Open in Colab

In [1]:
```
from google.colab import drive

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

```
Mounted at /content/gdrive
```

In [2]:
```
%cd /content/gdrive/MyDrive
```

```
/content/gdrive/MyDrive
```

Facial Keypoints Detection

Detect the location of keypoints on face images

Data:

There are 15 keypoints that specifies the position of facial features by an (x,y) real-valued pair. These 15 pairs comprise of 3 for each eye (center, inner corner and outer corner positions), 2 for each eyebrow (inner and out end positions), 1 for the nose and 4 (left corner, right corner, top and bottom positions) for the lip.

To predict: Keypoint Locations of the 1783 test images.

Application for such a model: Applying filters on Instagram, buying eyeglasses online

Exploratory Data Analysis

In [109…
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [110…
```
df_train = pd.read_csv('training.csv')
df_train.shape
```

Out[110]:
```
(7049, 31)
```

In [111…
```
df_train.columns
```

Out[111]:
```
Index(['left_eye_center_x', 'left_eye_center_y', 'right_eye_center_x',
       'right_eye_center_y', 'left_eye_inner_corner_x',
       'left_eye_inner_corner_y', 'left_eye_outer_corner_x',
       'left_eye_outer_corner_y', 'right_eye_inner_corner_x',
       'right_eye_inner_corner_y', 'right_eye_outer_corner_x',
       'right_eye_outer_corner_y', 'left_eyebrow_inner_end_x',
       'left_eyebrow_inner_end_y', 'left_eyebrow_outer_end_x',
       'left_eyebrow_outer_end_y', 'right_eyebrow_inner_end_x',
       'right_eyebrow_inner_end_y', 'right_eyebrow_outer_end_x',
       'right_eyebrow_outer_end_y', 'nose_tip_x', 'nose_tip_y',
       'mouth_left_corner_x', 'mouth_left_corner_y', 'mouth_right_corner_x',
       'mouth_right_corner_y', 'mouth_center_top_lip_x',
       'mouth_center_top_lip_y', 'mouth_center_bottom_lip_x',
       'mouth_center_bottom_lip_y', 'Image'],
      dtype='object')
```

In [112…
```
df_train.describe()
```

Out[112]:

	left_eye_center_x	left_eye_center_y	right_eye_center_x	right_eye_center_y	left_eye_
count	7039.000000	7039.000000	7036.000000	7036.000000	
mean	66.359021	37.651234	30.306102	37.976943	
std	3.448233	3.152926	3.083230	3.033621	
min	22.763345	1.616512	0.686592	4.091264	
25%	65.082895	35.900451	28.783339	36.327681	
50%	66.497566	37.528055	30.251378	37.813273	
75%	68.024752	39.258449	31.768334	39.566729	
max	94.689280	80.502649	85.039381	81.270911	

In [113…
```
df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7049 entries, 0 to 7048
Data columns (total 31 columns):
 #   Column                     Non-Null Count  Dtype  
---  ------                     --------------  -----  
 0   left_eye_center_x          7039 non-null   float64
 1   left_eye_center_y          7039 non-null   float64
 2   right_eye_center_x         7036 non-null   float64
 3   right_eye_center_y         7036 non-null   float64
 4   left_eye_inner_corner_x    2271 non-null   float64
 5   left_eye_inner_corner_y    2271 non-null   float64
 6   left_eye_outer_corner_x    2267 non-null   float64
 7   left_eye_outer_corner_y    2267 non-null   float64
 8   right_eye_inner_corner_x   2268 non-null   float64
 9   right_eye_inner_corner_y   2268 non-null   float64
 10  right_eye_outer_corner_x   2268 non-null   float64
 11  right_eye_outer_corner_y   2268 non-null   float64
 12  left_eyebrow_inner_end_x   2270 non-null   float64
 13  left_eyebrow_inner_end_y   2270 non-null   float64
 14  left_eyebrow_outer_end_x   2225 non-null   float64
 15  left_eyebrow_outer_end_y   2225 non-null   float64
 16  right_eyebrow_inner_end_x  2270 non-null   float64
 17  right_eyebrow_inner_end_y  2270 non-null   float64
 18  right_eyebrow_outer_end_x  2236 non-null   float64
 19  right_eyebrow_outer_end_y  2236 non-null   float64
 20  nose_tip_x                 7049 non-null   float64
 21  nose_tip_y                 7049 non-null   float64
 22  mouth_left_corner_x        2269 non-null   float64
 23  mouth_left_corner_y        2269 non-null   float64
 24  mouth_right_corner_x       2270 non-null   float64
 25  mouth_right_corner_y       2270 non-null   float64
 26  mouth_center_top_lip_x     2275 non-null   float64
 27  mouth_center_top_lip_y     2275 non-null   float64
 28  mouth_center_bottom_lip_x  7016 non-null   float64
 29  mouth_center_bottom_lip_y  7016 non-null   float64
 30  Image                      7049 non-null   object 
dtypes: float64(30), object(1)
memory usage: 1.7+ MB
```

In [114…

```
'''
The Image column is a string, need to convert that to numpy araay.
As the pixel values range from 0 to 256, apart from 0 the range is 255.
So dividing all the values by 255 will convert it to range from 0 to 1.
'''
def convert_data_to_image(image_data):
    images = []
    for _, sample in image_data.iterrows():
        image = np.array(sample["Image"].split(' '), dtype=int)
        image = np.reshape(image, (96,96,1))
        images.append(image)
    images = np.array(images)/255
    return images
```

In [115…

```
#Getting keypoint features from the training dataframe
def get_keypoints_features(keypoint_data):
    keypoint_data = keypoint_data.drop("Image", axis=1)
    keypoint_features = []
    for _, sample_keypoints in keypoint_data.iterrows():
        keypoint_features.append(sample_keypoints)
```

```
    keypoint_features = np.array(keypoint_features, dtype="float")
    return keypoint_features
```

In [116…

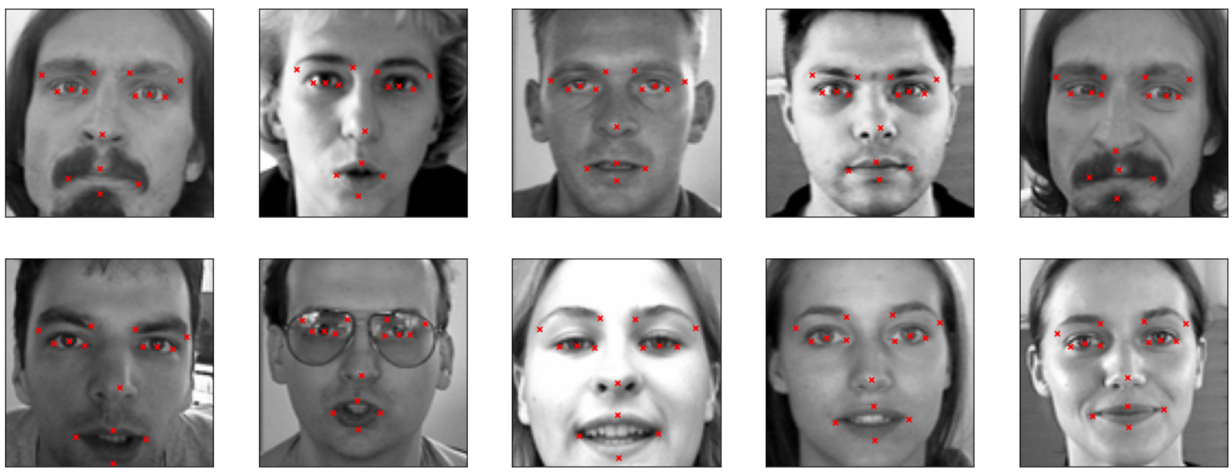
```
#Method to plot the images
def plot_sample(image, keypoint, axis, title):
    image = image.reshape(96,96)
    axis.imshow(image, cmap="gray")
    axis.scatter(keypoint[::2], keypoint[1::2], marker='x', color = 'red',s=20)
    plt.title(title)
```

In [117…

```
train_images = convert_data_to_image(df_train)
train_keypoints = get_keypoints_features(df_train)
```

In [118…

```
'''
Let's take a look at the first 10 images in the training dataframe.
We mark the keypoints on each face in red.
'''
fig = plt.figure(figsize=(20,16))
for i in range(10):
    axis = fig.add_subplot(4, 5, i+1, xticks=[], yticks=[])
    plot_sample(train_images[i], train_keypoints[i], axis, "")
plt.show()
```



The training data has a lot of missing keypoint features. Let's take a look at the number of missing rows for each column.

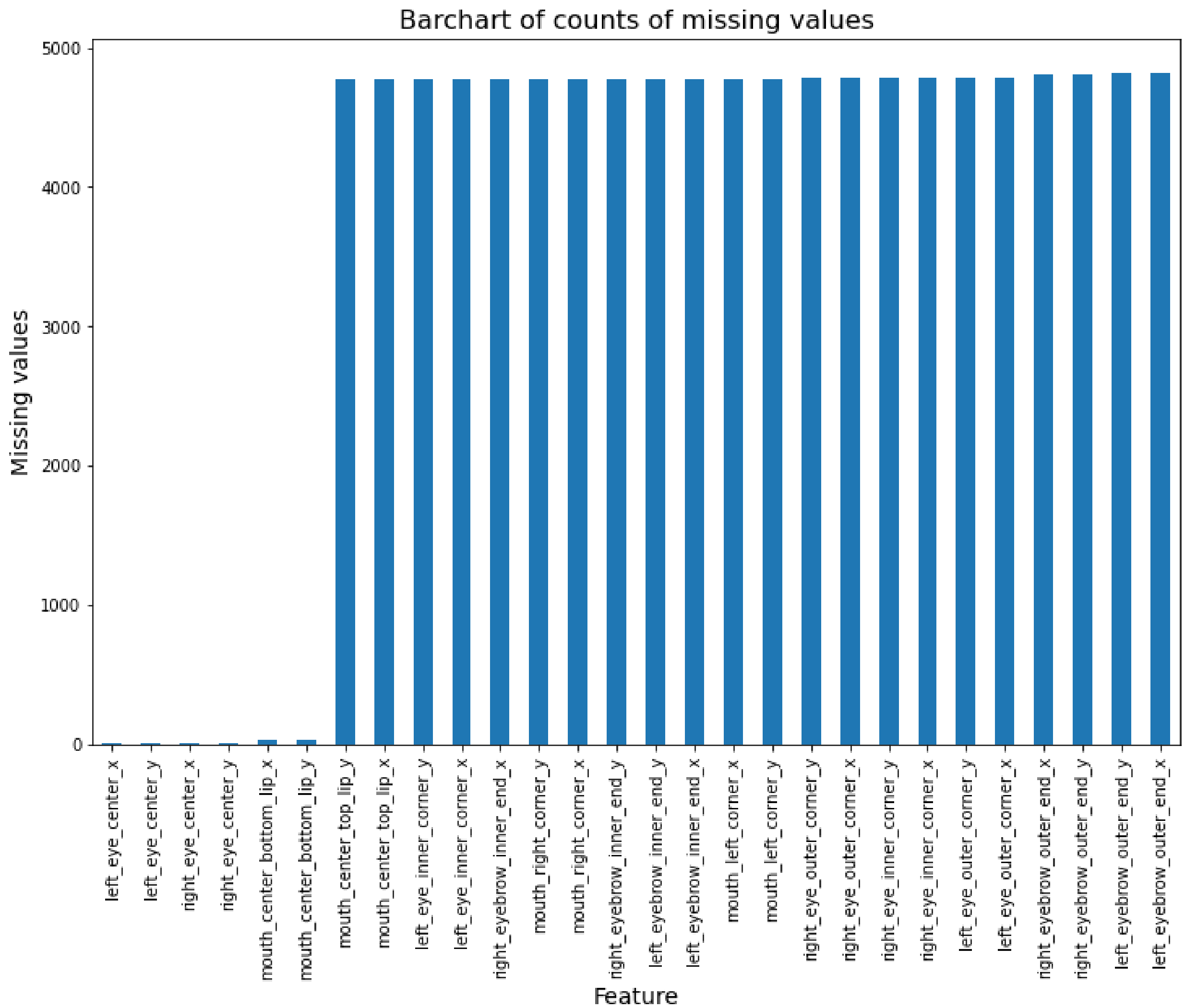
In [119…

```
#Check for empty entries
df_train.isnull().sum()
```

Out[119]:
```
left_eye_center_x              10
left_eye_center_y              10
right_eye_center_x             13
right_eye_center_y             13
left_eye_inner_corner_x      4778
left_eye_inner_corner_y      4778
left_eye_outer_corner_x      4782
left_eye_outer_corner_y      4782
right_eye_inner_corner_x     4781
right_eye_inner_corner_y     4781
right_eye_outer_corner_x     4781
right_eye_outer_corner_y     4781
left_eyebrow_inner_end_x     4779
left_eyebrow_inner_end_y     4779
left_eyebrow_outer_end_x     4824
left_eyebrow_outer_end_y     4824
right_eyebrow_inner_end_x    4779
right_eyebrow_inner_end_y    4779
right_eyebrow_outer_end_x    4813
right_eyebrow_outer_end_y    4813
nose_tip_x                      0
nose_tip_y                      0
mouth_left_corner_x          4780
mouth_left_corner_y          4780
mouth_right_corner_x         4779
mouth_right_corner_y         4779
mouth_center_top_lip_x       4774
mouth_center_top_lip_y       4774
mouth_center_bottom_lip_x      33
mouth_center_bottom_lip_y      33
Image                           0
dtype: int64
```

A bar plot shows it better ..

In [120…
```
missing_cols = df_train.isnull().sum()
missing_cols
plot_cols = missing_cols[missing_cols>0] # drop 0 count cols
plot_cols.sort_values(inplace=True)
plot_cols.plot.bar(figsize=(12,8))
plt.xlabel("Feature",fontsize=14)
plt.ylabel("Missing values",fontsize=14)
plt.title("Barchart of counts of missing values",fontsize=16)
plt.show()
```



The number of columns that are mising is < 5000 which is a significant amount of the data. We will talk about how to deal with these missing data in a forthcoming section.

Training of the Base Model

The rows that have missing data in the training set are classified as low-resolution images. On the contrary the images that have no missing keypoints are of high resolution. In this first model that we train, we will not separate the high resolution images from the low resolution ones in order to have a starting baseline. This base model will allow us to assess the approaches for filling missing values using a common model.

In [121…
```
# However to ensure we get a valid accuracy with the CNN model later, the missi
df_train_zero = df_train.replace(np.nan,0)
```

In [122…
```
df_train_zero['Image'] = df_train_zero['Image'].apply(lambda x: np.fromstring(x
```

In [123…
```
'''
This method groups the keypoints in a seperate dataframe called features and in
'''
def process_df(df):
  img = [im for im in df['Image']]
```

