 16s 494ms/step - loss: 2746.8408 - ms
e: 2746.8408
```

In [50]:
```
model.save()
```

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_28 (Conv2D)           (None, 224, 224, 32)      896       
_________________________________________________________________
batch_normalization_28 (Batc (None, 224, 224, 32)      128       
_________________________________________________________________
conv2d_29 (Conv2D)           (None, 224, 224, 32)      9248      
_________________________________________________________________
batch_normalization_29 (Batc (None, 224, 224, 32)      128       
_________________________________________________________________
max_pooling2d_12 (MaxPooling (None, 112, 112, 32)      0         
_________________________________________________________________
conv2d_30 (Conv2D)           (None, 112, 112, 64)      18496     
_________________________________________________________________
batch_normalization_30 (Batc (None, 112, 112, 64)      256       
_________________________________________________________________
conv2d_31 (Conv2D)           (None, 112, 112, 64)      36928     
_________________________________________________________________
batch_normalization_31 (Batc (None, 112, 112, 64)      256       
_________________________________________________________________
conv2d_32 (Conv2D)           (None, 112, 112, 64)      36928     
_________________________________________________________________
batch_normalization_32 (Batc (None, 112, 112, 64)      256       
_________________________________________________________________
max_pooling2d_13 (MaxPooling (None, 56, 56, 64)        0         
_________________________________________________________________
conv2d_33 (Conv2D)           (None, 56, 56, 128)       73856     
_________________________________________________________________
batch_normalization_33 (Batc (None, 56, 56, 128)       512       
_________________________________________________________________
conv2d_34 (Conv2D)           (None, 56, 56, 128)       147584    
_________________________________________________________________
batch_normalization_34 (Batc (None, 56, 56, 128)       512       
_________________________________________________________________
max_pooling2d_14 (MaxPooling (None, 28, 28, 128)       0         
_________________________________________________________________
flatten_3 (Flatten)          (None, 100352)            0         
_________________________________________________________________
dense_8 (Dense)              (None, 512)               51380736  
_________________________________________________________________
dropout_2 (Dropout)          (None, 512)               0         
_________________________________________________________________
dense_9 (Dense)              (None, 256)               131328    
_________________________________________________________________
dense_10 (Dense)             (None, 128)               32896     
_________________________________________________________________
dense_11 (Dense)             (None, 1)                 129       
=================================================================
Total params: 51,871,073
Trainable params: 51,870,049
Non-trainable params: 1,024
_________________________________________________________________
```

In [51]:
```
model.save('my_model_batch500.h5')
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:
```
df_scrub.to_csv('data/clean_dataframe.csv')
```