```
  img = np.array(img,dtype = 'float')
  print(img.shape)
  img = np.asarray(img, dtype=np.uint8).reshape(df.shape[0],96,96,1)
  print(img.shape)

  feature = df.drop('Image',axis = 1)
  y = []
  for i in range(df.shape[0]):
    points = feature.iloc[i,:]
    y.append(points)
  y = np.array(y,dtype = 'float')

  return img,y
```

In [124…

```
X, y = process_df(df_train_zero)
```

```
(7049, 96, 96)
(7049, 96, 96, 1)
```

In [125…

```
#Splitting 20% of the training data to dev set
from sklearn.model_selection import train_test_split
X_train, X_dev, y_train, y_dev = train_test_split(X, y, test_size=0.2, random_s
```

In [126…

```
from keras.models import Sequential
from keras.layers import Dense, Conv2D, Flatten, MaxPooling2D,Dropout
from keras.metrics import RootMeanSquaredError
from keras.layers.advanced_activations import LeakyReLU
from keras import metrics
```

In [132…

```
model = Sequential()

model.add(Conv2D(32, (3,3), padding='same', use_bias=False, input_shape=(96,96,
model.add(LeakyReLU(alpha = 0.1))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(128, (3,3), padding='same', use_bias=False))

model.add(Flatten())
model.add(Dropout(0.3))
model.add(Dense(30,activation='elu'))
```

In [133…

```
model.summary()
```

```
Model: "sequential_8"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_48 (Conv2D)           (None, 96, 96, 32)        288       
_________________________________________________________________
leaky_re_lu_40 (LeakyReLU)   (None, 96, 96, 32)        0         
_________________________________________________________________
conv2d_49 (Conv2D)           (None, 96, 96, 64)        18432     
_________________________________________________________________
leaky_re_lu_41 (LeakyReLU)   (None, 96, 96, 64)        0         
_________________________________________________________________
conv2d_50 (Conv2D)           (None, 96, 96, 64)        36864     
_________________________________________________________________
leaky_re_lu_42 (LeakyReLU)   (None, 96, 96, 64)        0         
_________________________________________________________________
max_pooling2d_24 (MaxPooling (None, 48, 48, 64)        0         
_________________________________________________________________
conv2d_51 (Conv2D)           (None, 48, 48, 64)        36864     
_________________________________________________________________
leaky_re_lu_43 (LeakyReLU)   (None, 48, 48, 64)        0         
_________________________________________________________________
max_pooling2d_25 (MaxPooling (None, 24, 24, 64)        0         
_________________________________________________________________
conv2d_52 (Conv2D)           (None, 24, 24, 64)        36864     
_________________________________________________________________
leaky_re_lu_44 (LeakyReLU)   (None, 24, 24, 64)        0         
_________________________________________________________________
max_pooling2d_26 (MaxPooling (None, 12, 12, 64)        0         
_________________________________________________________________
conv2d_53 (Conv2D)           (None, 12, 12, 128)       73728     
_________________________________________________________________
flatten_8 (Flatten)          (None, 18432)             0         
_________________________________________________________________
dropout_8 (Dropout)          (None, 18432)             0         
_________________________________________________________________
dense_8 (Dense)              (None, 30)                552990    
=================================================================
Total params: 756,030
Trainable params: 756,030
Non-trainable params: 0
_________________________________________________________________
```

In [134…

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

In [135…

```
history=model.fit(X_train,y_train,epochs = 100,batch_size = 256,validation_data
```

```
Epoch 1/100
23/23 [==============================] - 5s 167ms/step - loss: 1622.7681 - acc
uracy: 0.4564 - val_loss: 588.5978 - val_accuracy: 0.8702
Epoch 2/100
23/23 [==============================] - 3s 149ms/step - loss: 495.9973 - accu
racy: 0.8821 - val_loss: 447.8537 - val_accuracy: 0.8702
Epoch 3/100
23/23 [==============================] - 3s 148ms/step - loss: 418.5301 - accu
racy: 0.8944 - val_loss: 423.7028 - val_accuracy: 0.8702
Epoch 4/100
23/23 [==============================] - 3s 148ms/step - loss: 392.4079 - accu
racy: 0.9010 - val_loss: 379.7982 - val_accuracy: 0.8369
Epoch 5/100
23/23 [==============================] - 3s 149ms/step - loss: 380.8558 - accu
racy: 0.8836 - val_loss: 357.5596 - val_accuracy: 0.8702
Epoch 6/100
23/23 [==============================] - 3s 149ms/step - loss: 341.0533 - accu
racy: 0.8911 - val_loss: 347.0110 - val_accuracy: 0.8248
Epoch 7/100
23/23 [==============================] - 3s 148ms/step - loss: 322.8138 - accu
racy: 0.8738 - val_loss: 303.9173 - val_accuracy: 0.8220
Epoch 8/100
23/23 [==============================] - 3s 149ms/step - loss: 290.0356 - accu
racy: 0.8727 - val_loss: 301.9377 - val_accuracy: 0.8681
Epoch 9/100
23/23 [==============================] - 3s 149ms/step - loss: 279.5725 - accu
racy: 0.8712 - val_loss: 258.9967 - val_accuracy: 0.7957
Epoch 10/100
23/23 [==============================] - 3s 148ms/step - loss: 260.2481 - accu
racy: 0.8380 - val_loss: 257.1318 - val_accuracy: 0.8305
Epoch 11/100
23/23 [==============================] - 3s 148ms/step - loss: 249.3573 - accu
racy: 0.8603 - val_loss: 294.0212 - val_accuracy: 0.8603
Epoch 12/100
23/23 [==============================] - 3s 149ms/step - loss: 245.2263 - accu
racy: 0.8710 - val_loss: 242.4667 - val_accuracy: 0.7929
Epoch 13/100
23/23 [==============================] - 3s 148ms/step - loss: 232.2357 - accu
racy: 0.8574 - val_loss: 348.4321 - val_accuracy: 0.8695
Epoch 14/100
23/23 [==============================] - 3s 149ms/step - loss: 299.8841 - accu
racy: 0.8822 - val_loss: 329.0486 - val_accuracy: 0.8333
Epoch 15/100
23/23 [==============================] - 3s 149ms/step - loss: 240.4166 - accu
racy: 0.8668 - val_loss: 247.2322 - val_accuracy: 0.8035
Epoch 16/100
23/23 [==============================] - 3s 149ms/step - loss: 207.1120 - accu
racy: 0.8462 - val_loss: 206.4811 - val_accuracy: 0.8163
Epoch 17/100
23/23 [==============================] - 3s 149ms/step - loss: 189.3205 - accu
racy: 0.8401 - val_loss: 242.2174 - val_accuracy: 0.8674
Epoch 18/100
23/23 [==============================] - 3s 149ms/step - loss: 194.1059 - accu
racy: 0.8616 - val_loss: 250.1876 - val_accuracy: 0.7610
Epoch 19/100
23/23 [==============================] - 3s 148ms/step - loss: 198.2754 - accu
racy: 0.8304 - val_loss: 564.5015 - val_accuracy: 0.1057
Epoch 20/100
23/23 [==============================] - 3s 148ms/step - loss: 262.3670 - accu
racy: 0.7585 - val_loss: 213.3157 - val_accuracy: 0.8043
```

```
Epoch 21/100
23/23 [==============================] - 3s 148ms/step - loss: 202.7152 - accu
racy: 0.8461 - val_loss: 297.9091 - val_accuracy: 0.8702
Epoch 22/100
23/23 [==============================] - 3s 147ms/step - loss: 252.6471 - accu
racy: 0.8674 - val_loss: 215.2886 - val_accuracy: 0.7858
Epoch 23/100
23/23 [==============================] - 3s 149ms/step - loss: 190.5680 - accu
racy: 0.8238 - val_loss: 227.0015 - val_accuracy: 0.8567
Epoch 24/100
23/23 [==============================] - 3s 149ms/step - loss: 173.4198 - accu
racy: 0.8504 - val_loss: 196.2047 - val_accuracy: 0.8305
Epoch 25/100
23/23 [==============================] - 3s 148ms/step - loss: 153.3982 - accu
racy: 0.8359 - val_loss: 180.4961 - val_accuracy: 0.7929
Epoch 26/100
23/23 [==============================] - 3s 148ms/step - loss: 140.5612 - accu
racy: 0.8281 - val_loss: 178.3168 - val_accuracy: 0.7894
Epoch 27/100
23/23 [==============================] - 3s 149ms/step - loss: 132.1458 - accu
racy: 0.8228 - val_loss: 201.9178 - val_accuracy: 0.8617
Epoch 28/100
23/23 [==============================] - 3s 149ms/step - loss: 135.1636 - accu
racy: 0.8449 - val_loss: 177.7684 - val_accuracy: 0.8496
Epoch 29/100
23/23 [==============================] - 3s 148ms/step - loss: 119.8947 - accu
racy: 0.8489 - val_loss: 187.7137 - val_accuracy: 0.8092
Epoch 30/100
23/23 [==============================] - 3s 149ms/step - loss: 115.4245 - accu
racy: 0.8290 - val_loss: 175.5536 - val_accuracy: 0.7837
Epoch 31/100
23/23 [==============================] - 3s 149ms/step - loss: 116.5702 - accu
racy: 0.8186 - val_loss: 165.7075 - val_accuracy: 0.8021
Epoch 32/100
23/23 [==============================] - 3s 149ms/step - loss: 117.5300 - accu
racy: 0.8370 - val_loss: 288.1404 - val_accuracy: 0.7440
Epoch 33/100
23/23 [==============================] - 3s 148ms/step - loss: 153.3754 - accu
racy: 0.8249 - val_loss: 180.5504 - val_accuracy: 0.7943
Epoch 34/100
23/23 [==============================] - 3s 148ms/step - loss: 112.9783 - accu
racy: 0.8300 - val_loss: 165.8312 - val_accuracy: 0.8057
Epoch 35/100
23/23 [==============================] - 3s 150ms/step - loss: 91.5434 - accur
acy: 0.8320 - val_loss: 160.2437 - val_accuracy: 0.8255
Epoch 36/100
23/23 [==============================] - 3s 149ms/step - loss: 97.9697 - accur
acy: 0.8305 - val_loss: 173.6196 - val_accuracy: 0.7482
Epoch 37/100
23/23 [==============================] - 3s 147ms/step - loss: 101.9281 - accu
racy: 0.8068 - val_loss: 634.3821 - val_accuracy: 0.6851
Epoch 38/100
23/23 [==============================] - 3s 149ms/step - loss: 275.9308 - accu
racy: 0.8362 - val_loss: 381.9120 - val_accuracy: 0.6816
Epoch 39/100
23/23 [==============================] - 3s 148ms/step - loss: 209.7385 - accu
racy: 0.8287 - val_loss: 183.4550 - val_accuracy: 0.7610
Epoch 40/100
23/23 [==============================] - 3s 149ms/step - loss: 109.9488 - accu
racy: 0.8090 - val_loss: 190.3172 - val_accuracy: 0.8021
```

```
Epoch 41/100
23/23 [==============================] - 3s 148ms/step - loss: 104.6588 - accu
racy: 0.8287 - val_loss: 199.2072 - val_accuracy: 0.8433
Epoch 42/100
23/23 [==============================] - 3s 148ms/step - loss: 94.1169 - accur
acy: 0.8338 - val_loss: 164.5707 - val_accuracy: 0.8270
Epoch 43/100
23/23 [==============================] - 3s 148ms/step - loss: 73.2415 - accur
acy: 0.8299 - val_loss: 163.5483 - val_accuracy: 0.8014
Epoch 44/100
23/23 [==============================] - 3s 149ms/step - loss: 71.3638 - accur
acy: 0.8147 - val_loss: 180.1313 - val_accuracy: 0.8504
Epoch 45/100
23/23 [==============================] - 3s 149ms/step - loss: 72.3179 - accur
acy: 0.8414 - val_loss: 162.6873 - val_accuracy: 0.7879
Epoch 46/100
23/23 [==============================] - 3s 149ms/step - loss: 63.3692 - accur
acy: 0.8247 - val_loss: 181.1029 - val_accuracy: 0.8496
Epoch 47/100
23/23 [==============================] - 3s 149ms/step - loss: 62.1496 - accur
acy: 0.8466 - val_loss: 164.2500 - val_accuracy: 0.8277
Epoch 48/100
23/23 [==============================] - 3s 149ms/step - loss: 53.7545 - accur
acy: 0.8317 - val_loss: 162.3843 - val_accuracy: 0.7745
Epoch 49/100
23/23 [==============================] - 3s 149ms/step - loss: 50.2406 - accur
acy: 0.8159 - val_loss: 173.0798 - val_accuracy: 0.7468
Epoch 50/100
23/23 [==============================] - 3s 149ms/step - loss: 60.3194 - accur
acy: 0.8010 - val_loss: 161.4770 - val_accuracy: 0.8184
Epoch 51/100
23/23 [==============================] - 3s 149ms/step - loss: 47.0844 - accur
acy: 0.8306 - val_loss: 161.9439 - val_accuracy: 0.8043
Epoch 52/100
23/23 [==============================] - 3s 148ms/step - loss: 42.2752 - accur
acy: 0.8249 - val_loss: 168.1638 - val_accuracy: 0.8227
Epoch 53/100
23/23 [==============================] - 3s 149ms/step - loss: 40.5424 - accur
acy: 0.8407 - val_loss: 162.7090 - val_accuracy: 0.7908
Epoch 54/100
23/23 [==============================] - 4s 153ms/step - loss: 42.5721 - accur
acy: 0.8276 - val_loss: 199.5226 - val_accuracy: 0.8468
Epoch 55/100
23/23 [==============================] - 3s 148ms/step - loss: 65.3380 - accur
acy: 0.8376 - val_loss: 167.3572 - val_accuracy: 0.7844
Epoch 56/100
23/23 [==============================] - 3s 149ms/step - loss: 64.2700 - accur
acy: 0.8086 - val_loss: 173.0815 - val_accuracy: 0.8227
Epoch 57/100
23/23 [==============================] - 3s 148ms/step - loss: 47.6573 - accur
acy: 0.8377 - val_loss: 185.1741 - val_accuracy: 0.8255
Epoch 58/100
23/23 [==============================] - 3s 149ms/step - loss: 46.3510 - accur
acy: 0.8343 - val_loss: 185.8278 - val_accuracy: 0.8156
Epoch 59/100
23/23 [==============================] - 3s 148ms/step - loss: 56.7344 - accur
acy: 0.8353 - val_loss: 184.6636 - val_accuracy: 0.8525
Epoch 60/100
23/23 [==============================] - 3s 148ms/step - loss: 47.7540 - accur
acy: 0.8463 - val_loss: 161.6840 - val_accuracy: 0.8085
```

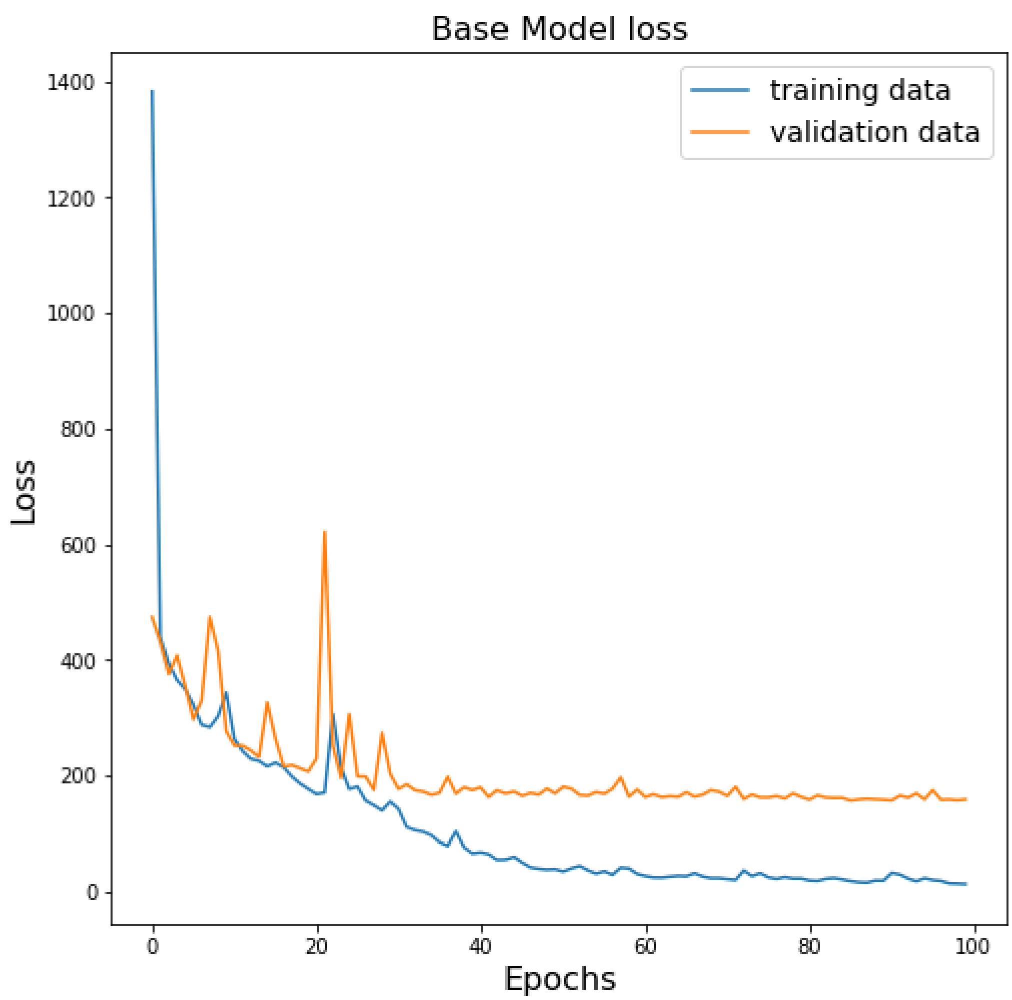
```
Epoch 61/100
23/23 [==============================] - 3s 149ms/step - loss: 32.3615 - accur
acy: 0.8267 - val_loss: 163.2087 - val_accuracy: 0.7894
Epoch 62/100
23/23 [==============================] - 3s 148ms/step - loss: 34.2095 - accur
acy: 0.8269 - val_loss: 167.8142 - val_accuracy: 0.8085
Epoch 63/100
23/23 [==============================] - 3s 148ms/step - loss: 35.1883 - accur
acy: 0.8444 - val_loss: 167.6818 - val_accuracy: 0.7809
Epoch 64/100
23/23 [==============================] - 3s 148ms/step - loss: 37.9827 - accur
acy: 0.8216 - val_loss: 160.6193 - val_accuracy: 0.7915
Epoch 65/100
23/23 [==============================] - 3s 149ms/step - loss: 30.4878 - accur
acy: 0.8297 - val_loss: 161.3469 - val_accuracy: 0.7986
Epoch 66/100
23/23 [==============================] - 3s 149ms/step - loss: 27.1698 - accur
acy: 0.8393 - val_loss: 166.3035 - val_accuracy: 0.8284
Epoch 67/100
23/23 [==============================] - 3s 148ms/step - loss: 27.2446 - accur
acy: 0.8314 - val_loss: 168.0039 - val_accuracy: 0.8121
Epoch 68/100
23/23 [==============================] - 3s 149ms/step - loss: 29.0655 - accur
acy: 0.8399 - val_loss: 163.0148 - val_accuracy: 0.8248
Epoch 69/100
23/23 [==============================] - 3s 148ms/step - loss: 23.6718 - accur
acy: 0.8573 - val_loss: 160.8478 - val_accuracy: 0.8021
Epoch 70/100
23/23 [==============================] - 3s 148ms/step - loss: 22.2448 - accur
acy: 0.8350 - val_loss: 164.4715 - val_accuracy: 0.8149
Epoch 71/100
23/23 [==============================] - 3s 149ms/step - loss: 25.2220 - accur
acy: 0.8331 - val_loss: 160.2004 - val_accuracy: 0.8099
Epoch 72/100
23/23 [==============================] - 3s 149ms/step - loss: 23.7649 - accur
acy: 0.8529 - val_loss: 169.7175 - val_accuracy: 0.8369
Epoch 73/100
23/23 [==============================] - 3s 149ms/step - loss: 22.9883 - accur
acy: 0.8579 - val_loss: 168.9706 - val_accuracy: 0.7943
Epoch 74/100
23/23 [==============================] - 3s 149ms/step - loss: 29.3217 - accur
acy: 0.8418 - val_loss: 165.4973 - val_accuracy: 0.8241
Epoch 75/100
23/23 [==============================] - 3s 148ms/step - loss: 22.3362 - accur
acy: 0.8436 - val_loss: 157.6921 - val_accuracy: 0.8000
Epoch 76/100
23/23 [==============================] - 3s 149ms/step - loss: 21.2661 - accur
acy: 0.8399 - val_loss: 163.4530 - val_accuracy: 0.8142
Epoch 77/100
23/23 [==============================] - 3s 148ms/step - loss: 23.1406 - accur
acy: 0.8533 - val_loss: 163.3778 - val_accuracy: 0.8021
Epoch 78/100
23/23 [==============================] - 3s 149ms/step - loss: 24.4433 - accur
acy: 0.8420 - val_loss: 165.5114 - val_accuracy: 0.7596
Epoch 79/100
23/23 [==============================] - 3s 149ms/step - loss: 32.8999 - accur
acy: 0.8258 - val_loss: 176.9936 - val_accuracy: 0.8447
Epoch 80/100
23/23 [==============================] - 3s 149ms/step - loss: 29.1680 - accur
acy: 0.8552 - val_loss: 162.2759 - val_accuracy: 0.8305
```

```
Epoch 81/100
23/23 [==============================] - 3s 149ms/step - loss: 22.7986 - accur
acy: 0.8474 - val_loss: 160.8953 - val_accuracy: 0.8021
Epoch 82/100
23/23 [==============================] - 3s 149ms/step - loss: 23.5321 - accur
acy: 0.8339 - val_loss: 161.8696 - val_accuracy: 0.8291
Epoch 83/100
23/23 [==============================] - 3s 148ms/step - loss: 23.7865 - accur
acy: 0.8494 - val_loss: 166.8689 - val_accuracy: 0.8255
Epoch 84/100
23/23 [==============================] - 3s 148ms/step - loss: 24.1419 - accur
acy: 0.8467 - val_loss: 157.5622 - val_accuracy: 0.7950
Epoch 85/100
23/23 [==============================] - 3s 149ms/step - loss: 23.0491 - accur
acy: 0.8213 - val_loss: 163.6806 - val_accuracy: 0.8206
Epoch 86/100
23/23 [==============================] - 3s 148ms/step - loss: 23.3787 - accur
acy: 0.8510 - val_loss: 160.0679 - val_accuracy: 0.8014
Epoch 87/100
23/23 [==============================] - 3s 149ms/step - loss: 24.6985 - accur
acy: 0.8491 - val_loss: 166.0200 - val_accuracy: 0.8135
Epoch 88/100
23/23 [==============================] - 3s 148ms/step - loss: 21.7969 - accur
acy: 0.8372 - val_loss: 164.0956 - val_accuracy: 0.8064
Epoch 89/100
23/23 [==============================] - 3s 149ms/step - loss: 18.2472 - accur
acy: 0.8423 - val_loss: 174.2126 - val_accuracy: 0.8170
Epoch 90/100
23/23 [==============================] - 3s 148ms/step - loss: 20.7562 - accur
acy: 0.8518 - val_loss: 180.6665 - val_accuracy: 0.8291
Epoch 91/100
23/23 [==============================] - 3s 148ms/step - loss: 23.8081 - accur
acy: 0.8427 - val_loss: 171.5925 - val_accuracy: 0.8355
Epoch 92/100
23/23 [==============================] - 3s 149ms/step - loss: 19.9987 - accur
acy: 0.8569 - val_loss: 162.6451 - val_accuracy: 0.7780
Epoch 93/100
23/23 [==============================] - 3s 148ms/step - loss: 20.5920 - accur
acy: 0.8272 - val_loss: 163.0576 - val_accuracy: 0.8206
Epoch 94/100
23/23 [==============================] - 3s 149ms/step - loss: 20.3519 - accur
acy: 0.8540 - val_loss: 164.7190 - val_accuracy: 0.8128
Epoch 95/100
23/23 [==============================] - 3s 149ms/step - loss: 17.5380 - accur
acy: 0.8498 - val_loss: 159.6213 - val_accuracy: 0.8078
Epoch 96/100
23/23 [==============================] - 3s 148ms/step - loss: 16.3832 - accur
acy: 0.8487 - val_loss: 160.5707 - val_accuracy: 0.8149
Epoch 97/100
23/23 [==============================] - 3s 149ms/step - loss: 15.5963 - accur
acy: 0.8532 - val_loss: 159.5867 - val_accuracy: 0.8043
Epoch 98/100
23/23 [==============================] - 3s 149ms/step - loss: 17.5459 - accur
acy: 0.8642 - val_loss: 168.1236 - val_accuracy: 0.8355
Epoch 99/100
23/23 [==============================] - 3s 149ms/step - loss: 25.1120 - accur
acy: 0.8473 - val_loss: 158.4703 - val_accuracy: 0.7915
Epoch 100/100
23/23 [==============================] - 3s 149ms/step - loss: 18.5275 - accur
acy: 0.8377 - val_loss: 162.5096 - val_accuracy: 0.8121
```

In the logs for training the Base model, 'val_accuracy' refers to the validation set. This is a measure of the accuracy for a set of samples that was not shown to the network during training and hence refers to how much our model works in general for cases outside the training set.

In [36]:
```
#Compare loss function computed as a Root Mean Squared error for both training
plt.figure(figsize=(8,8))
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Base Model loss',fontsize=16)
plt.ylabel('Loss',fontsize=16)
plt.xlabel('Epochs',fontsize=16)
plt.legend(['training data', 'validation data'], loc='upper right',fontsize=14)
```

Out[36]:
```
<matplotlib.legend.Legend at 0x7f220043aed0>
```



From the above two plots, it is clear that the model performs better for the training data as compared to the validation dev data.

We thus take a look at dealing with the missing values with an aim to improve the overall accuracy of our model.

Handling of Missing Values

Here we explore two different approaches to deal with the missing values in the training data.

KNN Imputation of Missing Values:

A popular approach to missing data imputation is to use a model to predict the missing values. The k-nearest neighbor (KNN) algorithm has proven to be generally effective, often referred to as “nearest neighbor imputation.” Here after some trial-and-error we choose k = 10.

In [37]:
```
# Dropping the image axis as that column has a different data structure as well
df_train_knn = df_train.drop('Image',axis = 1)
```

In [38]:
```
from sklearn.impute import KNNImputer
imputer = KNNImputer(n_neighbors=10, weights='uniform', metric='nan_euclidean')
```

In [39]:
```
imputer.fit_transform(df_train_knn)
```

Out[39]:
```
array([[66.03356391, 39.00227368, 30.22700752, ..., 72.93545865,
         43.13070677, 84.48577444],
        [64.33293617, 34.9700766 , 29.9492766 , ..., 70.26655319,
         45.46791489, 85.48017021],
        [65.05705263, 34.90964211, 30.90378947, ..., 70.19178947,
         47.27494737, 78.65936842],
        ...,
        [66.69073171, 36.84522146, 31.66641951, ..., 72.27907654,
         49.46257171, 78.11712   ],
        [70.96508235, 39.85366588, 30.54328471, ..., 76.87111833,
         50.06518588, 79.58644706],
        [66.93831111, 43.42450963, 31.09605926, ..., 78.10264274,
         45.90048   , 82.7730963 ]])
```

In [40]:
```
np_train_knn_trans = imputer.transform(df_train_knn)
```

In [41]:
```
#Change the kNN imputed numpy array to a dataframe and add back teh Image colum
df_train_knn_transform = pd.DataFrame(np_train_knn_trans, columns = df_train_kn
df_train_knn_transform['Image'] = df_train['Image']
df_train_knn_transform['Image'] = df_train_knn_transform['Image'].apply(lambda
```

In [42]:
```
#check for missing values again
df_train_knn_transform.isna().sum()
```

```
Out[42]: left_eye_center_x            0
left_eye_center_y            0
right_eye_center_x           0
right_eye_center_y           0
left_eye_inner_corner_x      0
left_eye_inner_corner_y      0
left_eye_outer_corner_x      0
left_eye_outer_corner_y      0
right_eye_inner_corner_x     0
right_eye_inner_corner_y     0
right_eye_outer_corner_x     0
right_eye_outer_corner_y     0
left_eyebrow_inner_end_x     0
left_eyebrow_inner_end_y     0
left_eyebrow_outer_end_x     0
left_eyebrow_outer_end_y     0
right_eyebrow_inner_end_x    0
right_eyebrow_inner_end_y    0
right_eyebrow_outer_end_x    0
right_eyebrow_outer_end_y    0
nose_tip_x                   0
nose_tip_y                   0
mouth_left_corner_x          0
mouth_left_corner_y          0
mouth_right_corner_x         0
mouth_right_corner_y         0
mouth_center_top_lip_x       0
mouth_center_top_lip_y       0
mouth_center_bottom_lip_x    0
mouth_center_bottom_lip_y    0
Image                        0
dtype: int64
```

In [43]:

```
# Pre-process data
X, y = process_df(df_train_knn_transform)
```

```
(7049, 96, 96)
(7049, 96, 96, 1)
```

In [44]:

```
#Split data into training and dev sets
X_train, X_dev, y_train, y_dev = train_test_split(X, y, test_size=0.2, random_s
```

In [45]:

```
# Apply CNN model to kNN imputed dataset
model = Sequential()

model.add(Conv2D(32, (3,3), padding='same', use_bias=False, input_shape=(96,96,
model.add(LeakyReLU(alpha = 0.1))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
```

```
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(128, (3,3), padding='same', use_bias=False))

model.add(Flatten())
model.add(Dropout(0.2))
model.add(Dense(30,activation='elu'))
```

In [46]:
```
model_knn_impute = model
```

In [47]:
```
model_knn_impute.compile(optimizer='adam',
              loss='mean_squared_error',
              metrics=['accuracy'])
```

In [48]:
```
history_knn_impute=model_knn_impute.fit(X_train,y_train,epochs = 100,batch_size
```

```
Epoch 1/100
23/23 [==============================] - 5s 173ms/step - loss: 2395.9391 - acc
uracy: 0.2060 - val_loss: 216.9492 - val_accuracy: 0.4135
Epoch 2/100
23/23 [==============================] - 3s 149ms/step - loss: 183.9983 - accu
racy: 0.4368 - val_loss: 160.9205 - val_accuracy: 0.4121
Epoch 3/100
23/23 [==============================] - 3s 150ms/step - loss: 130.5181 - accu
racy: 0.4341 - val_loss: 163.5114 - val_accuracy: 0.4298
Epoch 4/100
23/23 [==============================] - 3s 150ms/step - loss: 137.3070 - accu
racy: 0.4561 - val_loss: 179.5486 - val_accuracy: 0.5376
Epoch 5/100
23/23 [==============================] - 3s 150ms/step - loss: 135.0280 - accu
racy: 0.4894 - val_loss: 116.0010 - val_accuracy: 0.5574
Epoch 6/100
23/23 [==============================] - 3s 148ms/step - loss: 112.8174 - accu
racy: 0.5079 - val_loss: 112.5063 - val_accuracy: 0.5589
Epoch 7/100
23/23 [==============================] - 3s 149ms/step - loss: 106.6517 - accu
racy: 0.4864 - val_loss: 114.1932 - val_accuracy: 0.5078
Epoch 8/100
23/23 [==============================] - 3s 149ms/step - loss: 107.9273 - accu
racy: 0.4791 - val_loss: 113.0252 - val_accuracy: 0.5532
Epoch 9/100
23/23 [==============================] - 4s 153ms/step - loss: 104.6479 - accu
racy: 0.5058 - val_loss: 114.1150 - val_accuracy: 0.5773
Epoch 10/100
23/23 [==============================] - 3s 149ms/step - loss: 99.3133 - accur
acy: 0.5141 - val_loss: 158.6082 - val_accuracy: 0.5596
Epoch 11/100
23/23 [==============================] - 3s 148ms/step - loss: 112.4394 - accu
racy: 0.5076 - val_loss: 118.7199 - val_accuracy: 0.4333
Epoch 12/100
23/23 [==============================] - 3s 149ms/step - loss: 109.0480 - accu
racy: 0.4930 - val_loss: 146.9547 - val_accuracy: 0.5865
Epoch 13/100
23/23 [==============================] - 3s 148ms/step - loss: 105.0106 - accu
racy: 0.4978 - val_loss: 104.8330 - val_accuracy: 0.6000
Epoch 14/100
23/23 [==============================] - 3s 148ms/step - loss: 88.2684 - accur
acy: 0.5369 - val_loss: 89.0583 - val_accuracy: 0.5369
Epoch 15/100
23/23 [==============================] - 3s 149ms/step - loss: 86.6006 - accur
acy: 0.5256 - val_loss: 97.1773 - val_accuracy: 0.6000
Epoch 16/100
23/23 [==============================] - 3s 148ms/step - loss: 82.7099 - accur
acy: 0.5434 - val_loss: 127.0155 - val_accuracy: 0.5943
Epoch 17/100
23/23 [==============================] - 3s 148ms/step - loss: 91.2645 - accur
acy: 0.5224 - val_loss: 97.9323 - val_accuracy: 0.5433
Epoch 18/100
23/23 [==============================] - 3s 149ms/step - loss: 79.6923 - accur
acy: 0.5505 - val_loss: 73.7394 - val_accuracy: 0.6142
Epoch 19/100
23/23 [==============================] - 3s 150ms/step - loss: 74.4373 - accur
acy: 0.5426 - val_loss: 87.0116 - val_accuracy: 0.5851
Epoch 20/100
23/23 [==============================] - 3s 148ms/step - loss: 74.9308 - accur
acy: 0.5666 - val_loss: 69.3536 - val_accuracy: 0.6050
```

```
Epoch 21/100
23/23 [==============================] - 3s 149ms/step - loss: 65.5831 - accur
acy: 0.5800 - val_loss: 94.1847 - val_accuracy: 0.6191
Epoch 22/100
23/23 [==============================] - 3s 149ms/step - loss: 74.3802 - accur
acy: 0.5795 - val_loss: 82.7781 - val_accuracy: 0.6418
Epoch 23/100
23/23 [==============================] - 3s 149ms/step - loss: 71.3055 - accur
acy: 0.5444 - val_loss: 64.4762 - val_accuracy: 0.6504
Epoch 24/100
23/23 [==============================] - 3s 149ms/step - loss: 61.4112 - accur
acy: 0.5935 - val_loss: 63.2816 - val_accuracy: 0.6582
Epoch 25/100
23/23 [==============================] - 3s 148ms/step - loss: 59.3713 - accur
acy: 0.6002 - val_loss: 63.5000 - val_accuracy: 0.6504
Epoch 26/100
23/23 [==============================] - 3s 149ms/step - loss: 58.1134 - accur
acy: 0.6027 - val_loss: 67.4386 - val_accuracy: 0.6539
Epoch 27/100
23/23 [==============================] - 4s 153ms/step - loss: 58.9344 - accur
acy: 0.6012 - val_loss: 79.0270 - val_accuracy: 0.6879
Epoch 28/100
23/23 [==============================] - 3s 148ms/step - loss: 63.9231 - accur
acy: 0.6286 - val_loss: 61.5653 - val_accuracy: 0.6539
Epoch 29/100
23/23 [==============================] - 3s 148ms/step - loss: 57.1703 - accur
acy: 0.6127 - val_loss: 58.9768 - val_accuracy: 0.6801
Epoch 30/100
23/23 [==============================] - 3s 148ms/step - loss: 55.2918 - accur
acy: 0.6452 - val_loss: 73.7542 - val_accuracy: 0.6858
Epoch 31/100
23/23 [==============================] - 3s 149ms/step - loss: 58.8290 - accur
acy: 0.6340 - val_loss: 75.9833 - val_accuracy: 0.6816
Epoch 32/100
23/23 [==============================] - 3s 148ms/step - loss: 61.9800 - accur
acy: 0.6420 - val_loss: 76.9183 - val_accuracy: 0.6830
Epoch 33/100
23/23 [==============================] - 3s 148ms/step - loss: 58.3059 - accur
acy: 0.6229 - val_loss: 58.4172 - val_accuracy: 0.6142
Epoch 34/100
23/23 [==============================] - 3s 149ms/step - loss: 51.5642 - accur
acy: 0.6292 - val_loss: 86.6641 - val_accuracy: 0.6617
Epoch 35/100
23/23 [==============================] - 3s 149ms/step - loss: 71.3222 - accur
acy: 0.6102 - val_loss: 65.1720 - val_accuracy: 0.6582
Epoch 36/100
23/23 [==============================] - 3s 149ms/step - loss: 57.5566 - accur
acy: 0.6037 - val_loss: 56.6748 - val_accuracy: 0.6972
Epoch 37/100
23/23 [==============================] - 3s 149ms/step - loss: 49.3588 - accur
acy: 0.6578 - val_loss: 93.5159 - val_accuracy: 0.6000
Epoch 38/100
23/23 [==============================] - 3s 148ms/step - loss: 110.7373 - accu
racy: 0.5264 - val_loss: 104.8854 - val_accuracy: 0.6312
Epoch 39/100
23/23 [==============================] - 3s 153ms/step - loss: 85.6849 - accur
acy: 0.5703 - val_loss: 70.0003 - val_accuracy: 0.6589
Epoch 40/100
23/23 [==============================] - 3s 148ms/step - loss: 64.5495 - accur
acy: 0.6434 - val_loss: 132.0723 - val_accuracy: 0.6638
```

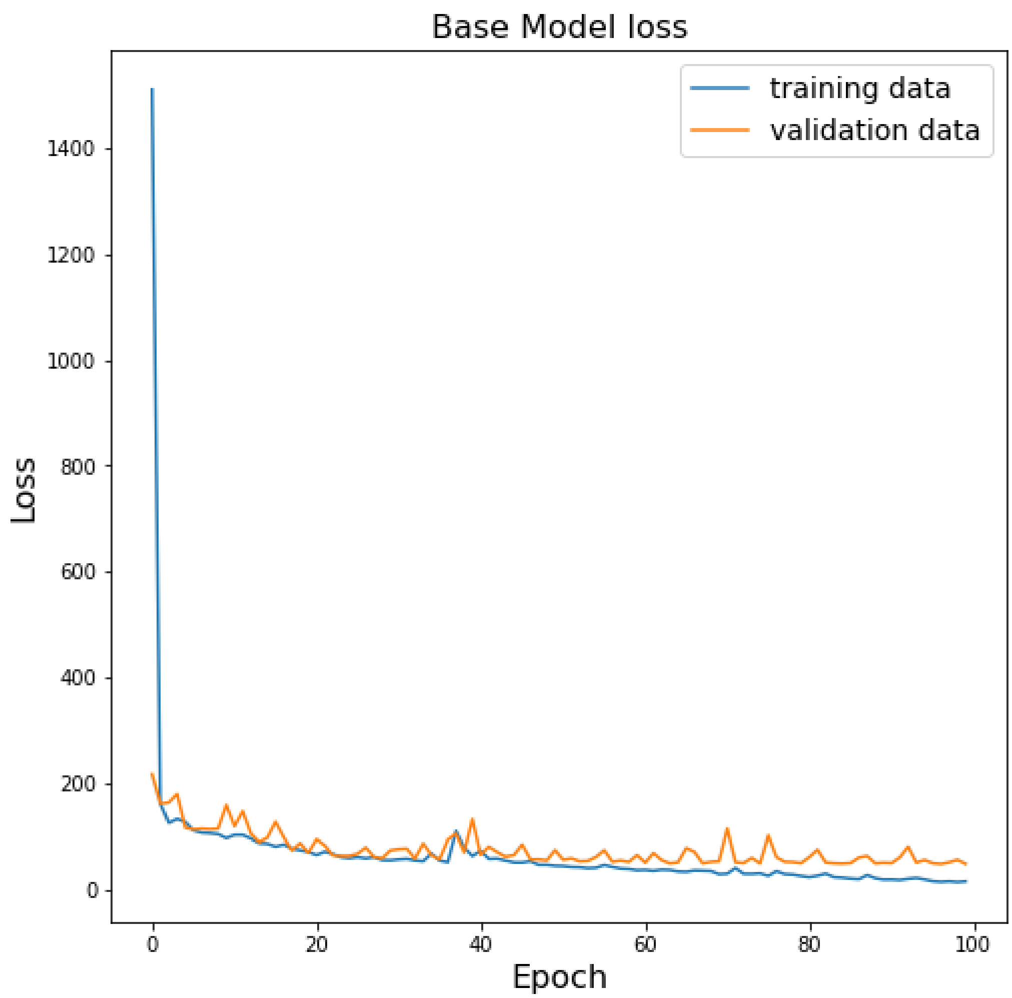
```
Epoch 41/100
23/23 [==============================] - 3s 149ms/step - loss: 83.7274 - accur
acy: 0.6071 - val_loss: 65.4577 - val_accuracy: 0.6184
Epoch 42/100
23/23 [==============================] - 3s 148ms/step - loss: 58.7070 - accur
acy: 0.6208 - val_loss: 80.5502 - val_accuracy: 0.6801
Epoch 43/100
23/23 [==============================] - 3s 148ms/step - loss: 62.3902 - accur
acy: 0.6415 - val_loss: 70.7854 - val_accuracy: 0.6794
Epoch 44/100
23/23 [==============================] - 3s 148ms/step - loss: 56.2390 - accur
acy: 0.6478 - val_loss: 62.2655 - val_accuracy: 0.6915
Epoch 45/100
23/23 [==============================] - 3s 148ms/step - loss: 50.9530 - accur
acy: 0.6522 - val_loss: 65.0191 - val_accuracy: 0.6610
Epoch 46/100
23/23 [==============================] - 3s 149ms/step - loss: 53.2182 - accur
acy: 0.6354 - val_loss: 84.6959 - val_accuracy: 0.7000
Epoch 47/100
23/23 [==============================] - 3s 148ms/step - loss: 60.9291 - accur
acy: 0.6347 - val_loss: 56.7129 - val_accuracy: 0.6965
Epoch 48/100
23/23 [==============================] - 3s 149ms/step - loss: 45.8672 - accur
acy: 0.6524 - val_loss: 56.9951 - val_accuracy: 0.7064
Epoch 49/100
23/23 [==============================] - 3s 148ms/step - loss: 45.9454 - accur
acy: 0.6681 - val_loss: 55.0476 - val_accuracy: 0.7028
Epoch 50/100
23/23 [==============================] - 3s 148ms/step - loss: 43.4515 - accur
acy: 0.6801 - val_loss: 74.2733 - val_accuracy: 0.6957
Epoch 51/100
23/23 [==============================] - 3s 149ms/step - loss: 46.1783 - accur
acy: 0.6578 - val_loss: 56.1101 - val_accuracy: 0.6957
Epoch 52/100
23/23 [==============================] - 3s 150ms/step - loss: 42.8379 - accur
acy: 0.6649 - val_loss: 58.8008 - val_accuracy: 0.6879
Epoch 53/100
23/23 [==============================] - 3s 149ms/step - loss: 43.2435 - accur
acy: 0.6590 - val_loss: 53.3731 - val_accuracy: 0.7057
Epoch 54/100
23/23 [==============================] - 3s 149ms/step - loss: 40.3690 - accur
acy: 0.6637 - val_loss: 54.0430 - val_accuracy: 0.7071
Epoch 55/100
23/23 [==============================] - 3s 149ms/step - loss: 41.8230 - accur
acy: 0.6617 - val_loss: 61.4691 - val_accuracy: 0.6950
Epoch 56/100
23/23 [==============================] - 3s 149ms/step - loss: 44.9147 - accur
acy: 0.6727 - val_loss: 74.0165 - val_accuracy: 0.6986
Epoch 57/100
23/23 [==============================] - 3s 149ms/step - loss: 45.4442 - accur
acy: 0.6677 - val_loss: 52.4223 - val_accuracy: 0.6993
Epoch 58/100
23/23 [==============================] - 3s 148ms/step - loss: 39.3291 - accur
acy: 0.6626 - val_loss: 54.8468 - val_accuracy: 0.7028
Epoch 59/100
23/23 [==============================] - 3s 148ms/step - loss: 38.7089 - accur
acy: 0.6542 - val_loss: 52.1265 - val_accuracy: 0.6759
Epoch 60/100
23/23 [==============================] - 3s 148ms/step - loss: 36.0738 - accur
acy: 0.6591 - val_loss: 64.9079 - val_accuracy: 0.6801
```

```
Epoch 61/100
23/23 [==============================] - 3s 148ms/step - loss: 39.1918 - accur
acy: 0.6699 - val_loss: 50.7380 - val_accuracy: 0.6929
Epoch 62/100
23/23 [==============================] - 3s 148ms/step - loss: 35.5496 - accur
acy: 0.6654 - val_loss: 68.6664 - val_accuracy: 0.7064
Epoch 63/100
23/23 [==============================] - 3s 149ms/step - loss: 39.7608 - accur
acy: 0.6622 - val_loss: 55.5388 - val_accuracy: 0.6986
Epoch 64/100
23/23 [==============================] - 3s 148ms/step - loss: 38.2328 - accur
acy: 0.6534 - val_loss: 49.4109 - val_accuracy: 0.7113
Epoch 65/100
23/23 [==============================] - 3s 148ms/step - loss: 33.4784 - accur
acy: 0.6758 - val_loss: 51.3138 - val_accuracy: 0.6156
Epoch 66/100
23/23 [==============================] - 3s 149ms/step - loss: 33.9785 - accur
acy: 0.6361 - val_loss: 78.0631 - val_accuracy: 0.6688
Epoch 67/100
23/23 [==============================] - 3s 148ms/step - loss: 41.6511 - accur
acy: 0.6645 - val_loss: 70.8900 - val_accuracy: 0.6943
Epoch 68/100
23/23 [==============================] - 3s 148ms/step - loss: 36.4899 - accur
acy: 0.6776 - val_loss: 49.5221 - val_accuracy: 0.6979
Epoch 69/100
23/23 [==============================] - 3s 149ms/step - loss: 36.3008 - accur
acy: 0.6759 - val_loss: 52.2193 - val_accuracy: 0.6943
Epoch 70/100
23/23 [==============================] - 3s 149ms/step - loss: 28.3786 - accur
acy: 0.6744 - val_loss: 53.2602 - val_accuracy: 0.6993
Epoch 71/100
23/23 [==============================] - 3s 149ms/step - loss: 29.6022 - accur
acy: 0.6868 - val_loss: 114.2544 - val_accuracy: 0.6957
Epoch 72/100
23/23 [==============================] - 3s 149ms/step - loss: 48.4528 - accur
acy: 0.6726 - val_loss: 51.4631 - val_accuracy: 0.6894
Epoch 73/100
23/23 [==============================] - 3s 149ms/step - loss: 28.9211 - accur
acy: 0.6836 - val_loss: 50.0005 - val_accuracy: 0.7057
Epoch 74/100
23/23 [==============================] - 3s 149ms/step - loss: 28.8911 - accur
acy: 0.6642 - val_loss: 59.5929 - val_accuracy: 0.6901
Epoch 75/100
23/23 [==============================] - 3s 149ms/step - loss: 31.4430 - accur
acy: 0.6629 - val_loss: 49.1747 - val_accuracy: 0.7035
Epoch 76/100
23/23 [==============================] - 3s 148ms/step - loss: 25.3338 - accur
acy: 0.6805 - val_loss: 101.4220 - val_accuracy: 0.6936
Epoch 77/100
23/23 [==============================] - 3s 149ms/step - loss: 39.9271 - accur
acy: 0.6799 - val_loss: 60.9569 - val_accuracy: 0.6730
Epoch 78/100
23/23 [==============================] - 3s 148ms/step - loss: 32.4513 - accur
acy: 0.6657 - val_loss: 52.3254 - val_accuracy: 0.6851
Epoch 79/100
23/23 [==============================] - 3s 149ms/step - loss: 28.8150 - accur
acy: 0.6695 - val_loss: 51.7736 - val_accuracy: 0.6844
Epoch 80/100
23/23 [==============================] - 3s 149ms/step - loss: 25.9778 - accur
acy: 0.6652 - val_loss: 49.8757 - val_accuracy: 0.7014
```

```
Epoch 81/100
23/23 [==============================] - 3s 148ms/step - loss: 23.4374 - accur
acy: 0.6761 - val_loss: 61.1254 - val_accuracy: 0.6943
Epoch 82/100
23/23 [==============================] - 3s 149ms/step - loss: 28.7301 - accur
acy: 0.6750 - val_loss: 74.9950 - val_accuracy: 0.6872
Epoch 83/100
23/23 [==============================] - 3s 148ms/step - loss: 34.9349 - accur
acy: 0.6912 - val_loss: 50.9989 - val_accuracy: 0.7121
Epoch 84/100
23/23 [==============================] - 3s 149ms/step - loss: 23.9447 - accur
acy: 0.6665 - val_loss: 49.5962 - val_accuracy: 0.6943
Epoch 85/100
23/23 [==============================] - 3s 148ms/step - loss: 23.2514 - accur
acy: 0.6956 - val_loss: 49.0420 - val_accuracy: 0.7000
Epoch 86/100
23/23 [==============================] - 3s 148ms/step - loss: 20.7369 - accur
acy: 0.6934 - val_loss: 49.9935 - val_accuracy: 0.6794
Epoch 87/100
23/23 [==============================] - 3s 149ms/step - loss: 19.9808 - accur
acy: 0.6700 - val_loss: 60.3992 - val_accuracy: 0.7000
Epoch 88/100
23/23 [==============================] - 3s 149ms/step - loss: 29.2610 - accur
acy: 0.6878 - val_loss: 63.2707 - val_accuracy: 0.6979
Epoch 89/100
23/23 [==============================] - 3s 149ms/step - loss: 23.3270 - accur
acy: 0.7083 - val_loss: 49.4041 - val_accuracy: 0.6809
Epoch 90/100
23/23 [==============================] - 3s 148ms/step - loss: 19.5220 - accur
acy: 0.7022 - val_loss: 50.5838 - val_accuracy: 0.6986
Epoch 91/100
23/23 [==============================] - 3s 149ms/step - loss: 17.9123 - accur
acy: 0.6996 - val_loss: 49.9242 - val_accuracy: 0.6908
Epoch 92/100
23/23 [==============================] - 3s 149ms/step - loss: 18.3454 - accur
acy: 0.6947 - val_loss: 60.8623 - val_accuracy: 0.6972
Epoch 93/100
23/23 [==============================] - 3s 149ms/step - loss: 23.7847 - accur
acy: 0.6995 - val_loss: 80.8243 - val_accuracy: 0.6759
Epoch 94/100
23/23 [==============================] - 3s 148ms/step - loss: 25.3840 - accur
acy: 0.6954 - val_loss: 50.8452 - val_accuracy: 0.6809
Epoch 95/100
23/23 [==============================] - 3s 149ms/step - loss: 19.8317 - accur
acy: 0.7096 - val_loss: 55.7776 - val_accuracy: 0.6943
Epoch 96/100
23/23 [==============================] - 3s 149ms/step - loss: 15.7526 - accur
acy: 0.7112 - val_loss: 49.7044 - val_accuracy: 0.6986
Epoch 97/100
23/23 [==============================] - 3s 149ms/step - loss: 14.4444 - accur
acy: 0.7079 - val_loss: 48.2551 - val_accuracy: 0.6936
Epoch 98/100
23/23 [==============================] - 3s 149ms/step - loss: 15.9809 - accur
acy: 0.6946 - val_loss: 51.3659 - val_accuracy: 0.6851
Epoch 99/100
23/23 [==============================] - 3s 148ms/step - loss: 13.7429 - accur
acy: 0.6921 - val_loss: 56.2173 - val_accuracy: 0.6766
Epoch 100/100
23/23 [==============================] - 3s 149ms/step - loss: 17.5017 - accur
acy: 0.7044 - val_loss: 48.4686 - val_accuracy: 0.6865
```

In [49]:
```
plt.figure(figsize=(8,8))
plt.plot(history_knn_impute.history['loss'])
plt.plot(history_knn_impute.history['val_loss'])
plt.title('Base Model loss',fontsize=16)
plt.ylabel('Loss',fontsize=16)
plt.xlabel('Epoch',fontsize=16)
plt.legend(['training data', 'validation data'], loc='upper right',fontsize=14)
```

Out[49]: <matplotlib.legend.Legend at 0x7f1ee24ff290>



Replacing the missing values by k nearest neighbor values brings down our accuracy to 0.704 for the training set and the 0.656 for the dev set. However, the agreement of the model performance (loss and accuracy vs the number of epochs) for both training and dev dataset is great.

We are on the right track, but can we make the accuracy better ?

Dropping NA values

Here we try to drop missing values instead of filling them in. With this approach, we only keep our highest quality data.

In [51]:
```
df_train_drop_na = df_train.dropna()
#df_train_drop_na = df_train_drop_na.reset_index(drop=True)
```

In [52]:
```
df_train_drop_na['Image'] = df_train_drop_na['Image'].apply(lambda x: np.fromst
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: SettingWithCop
yWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/st
able/user_guide/indexing.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
```

In [53]:
```
X,y = process_df(df_train_drop_na)
```

```
(2140, 96, 96)
(2140, 96, 96, 1)
```

In [54]:
```
X_train, X_dev, y_train, y_dev = train_test_split(X, y, test_size=0.2, random_s
```

In [55]:
```
#Applying CNN model
model = Sequential()

model.add(Conv2D(32, (3,3), padding='same', use_bias=False, input_shape=(96,96,
model.add(LeakyReLU(alpha = 0.1))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(128, (3,3), padding='same', use_bias=False))

model.add(Flatten())
model.add(Dropout(0.2))
model.add(Dense(30,activation='elu'))
```

In [56]:
```
model_drop_na = model
```

In [57]:
```
model_drop_na.compile(optimizer='adam',
              loss='mean_squared_error',
              metrics=['accuracy'])
```

In [58]:
```
history_drop_na=model_drop_na.fit(X_train,y_train,epochs = 100,batch_size = 256
```

```
Epoch 1/100
7/7 [==============================] - 4s 464ms/step - loss: 3654.2307 - accur
acy: 0.2296 - val_loss: 870.8874 - val_accuracy: 0.6869
Epoch 2/100
7/7 [==============================] - 1s 151ms/step - loss: 690.0060 - accura
cy: 0.4831 - val_loss: 208.1756 - val_accuracy: 0.0257
Epoch 3/100
7/7 [==============================] - 1s 151ms/step - loss: 241.8057 - accura
cy: 0.0873 - val_loss: 180.9249 - val_accuracy: 0.7103
Epoch 4/100
7/7 [==============================] - 1s 151ms/step - loss: 179.2873 - accura
cy: 0.6610 - val_loss: 153.4494 - val_accuracy: 0.7150
Epoch 5/100
7/7 [==============================] - 1s 164ms/step - loss: 155.2622 - accura
cy: 0.7097 - val_loss: 119.0292 - val_accuracy: 0.6846
Epoch 6/100
7/7 [==============================] - 1s 152ms/step - loss: 131.4468 - accura
cy: 0.4907 - val_loss: 115.0534 - val_accuracy: 0.6262
Epoch 7/100
7/7 [==============================] - 1s 151ms/step - loss: 115.6772 - accura
cy: 0.6407 - val_loss: 113.7549 - val_accuracy: 0.7079
Epoch 8/100
7/7 [==============================] - 1s 151ms/step - loss: 111.5806 - accura
cy: 0.6996 - val_loss: 110.8162 - val_accuracy: 0.6916
Epoch 9/100
7/7 [==============================] - 1s 151ms/step - loss: 111.7481 - accura
cy: 0.6829 - val_loss: 110.5119 - val_accuracy: 0.6986
Epoch 10/100
7/7 [==============================] - 1s 151ms/step - loss: 108.5572 - accura
cy: 0.6948 - val_loss: 111.3105 - val_accuracy: 0.6939
Epoch 11/100
7/7 [==============================] - 1s 151ms/step - loss: 110.8121 - accura
cy: 0.6776 - val_loss: 118.4326 - val_accuracy: 0.6939
Epoch 12/100
7/7 [==============================] - 1s 151ms/step - loss: 112.8006 - accura
cy: 0.6930 - val_loss: 106.8800 - val_accuracy: 0.6986
Epoch 13/100
7/7 [==============================] - 1s 150ms/step - loss: 103.7660 - accura
cy: 0.7011 - val_loss: 104.8786 - val_accuracy: 0.6916
Epoch 14/100
7/7 [==============================] - 1s 150ms/step - loss: 101.0331 - accura
cy: 0.6870 - val_loss: 103.6887 - val_accuracy: 0.6986
Epoch 15/100
7/7 [==============================] - 1s 152ms/step - loss: 96.3701 - accurac
y: 0.6911 - val_loss: 107.2504 - val_accuracy: 0.6986
Epoch 16/100
7/7 [==============================] - 1s 153ms/step - loss: 103.2787 - accura
cy: 0.6945 - val_loss: 100.4560 - val_accuracy: 0.6939
Epoch 17/100
7/7 [==============================] - 1s 151ms/step - loss: 95.7495 - accurac
y: 0.6824 - val_loss: 98.7006 - val_accuracy: 0.6916
Epoch 18/100
7/7 [==============================] - 1s 151ms/step - loss: 91.5896 - accurac
y: 0.6861 - val_loss: 98.4912 - val_accuracy: 0.6939
Epoch 19/100
7/7 [==============================] - 1s 151ms/step - loss: 91.7112 - accurac
y: 0.6997 - val_loss: 99.6339 - val_accuracy: 0.6916
Epoch 20/100
7/7 [==============================] - 1s 151ms/step - loss: 90.4376 - accurac
y: 0.6964 - val_loss: 94.0948 - val_accuracy: 0.6963
```

```
Epoch 21/100
7/7 [==============================] - 1s 152ms/step - loss: 92.4781 - accurac
y: 0.6929 - val_loss: 103.4442 - val_accuracy: 0.6916
Epoch 22/100
7/7 [==============================] - 1s 152ms/step - loss: 90.6783 - accurac
y: 0.6929 - val_loss: 91.8392 - val_accuracy: 0.6916
Epoch 23/100
7/7 [==============================] - 1s 151ms/step - loss: 83.6917 - accurac
y: 0.6841 - val_loss: 103.9142 - val_accuracy: 0.7056
Epoch 24/100
7/7 [==============================] - 1s 152ms/step - loss: 88.0289 - accurac
y: 0.6886 - val_loss: 86.6916 - val_accuracy: 0.6963
Epoch 25/100
7/7 [==============================] - 1s 164ms/step - loss: 81.5394 - accurac
y: 0.7000 - val_loss: 85.1416 - val_accuracy: 0.6939
Epoch 26/100
7/7 [==============================] - 1s 150ms/step - loss: 78.7361 - accurac
y: 0.6768 - val_loss: 89.8719 - val_accuracy: 0.7056
Epoch 27/100
7/7 [==============================] - 1s 153ms/step - loss: 77.8129 - accurac
y: 0.6846 - val_loss: 82.4336 - val_accuracy: 0.7033
Epoch 28/100
7/7 [==============================] - 1s 151ms/step - loss: 74.2694 - accurac
y: 0.7005 - val_loss: 79.4135 - val_accuracy: 0.7033
Epoch 29/100
7/7 [==============================] - 1s 152ms/step - loss: 75.0013 - accurac
y: 0.6921 - val_loss: 82.6672 - val_accuracy: 0.6822
Epoch 30/100
7/7 [==============================] - 1s 151ms/step - loss: 84.7717 - accurac
y: 0.6802 - val_loss: 78.8110 - val_accuracy: 0.7126
Epoch 31/100
7/7 [==============================] - 1s 151ms/step - loss: 80.4357 - accurac
y: 0.7007 - val_loss: 108.5227 - val_accuracy: 0.7056
Epoch 32/100
7/7 [==============================] - 1s 151ms/step - loss: 87.5247 - accurac
y: 0.6838 - val_loss: 86.8174 - val_accuracy: 0.6963
Epoch 33/100
7/7 [==============================] - 1s 151ms/step - loss: 79.0135 - accurac
y: 0.6914 - val_loss: 82.0365 - val_accuracy: 0.7079
Epoch 34/100
7/7 [==============================] - 1s 150ms/step - loss: 73.4101 - accurac
y: 0.6876 - val_loss: 93.2229 - val_accuracy: 0.6916
Epoch 35/100
7/7 [==============================] - 1s 151ms/step - loss: 82.9690 - accurac
y: 0.6664 - val_loss: 76.1787 - val_accuracy: 0.6916
Epoch 36/100
7/7 [==============================] - 1s 152ms/step - loss: 70.8298 - accurac
y: 0.6957 - val_loss: 86.4989 - val_accuracy: 0.7033
Epoch 37/100
7/7 [==============================] - 1s 152ms/step - loss: 72.4448 - accurac
y: 0.6978 - val_loss: 70.4009 - val_accuracy: 0.6005
Epoch 38/100
7/7 [==============================] - 1s 151ms/step - loss: 68.0871 - accurac
y: 0.6598 - val_loss: 67.7438 - val_accuracy: 0.7150
Epoch 39/100
7/7 [==============================] - 1s 164ms/step - loss: 60.4639 - accurac
y: 0.6910 - val_loss: 66.4477 - val_accuracy: 0.6776
Epoch 40/100
7/7 [==============================] - 1s 151ms/step - loss: 61.6274 - accurac
y: 0.6894 - val_loss: 88.2872 - val_accuracy: 0.7150
```

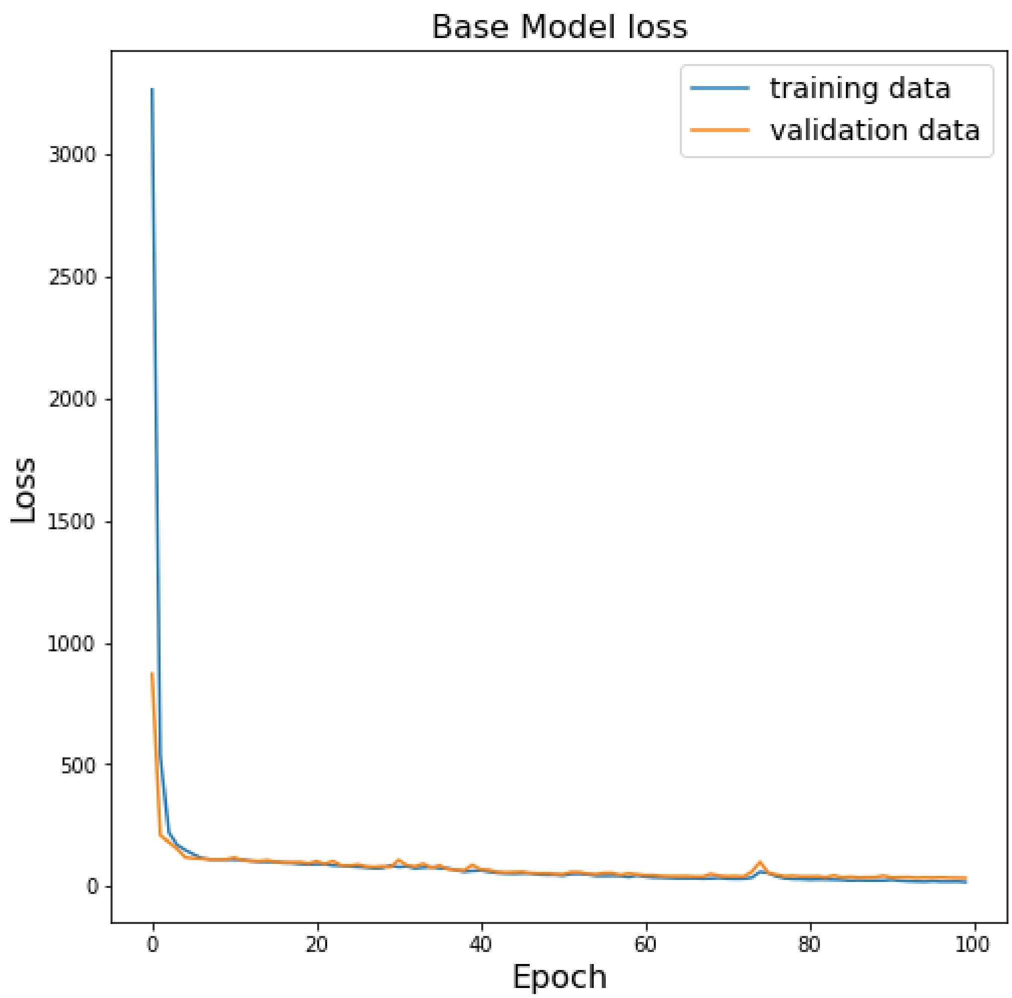
```
Epoch 41/100
7/7 [==============================] - 1s 150ms/step - loss: 67.0245 - accurac
y: 0.6808 - val_loss: 69.3787 - val_accuracy: 0.6963
Epoch 42/100
7/7 [==============================] - 1s 152ms/step - loss: 62.1374 - accurac
y: 0.7081 - val_loss: 67.7417 - val_accuracy: 0.7173
Epoch 43/100
7/7 [==============================] - 1s 152ms/step - loss: 55.0619 - accurac
y: 0.7110 - val_loss: 59.1175 - val_accuracy: 0.7173
Epoch 44/100
7/7 [==============================] - 1s 151ms/step - loss: 53.7829 - accurac
y: 0.6708 - val_loss: 57.3463 - val_accuracy: 0.7126
Epoch 45/100
7/7 [==============================] - 1s 150ms/step - loss: 50.4324 - accurac
y: 0.7062 - val_loss: 58.7546 - val_accuracy: 0.7103
Epoch 46/100
7/7 [==============================] - 1s 153ms/step - loss: 52.4831 - accurac
y: 0.7211 - val_loss: 59.9405 - val_accuracy: 0.7126
Epoch 47/100
7/7 [==============================] - 1s 151ms/step - loss: 51.3181 - accurac
y: 0.7005 - val_loss: 54.7260 - val_accuracy: 0.7079
Epoch 48/100
7/7 [==============================] - 1s 151ms/step - loss: 47.8687 - accurac
y: 0.7021 - val_loss: 53.3082 - val_accuracy: 0.6752
Epoch 49/100
7/7 [==============================] - 1s 151ms/step - loss: 46.2538 - accurac
y: 0.7014 - val_loss: 53.9630 - val_accuracy: 0.7150
Epoch 50/100
7/7 [==============================] - 1s 151ms/step - loss: 46.5754 - accurac
y: 0.7000 - val_loss: 50.7689 - val_accuracy: 0.6939
Epoch 51/100
7/7 [==============================] - 1s 150ms/step - loss: 45.1900 - accurac
y: 0.6735 - val_loss: 50.1546 - val_accuracy: 0.7126
Epoch 52/100
7/7 [==============================] - 1s 150ms/step - loss: 48.6229 - accurac
y: 0.7033 - val_loss: 58.7421 - val_accuracy: 0.7079
Epoch 53/100
7/7 [==============================] - 1s 150ms/step - loss: 51.6931 - accurac
y: 0.7095 - val_loss: 58.5223 - val_accuracy: 0.7220
Epoch 54/100
7/7 [==============================] - 1s 164ms/step - loss: 47.8444 - accurac
y: 0.7086 - val_loss: 51.7766 - val_accuracy: 0.7150
Epoch 55/100
7/7 [==============================] - 1s 152ms/step - loss: 42.2050 - accurac
y: 0.7123 - val_loss: 50.3349 - val_accuracy: 0.6869
Epoch 56/100
7/7 [==============================] - 1s 150ms/step - loss: 41.5848 - accurac
y: 0.7223 - val_loss: 54.9656 - val_accuracy: 0.7150
Epoch 57/100
7/7 [==============================] - 1s 151ms/step - loss: 43.0715 - accurac
y: 0.7189 - val_loss: 54.8469 - val_accuracy: 0.7033
Epoch 58/100
7/7 [==============================] - 1s 149ms/step - loss: 44.3901 - accurac
y: 0.6999 - val_loss: 47.1507 - val_accuracy: 0.7173
Epoch 59/100
7/7 [==============================] - 1s 150ms/step - loss: 38.3299 - accurac
y: 0.7214 - val_loss: 53.2548 - val_accuracy: 0.7056
Epoch 60/100
7/7 [==============================] - 1s 151ms/step - loss: 47.0253 - accurac
y: 0.7067 - val_loss: 49.6065 - val_accuracy: 0.7009
```

```
Epoch 61/100
7/7 [==============================] - 1s 151ms/step - loss: 40.9013 - accurac
y: 0.7055 - val_loss: 45.7324 - val_accuracy: 0.7243
Epoch 62/100
7/7 [==============================] - 1s 151ms/step - loss: 36.1899 - accurac
y: 0.7187 - val_loss: 46.0727 - val_accuracy: 0.7056
Epoch 63/100
7/7 [==============================] - 1s 150ms/step - loss: 36.3646 - accurac
y: 0.7056 - val_loss: 42.6774 - val_accuracy: 0.7150
Epoch 64/100
7/7 [==============================] - 1s 151ms/step - loss: 34.2125 - accurac
y: 0.7181 - val_loss: 41.9763 - val_accuracy: 0.7126
Epoch 65/100
7/7 [==============================] - 1s 152ms/step - loss: 33.0733 - accurac
y: 0.7207 - val_loss: 41.6354 - val_accuracy: 0.6893
Epoch 66/100
7/7 [==============================] - 1s 150ms/step - loss: 32.4088 - accurac
y: 0.7065 - val_loss: 42.3136 - val_accuracy: 0.7150
Epoch 67/100
7/7 [==============================] - 1s 149ms/step - loss: 33.3517 - accurac
y: 0.7293 - val_loss: 40.2500 - val_accuracy: 0.7009
Epoch 68/100
7/7 [==============================] - 1s 150ms/step - loss: 32.2929 - accurac
y: 0.7326 - val_loss: 39.9855 - val_accuracy: 0.7150
Epoch 69/100
7/7 [==============================] - 1s 151ms/step - loss: 32.0029 - accurac
y: 0.7110 - val_loss: 51.0777 - val_accuracy: 0.7243
Epoch 70/100
7/7 [==============================] - 1s 151ms/step - loss: 35.2638 - accurac
y: 0.7342 - val_loss: 44.0443 - val_accuracy: 0.7360
Epoch 71/100
7/7 [==============================] - 1s 150ms/step - loss: 31.6683 - accurac
y: 0.7251 - val_loss: 40.7434 - val_accuracy: 0.7220
Epoch 72/100
7/7 [==============================] - 1s 151ms/step - loss: 32.4739 - accurac
y: 0.7296 - val_loss: 42.4790 - val_accuracy: 0.7196
Epoch 73/100
7/7 [==============================] - 1s 151ms/step - loss: 32.3064 - accurac
y: 0.7247 - val_loss: 40.0549 - val_accuracy: 0.7103
Epoch 74/100
7/7 [==============================] - 1s 152ms/step - loss: 33.5671 - accurac
y: 0.7156 - val_loss: 59.9113 - val_accuracy: 0.7290
Epoch 75/100
7/7 [==============================] - 1s 152ms/step - loss: 56.9740 - accurac
y: 0.7360 - val_loss: 100.7020 - val_accuracy: 0.7009
Epoch 76/100
7/7 [==============================] - 1s 151ms/step - loss: 60.4268 - accurac
y: 0.7175 - val_loss: 55.6664 - val_accuracy: 0.7266
Epoch 77/100
7/7 [==============================] - 1s 151ms/step - loss: 41.4352 - accurac
y: 0.7259 - val_loss: 48.2760 - val_accuracy: 0.7243
Epoch 78/100
7/7 [==============================] - 1s 151ms/step - loss: 33.5781 - accurac
y: 0.7203 - val_loss: 41.7272 - val_accuracy: 0.7056
Epoch 79/100
7/7 [==============================] - 1s 150ms/step - loss: 31.7930 - accurac
y: 0.7303 - val_loss: 43.2835 - val_accuracy: 0.6752
Epoch 80/100
7/7 [==============================] - 1s 151ms/step - loss: 29.5468 - accurac
y: 0.7204 - val_loss: 40.7482 - val_accuracy: 0.7313
```

```
Epoch 81/100
7/7 [==============================] - 1s 152ms/step - loss: 29.8896 - accurac
y: 0.7260 - val_loss: 40.8462 - val_accuracy: 0.7173
Epoch 82/100
7/7 [==============================] - 1s 153ms/step - loss: 28.4476 - accurac
y: 0.7411 - val_loss: 41.4451 - val_accuracy: 0.7290
Epoch 83/100
7/7 [==============================] - 1s 150ms/step - loss: 28.5137 - accurac
y: 0.7430 - val_loss: 37.3875 - val_accuracy: 0.7266
Epoch 84/100
7/7 [==============================] - 1s 150ms/step - loss: 26.9242 - accurac
y: 0.7599 - val_loss: 44.5241 - val_accuracy: 0.7313
Epoch 85/100
7/7 [==============================] - 1s 153ms/step - loss: 27.5895 - accurac
y: 0.7382 - val_loss: 37.2906 - val_accuracy: 0.7056
Epoch 86/100
7/7 [==============================] - 1s 152ms/step - loss: 24.0364 - accurac
y: 0.7333 - val_loss: 39.0999 - val_accuracy: 0.7336
Epoch 87/100
7/7 [==============================] - 1s 150ms/step - loss: 26.2805 - accurac
y: 0.7270 - val_loss: 36.4109 - val_accuracy: 0.7360
Epoch 88/100
7/7 [==============================] - 1s 151ms/step - loss: 23.5770 - accurac
y: 0.7455 - val_loss: 37.6831 - val_accuracy: 0.7313
Epoch 89/100
7/7 [==============================] - 1s 150ms/step - loss: 23.8588 - accurac
y: 0.7451 - val_loss: 37.5432 - val_accuracy: 0.6963
Epoch 90/100
7/7 [==============================] - 1s 151ms/step - loss: 24.4647 - accurac
y: 0.7321 - val_loss: 43.6159 - val_accuracy: 0.7407
Epoch 91/100
7/7 [==============================] - 1s 151ms/step - loss: 25.7563 - accurac
y: 0.7390 - val_loss: 36.7539 - val_accuracy: 0.7313
Epoch 92/100
7/7 [==============================] - 1s 153ms/step - loss: 24.5578 - accurac
y: 0.7621 - val_loss: 37.4718 - val_accuracy: 0.7266
Epoch 93/100
7/7 [==============================] - 1s 151ms/step - loss: 21.3947 - accurac
y: 0.7479 - val_loss: 37.8671 - val_accuracy: 0.7290
Epoch 94/100
7/7 [==============================] - 1s 153ms/step - loss: 21.6827 - accurac
y: 0.7384 - val_loss: 35.5046 - val_accuracy: 0.7313
Epoch 95/100
7/7 [==============================] - 1s 164ms/step - loss: 18.9134 - accurac
y: 0.7344 - val_loss: 36.6192 - val_accuracy: 0.7336
Epoch 96/100
7/7 [==============================] - 1s 154ms/step - loss: 20.6626 - accurac
y: 0.7565 - val_loss: 35.8933 - val_accuracy: 0.7336
Epoch 97/100
7/7 [==============================] - 1s 152ms/step - loss: 19.6974 - accurac
y: 0.7424 - val_loss: 37.2326 - val_accuracy: 0.7173
Epoch 98/100
7/7 [==============================] - 1s 150ms/step - loss: 20.1587 - accurac
y: 0.7456 - val_loss: 35.4937 - val_accuracy: 0.7243
Epoch 99/100
7/7 [==============================] - 1s 152ms/step - loss: 20.2755 - accurac
y: 0.7602 - val_loss: 34.9513 - val_accuracy: 0.7453
Epoch 100/100
7/7 [==============================] - 1s 150ms/step - loss: 18.2805 - accurac
y: 0.7556 - val_loss: 34.4209 - val_accuracy: 0.7290
```

In [60]:
```
plt.figure(figsize=(8,8))
plt.plot(history_drop_na.history['loss'])
plt.plot(history_drop_na.history['val_loss'])
plt.title('Base Model loss',fontsize=16)
plt.ylabel('Loss',fontsize=16)
plt.xlabel('Epoch',fontsize=16)
plt.legend(['training data', 'validation data'], loc='upper right',fontsize=14)
```

Out[60]:
```
<matplotlib.legend.Legend at 0x7f1ed6576610>
```



Data Augmentation

Here we augment our data and add it to the training data so that our model will train on a variety of image types and image angles. Data Augmentation is a popular way to get more data by making minor alterations to our existing dataset. To implement these transformations and rotations we used the ndimage package from scipy.

In [61]:
```
from scipy import ndimage
```

In [62]:
```
df_train = pd.read_csv('training.csv')
df_train.shape
```

Out[62]:
```
(7049, 31)
```

In [63]:
```
df_train['Image'] = df_train['Image'].apply(lambda x: np.fromstring(x, dtype=ir
```

In [64]:
```
df_train = df_train.dropna()
df_train = df_train.reset_index(drop=True)
df_train.shape
```

Out[64]:
```
(2140, 31)
```

In [65]:
```
slice_num = int(df_train.shape[0]/4)
slice_num
```

Out[65]:
```
535
```

In [66]:
```
#created eight different slices of the data
df_train_slice_1a = df_train.iloc[:slice_num].copy()
df_train_slice_1b = df_train.iloc[:slice_num].copy()

df_train_slice_2a = df_train.iloc[slice_num:slice_num*2].copy()
df_train_slice_2b = df_train.iloc[slice_num:slice_num*2].copy()

df_train_slice_3a = df_train.iloc[2*slice_num:3*slice_num].copy()
df_train_slice_3b = df_train.iloc[2*slice_num:3*slice_num].copy()

df_train_slice_4a = df_train.iloc[3*slice_num:].copy()
df_train_slice_4b = df_train.iloc[3*slice_num:].copy()
```

In [67]:
```
#added a Gaussian blurring and 90 degree rotation to a fourth of the data
df_train_slice_1a['Image'] = df_train_slice_1a['Image'].apply(lambda row: ndima
df_train_slice_1b['Image'] = df_train_slice_1b['Image'].apply(lambda row: ndima
#added a 180 and 270 degree rotation to a fourth of the data
df_train_slice_2a['Image'] = df_train_slice_2a['Image'].apply(lambda row: ndima
df_train_slice_2b['Image'] = df_train_slice_2b['Image'].apply(lambda row: ndima
#added a gaussian filter and a increased the brightness to a fourth of the data
df_train_slice_3a['Image'] = df_train_slice_3a['Image'].apply(lambda row: ndima
df_train_slice_3b['Image'] = df_train_slice_3b['Image'].apply(lambda row: row +
#shifted the images in two different directions
df_train_slice_4a['Image'] = df_train_slice_4a['Image'].apply(lambda row: ndima
df_train_slice_4b['Image'] = df_train_slice_4b['Image'].apply(lambda row: ndima
```

In [68]:
```
#concatenated all the slices of training data to the orginial training data
df_train_augment = pd.concat([df_train,df_train_slice_1a,df_train_slice_2a,df_t
```

In [69]:
```
X,y = process_df(df_train_augment)
```

```
(6420, 96, 96)
(6420, 96, 96, 1)
```

In [70]:
```
X_train, X_dev, y_train, y_dev = train_test_split(X, y, test_size=0.2, random_s
```

In [71]:
```
model = Sequential()

model.add(Conv2D(32, (3,3), padding='same', use_bias=False, input_shape=(96,96,
model.add(LeakyReLU(alpha = 0.1))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
```

```
model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3,3), padding='same', use_bias=False))
model.add(LeakyReLU(alpha = 0.1))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(128, (3,3), padding='same', use_bias=False))

model.add(Flatten())
model.add(Dropout(0.2))
model.add(Dense(30,activation='elu'))
```

In [72]:

```
model_augment = model
```

In [73]:

```
model_augment.compile(optimizer='adam',
              loss='mean_squared_error',
              metrics=['accuracy'])
```

In [74]:

```
history_augment=model_augment.fit(X_train,y_train,epochs = 200,batch_size = 256
```

```
Epoch 1/200
21/21 [==============================] - 5s 178ms/step - loss: 2560.3956 - acc
uracy: 0.2732 - val_loss: 168.9467 - val_accuracy: 0.7017
Epoch 2/200
21/21 [==============================] - 3s 148ms/step - loss: 176.6448 - accu
racy: 0.6864 - val_loss: 148.3164 - val_accuracy: 0.7017
Epoch 3/200
21/21 [==============================] - 3s 148ms/step - loss: 147.1530 - accu
racy: 0.6786 - val_loss: 136.9845 - val_accuracy: 0.7025
Epoch 4/200
21/21 [==============================] - 3s 148ms/step - loss: 133.4368 - accu
racy: 0.6782 - val_loss: 137.7667 - val_accuracy: 0.6970
Epoch 5/200
21/21 [==============================] - 3s 148ms/step - loss: 132.8047 - accu
racy: 0.6668 - val_loss: 146.5528 - val_accuracy: 0.6900
Epoch 6/200
21/21 [==============================] - 3s 148ms/step - loss: 138.5589 - accu
racy: 0.6712 - val_loss: 146.6472 - val_accuracy: 0.6939
Epoch 7/200
21/21 [==============================] - 3s 148ms/step - loss: 130.4639 - accu
racy: 0.6837 - val_loss: 114.3054 - val_accuracy: 0.6939
Epoch 8/200
21/21 [==============================] - 3s 148ms/step - loss: 120.3496 - accu
racy: 0.6889 - val_loss: 126.4413 - val_accuracy: 0.6970
Epoch 9/200
21/21 [==============================] - 3s 149ms/step - loss: 113.9748 - accu
racy: 0.6902 - val_loss: 121.8548 - val_accuracy: 0.6955
Epoch 10/200
21/21 [==============================] - 3s 148ms/step - loss: 117.9809 - accu
racy: 0.6959 - val_loss: 121.3859 - val_accuracy: 0.6970
Epoch 11/200
21/21 [==============================] - 3s 148ms/step - loss: 116.4130 - accu
racy: 0.6904 - val_loss: 114.1011 - val_accuracy: 0.7040
Epoch 12/200
21/21 [==============================] - 3s 148ms/step - loss: 108.0239 - accu
racy: 0.6847 - val_loss: 121.9698 - val_accuracy: 0.6970
Epoch 13/200
21/21 [==============================] - 3s 148ms/step - loss: 111.6195 - accu
racy: 0.6784 - val_loss: 101.5714 - val_accuracy: 0.6994
Epoch 14/200
21/21 [==============================] - 3s 148ms/step - loss: 98.9643 - accur
acy: 0.6902 - val_loss: 120.4386 - val_accuracy: 0.6986
Epoch 15/200
21/21 [==============================] - 3s 154ms/step - loss: 100.4922 - accu
racy: 0.6815 - val_loss: 122.3647 - val_accuracy: 0.6947
Epoch 16/200
21/21 [==============================] - 3s 148ms/step - loss: 108.7496 - accu
racy: 0.6769 - val_loss: 136.4570 - val_accuracy: 0.7048
Epoch 17/200
21/21 [==============================] - 3s 148ms/step - loss: 107.2415 - accu
racy: 0.6757 - val_loss: 134.3191 - val_accuracy: 0.6939
Epoch 18/200
21/21 [==============================] - 3s 149ms/step - loss: 115.0258 - accu
racy: 0.6854 - val_loss: 120.7891 - val_accuracy: 0.7002
Epoch 19/200
21/21 [==============================] - 3s 149ms/step - loss: 99.0868 - accur
acy: 0.6821 - val_loss: 92.6014 - val_accuracy: 0.7025
Epoch 20/200
21/21 [==============================] - 3s 148ms/step - loss: 97.6257 - accur
acy: 0.6915 - val_loss: 98.7767 - val_accuracy: 0.6986
```

```
Epoch 21/200
21/21 [==============================] - 3s 148ms/step - loss: 87.1391 - accur
acy: 0.6801 - val_loss: 86.8349 - val_accuracy: 0.7033
Epoch 22/200
21/21 [==============================] - 3s 154ms/step - loss: 86.3152 - accur
acy: 0.6864 - val_loss: 97.8043 - val_accuracy: 0.7017
Epoch 23/200
21/21 [==============================] - 3s 149ms/step - loss: 87.4074 - accur
acy: 0.6933 - val_loss: 102.8431 - val_accuracy: 0.7009
Epoch 24/200
21/21 [==============================] - 3s 149ms/step - loss: 88.6143 - accur
acy: 0.6851 - val_loss: 85.0874 - val_accuracy: 0.7017
Epoch 25/200
21/21 [==============================] - 3s 148ms/step - loss: 77.4805 - accur
acy: 0.6903 - val_loss: 76.6253 - val_accuracy: 0.7056
Epoch 26/200
21/21 [==============================] - 3s 148ms/step - loss: 75.7937 - accur
acy: 0.6926 - val_loss: 73.0289 - val_accuracy: 0.6939
Epoch 27/200
21/21 [==============================] - 3s 148ms/step - loss: 71.5914 - accur
acy: 0.6824 - val_loss: 79.9909 - val_accuracy: 0.6978
Epoch 28/200
21/21 [==============================] - 3s 148ms/step - loss: 72.3585 - accur
acy: 0.6926 - val_loss: 67.2067 - val_accuracy: 0.6900
Epoch 29/200
21/21 [==============================] - 3s 149ms/step - loss: 71.5807 - accur
acy: 0.6895 - val_loss: 67.1310 - val_accuracy: 0.6846
Epoch 30/200
21/21 [==============================] - 3s 154ms/step - loss: 60.3306 - accur
acy: 0.6776 - val_loss: 88.5568 - val_accuracy: 0.6924
Epoch 31/200
21/21 [==============================] - 3s 149ms/step - loss: 88.1200 - accur
acy: 0.6925 - val_loss: 137.4253 - val_accuracy: 0.6963
Epoch 32/200
21/21 [==============================] - 3s 149ms/step - loss: 116.7804 - accu
racy: 0.6872 - val_loss: 93.5674 - val_accuracy: 0.7025
Epoch 33/200
21/21 [==============================] - 3s 148ms/step - loss: 85.0203 - accur
acy: 0.6908 - val_loss: 84.6310 - val_accuracy: 0.7040
Epoch 34/200
21/21 [==============================] - 3s 149ms/step - loss: 78.3855 - accur
acy: 0.6829 - val_loss: 77.5540 - val_accuracy: 0.7048
Epoch 35/200
21/21 [==============================] - 3s 148ms/step - loss: 79.4368 - accur
acy: 0.6922 - val_loss: 74.4841 - val_accuracy: 0.6628
Epoch 36/200
21/21 [==============================] - 3s 148ms/step - loss: 67.6211 - accur
acy: 0.6827 - val_loss: 86.0085 - val_accuracy: 0.7009
Epoch 37/200
21/21 [==============================] - 3s 148ms/step - loss: 72.4259 - accur
acy: 0.6900 - val_loss: 71.1058 - val_accuracy: 0.6776
Epoch 38/200
21/21 [==============================] - 3s 148ms/step - loss: 64.5928 - accur
acy: 0.6950 - val_loss: 62.8157 - val_accuracy: 0.7040
Epoch 39/200
21/21 [==============================] - 3s 148ms/step - loss: 57.2763 - accur
acy: 0.7029 - val_loss: 67.0465 - val_accuracy: 0.7087
Epoch 40/200
21/21 [==============================] - 3s 148ms/step - loss: 57.4389 - accur
acy: 0.6994 - val_loss: 61.5836 - val_accuracy: 0.6986
```

```
Epoch 41/200
21/21 [==============================] - 3s 149ms/step - loss: 54.4170 - accur
acy: 0.7021 - val_loss: 58.0882 - val_accuracy: 0.7150
Epoch 42/200
21/21 [==============================] - 3s 148ms/step - loss: 55.2077 - accur
acy: 0.7165 - val_loss: 141.2835 - val_accuracy: 0.7126
Epoch 43/200
21/21 [==============================] - 3s 148ms/step - loss: 79.4171 - accur
acy: 0.6921 - val_loss: 65.6820 - val_accuracy: 0.7235
Epoch 44/200
21/21 [==============================] - 3s 148ms/step - loss: 53.3954 - accur
acy: 0.7146 - val_loss: 61.0392 - val_accuracy: 0.7126
Epoch 45/200
21/21 [==============================] - 3s 148ms/step - loss: 51.3042 - accur
acy: 0.7089 - val_loss: 50.9803 - val_accuracy: 0.7064
Epoch 46/200
21/21 [==============================] - 3s 148ms/step - loss: 45.9732 - accur
acy: 0.7048 - val_loss: 48.4901 - val_accuracy: 0.7150
Epoch 47/200
21/21 [==============================] - 3s 149ms/step - loss: 42.5057 - accur
acy: 0.7202 - val_loss: 66.8198 - val_accuracy: 0.7298
Epoch 48/200
21/21 [==============================] - 3s 148ms/step - loss: 47.5811 - accur
acy: 0.7043 - val_loss: 49.8729 - val_accuracy: 0.7173
Epoch 49/200
21/21 [==============================] - 3s 148ms/step - loss: 41.9696 - accur
acy: 0.7176 - val_loss: 46.1525 - val_accuracy: 0.7188
Epoch 50/200
21/21 [==============================] - 3s 147ms/step - loss: 39.1479 - accur
acy: 0.7115 - val_loss: 68.1604 - val_accuracy: 0.7243
Epoch 51/200
21/21 [==============================] - 3s 147ms/step - loss: 65.3010 - accur
acy: 0.7189 - val_loss: 60.1104 - val_accuracy: 0.7009
Epoch 52/200
21/21 [==============================] - 3s 148ms/step - loss: 49.0117 - accur
acy: 0.7038 - val_loss: 60.5079 - val_accuracy: 0.7103
Epoch 53/200
21/21 [==============================] - 3s 148ms/step - loss: 44.0480 - accur
acy: 0.6980 - val_loss: 45.4619 - val_accuracy: 0.7243
Epoch 54/200
21/21 [==============================] - 3s 149ms/step - loss: 40.4354 - accur
acy: 0.7268 - val_loss: 43.8406 - val_accuracy: 0.7352
Epoch 55/200
21/21 [==============================] - 3s 148ms/step - loss: 41.3043 - accur
acy: 0.7255 - val_loss: 44.3019 - val_accuracy: 0.7212
Epoch 56/200
21/21 [==============================] - 3s 149ms/step - loss: 35.6352 - accur
acy: 0.7212 - val_loss: 44.2032 - val_accuracy: 0.7336
Epoch 57/200
21/21 [==============================] - 3s 154ms/step - loss: 33.1851 - accur
acy: 0.7294 - val_loss: 43.9554 - val_accuracy: 0.7375
Epoch 58/200
21/21 [==============================] - 3s 147ms/step - loss: 35.0185 - accur
acy: 0.7232 - val_loss: 50.6863 - val_accuracy: 0.7422
Epoch 59/200
21/21 [==============================] - 3s 148ms/step - loss: 36.9194 - accur
acy: 0.7151 - val_loss: 46.1615 - val_accuracy: 0.7461
Epoch 60/200
21/21 [==============================] - 3s 148ms/step - loss: 35.0919 - accur
acy: 0.7371 - val_loss: 62.9050 - val_accuracy: 0.7422
```

```
Epoch 61/200
21/21 [==============================] - 3s 148ms/step - loss: 38.7329 - accur
acy: 0.7349 - val_loss: 59.0407 - val_accuracy: 0.7422
Epoch 62/200
21/21 [==============================] - 3s 150ms/step - loss: 35.5012 - accur
acy: 0.7408 - val_loss: 43.0336 - val_accuracy: 0.7453
Epoch 63/200
21/21 [==============================] - 3s 154ms/step - loss: 28.3076 - accur
acy: 0.7396 - val_loss: 41.5988 - val_accuracy: 0.7368
Epoch 64/200
21/21 [==============================] - 3s 148ms/step - loss: 27.9236 - accur
acy: 0.7329 - val_loss: 54.1619 - val_accuracy: 0.7383
Epoch 65/200
21/21 [==============================] - 3s 148ms/step - loss: 32.3189 - accur
acy: 0.7330 - val_loss: 48.3603 - val_accuracy: 0.7500
Epoch 66/200
21/21 [==============================] - 3s 148ms/step - loss: 31.4785 - accur
acy: 0.7377 - val_loss: 49.0330 - val_accuracy: 0.7445
Epoch 67/200
21/21 [==============================] - 3s 149ms/step - loss: 32.2794 - accur
acy: 0.7277 - val_loss: 38.7384 - val_accuracy: 0.7484
Epoch 68/200
21/21 [==============================] - 3s 149ms/step - loss: 26.3569 - accur
acy: 0.7362 - val_loss: 40.2683 - val_accuracy: 0.7508
Epoch 69/200
21/21 [==============================] - 3s 148ms/step - loss: 30.5678 - accur
acy: 0.7433 - val_loss: 40.8001 - val_accuracy: 0.7516
Epoch 70/200
21/21 [==============================] - 3s 148ms/step - loss: 25.6866 - accur
acy: 0.7398 - val_loss: 37.4663 - val_accuracy: 0.7500
Epoch 71/200
21/21 [==============================] - 3s 148ms/step - loss: 23.5804 - accur
acy: 0.7391 - val_loss: 48.2566 - val_accuracy: 0.7547
Epoch 72/200
21/21 [==============================] - 3s 148ms/step - loss: 27.5498 - accur
acy: 0.7427 - val_loss: 41.2159 - val_accuracy: 0.7516
Epoch 73/200
21/21 [==============================] - 3s 147ms/step - loss: 23.4792 - accur
acy: 0.7552 - val_loss: 47.0333 - val_accuracy: 0.7508
Epoch 74/200
21/21 [==============================] - 3s 148ms/step - loss: 23.9183 - accur
acy: 0.7389 - val_loss: 60.5669 - val_accuracy: 0.7492
Epoch 75/200
21/21 [==============================] - 3s 147ms/step - loss: 27.8805 - accur
acy: 0.7535 - val_loss: 37.5579 - val_accuracy: 0.7484
Epoch 76/200
21/21 [==============================] - 3s 148ms/step - loss: 23.5481 - accur
acy: 0.7372 - val_loss: 45.6291 - val_accuracy: 0.7578
Epoch 77/200
21/21 [==============================] - 3s 147ms/step - loss: 22.3065 - accur
acy: 0.7480 - val_loss: 34.7808 - val_accuracy: 0.7601
Epoch 78/200
21/21 [==============================] - 3s 148ms/step - loss: 20.5455 - accur
acy: 0.7580 - val_loss: 43.0969 - val_accuracy: 0.7469
Epoch 79/200
21/21 [==============================] - 3s 149ms/step - loss: 23.8569 - accur
acy: 0.7589 - val_loss: 41.5178 - val_accuracy: 0.7555
Epoch 80/200
21/21 [==============================] - 3s 148ms/step - loss: 22.1201 - accur
acy: 0.7434 - val_loss: 40.6774 - val_accuracy: 0.7477
```

```
Epoch 81/200
21/21 [==============================] - 3s 149ms/step - loss: 21.6988 - accur
acy: 0.7577 - val_loss: 35.5753 - val_accuracy: 0.7578
Epoch 82/200
21/21 [==============================] - 3s 148ms/step - loss: 19.3007 - accur
acy: 0.7469 - val_loss: 42.4596 - val_accuracy: 0.7687
Epoch 83/200
21/21 [==============================] - 3s 148ms/step - loss: 22.4923 - accur
acy: 0.7578 - val_loss: 33.4553 - val_accuracy: 0.7586
Epoch 84/200
21/21 [==============================] - 3s 149ms/step - loss: 17.6381 - accur
acy: 0.7631 - val_loss: 40.0883 - val_accuracy: 0.7679
Epoch 85/200
21/21 [==============================] - 3s 149ms/step - loss: 20.3053 - accur
acy: 0.7585 - val_loss: 38.0867 - val_accuracy: 0.7601
Epoch 86/200
21/21 [==============================] - 3s 148ms/step - loss: 17.2584 - accur
acy: 0.7540 - val_loss: 32.8016 - val_accuracy: 0.7671
Epoch 87/200
21/21 [==============================] - 3s 149ms/step - loss: 16.2428 - accur
acy: 0.7622 - val_loss: 49.2982 - val_accuracy: 0.7625
Epoch 88/200
21/21 [==============================] - 3s 149ms/step - loss: 21.0688 - accur
acy: 0.7545 - val_loss: 34.1650 - val_accuracy: 0.7570
Epoch 89/200
21/21 [==============================] - 3s 148ms/step - loss: 16.5791 - accur
acy: 0.7528 - val_loss: 38.7896 - val_accuracy: 0.7601
Epoch 90/200
21/21 [==============================] - 3s 149ms/step - loss: 18.3102 - accur
acy: 0.7528 - val_loss: 54.6712 - val_accuracy: 0.7609
Epoch 91/200
21/21 [==============================] - 3s 148ms/step - loss: 22.6212 - accur
acy: 0.7674 - val_loss: 34.7804 - val_accuracy: 0.7656
Epoch 92/200
21/21 [==============================] - 3s 148ms/step - loss: 17.9656 - accur
acy: 0.7558 - val_loss: 34.6408 - val_accuracy: 0.7570
Epoch 93/200
21/21 [==============================] - 3s 148ms/step - loss: 15.5754 - accur
acy: 0.7686 - val_loss: 38.6314 - val_accuracy: 0.7586
Epoch 94/200
21/21 [==============================] - 3s 149ms/step - loss: 17.8101 - accur
acy: 0.7716 - val_loss: 35.3685 - val_accuracy: 0.7632
Epoch 95/200
21/21 [==============================] - 3s 149ms/step - loss: 14.5422 - accur
acy: 0.7594 - val_loss: 40.9439 - val_accuracy: 0.7656
Epoch 96/200
21/21 [==============================] - 3s 147ms/step - loss: 15.8901 - accur
acy: 0.7707 - val_loss: 35.5840 - val_accuracy: 0.7640
Epoch 97/200
21/21 [==============================] - 3s 148ms/step - loss: 14.1911 - accur
acy: 0.7837 - val_loss: 35.0524 - val_accuracy: 0.7640
Epoch 98/200
21/21 [==============================] - 3s 149ms/step - loss: 14.7462 - accur
acy: 0.7679 - val_loss: 32.1291 - val_accuracy: 0.7570
Epoch 99/200
21/21 [==============================] - 3s 148ms/step - loss: 13.8889 - accur
acy: 0.7840 - val_loss: 35.6214 - val_accuracy: 0.7640
Epoch 100/200
21/21 [==============================] - 3s 149ms/step - loss: 13.2825 - accur
acy: 0.7649 - val_loss: 45.2029 - val_accuracy: 0.7656
```

```
Epoch 101/200
21/21 [==============================] - 3s 149ms/step - loss: 16.2774 - accur
acy: 0.7768 - val_loss: 38.4728 - val_accuracy: 0.7625
Epoch 102/200
21/21 [==============================] - 3s 149ms/step - loss: 15.1979 - accur
acy: 0.7810 - val_loss: 31.5490 - val_accuracy: 0.7695
Epoch 103/200
21/21 [==============================] - 3s 148ms/step - loss: 12.7616 - accur
acy: 0.7723 - val_loss: 32.7713 - val_accuracy: 0.7609
Epoch 104/200
21/21 [==============================] - 3s 148ms/step - loss: 13.4336 - accur
acy: 0.7708 - val_loss: 37.5372 - val_accuracy: 0.7609
Epoch 105/200
21/21 [==============================] - 3s 149ms/step - loss: 17.5106 - accur
acy: 0.7750 - val_loss: 41.9357 - val_accuracy: 0.7679
Epoch 106/200
21/21 [==============================] - 3s 148ms/step - loss: 21.6620 - accur
acy: 0.7590 - val_loss: 35.3963 - val_accuracy: 0.7570
Epoch 107/200
21/21 [==============================] - 3s 149ms/step - loss: 13.8806 - accur
acy: 0.7752 - val_loss: 31.4733 - val_accuracy: 0.7640
Epoch 108/200
21/21 [==============================] - 3s 149ms/step - loss: 12.2190 - accur
acy: 0.7886 - val_loss: 33.0452 - val_accuracy: 0.7671
Epoch 109/200
21/21 [==============================] - 3s 148ms/step - loss: 11.6924 - accur
acy: 0.7747 - val_loss: 40.5541 - val_accuracy: 0.7664
Epoch 110/200
21/21 [==============================] - 3s 148ms/step - loss: 13.9788 - accur
acy: 0.7875 - val_loss: 32.0359 - val_accuracy: 0.7593
Epoch 111/200
21/21 [==============================] - 3s 148ms/step - loss: 11.5439 - accur
acy: 0.7855 - val_loss: 32.0676 - val_accuracy: 0.7562
Epoch 112/200
21/21 [==============================] - 3s 148ms/step - loss: 12.8182 - accur
acy: 0.7855 - val_loss: 46.7910 - val_accuracy: 0.7531
Epoch 113/200
21/21 [==============================] - 3s 148ms/step - loss: 14.8484 - accur
acy: 0.7804 - val_loss: 39.2376 - val_accuracy: 0.7726
Epoch 114/200
21/21 [==============================] - 3s 148ms/step - loss: 16.4041 - accur
acy: 0.7684 - val_loss: 33.0184 - val_accuracy: 0.7617
Epoch 115/200
21/21 [==============================] - 3s 148ms/step - loss: 12.3557 - accur
acy: 0.7875 - val_loss: 30.5354 - val_accuracy: 0.7656
Epoch 116/200
21/21 [==============================] - 3s 148ms/step - loss: 11.2676 - accur
acy: 0.7938 - val_loss: 32.1316 - val_accuracy: 0.7609
Epoch 117/200
21/21 [==============================] - 3s 148ms/step - loss: 10.4786 - accur
acy: 0.7774 - val_loss: 30.3148 - val_accuracy: 0.7648
Epoch 118/200
21/21 [==============================] - 3s 148ms/step - loss: 9.6125 - accura
cy: 0.7941 - val_loss: 30.7713 - val_accuracy: 0.7640
Epoch 119/200
21/21 [==============================] - 3s 148ms/step - loss: 9.6779 - accura
cy: 0.7854 - val_loss: 32.0319 - val_accuracy: 0.7702
Epoch 120/200
21/21 [==============================] - 3s 147ms/step - loss: 11.1228 - accur
acy: 0.7891 - val_loss: 33.3846 - val_accuracy: 0.7734
```

```
Epoch 121/200
21/21 [==============================] - 3s 148ms/step - loss: 12.5378 - accur
acy: 0.7921 - val_loss: 30.9642 - val_accuracy: 0.7765
Epoch 122/200
21/21 [==============================] - 3s 148ms/step - loss: 10.6121 - accur
acy: 0.7769 - val_loss: 31.0281 - val_accuracy: 0.7671
Epoch 123/200
21/21 [==============================] - 3s 149ms/step - loss: 9.3575 - accura
cy: 0.8023 - val_loss: 44.2640 - val_accuracy: 0.7765
Epoch 124/200
21/21 [==============================] - 3s 148ms/step - loss: 12.4139 - accur
acy: 0.8071 - val_loss: 31.8570 - val_accuracy: 0.7749
Epoch 125/200
21/21 [==============================] - 3s 149ms/step - loss: 9.8766 - accura
cy: 0.7891 - val_loss: 31.8757 - val_accuracy: 0.7664
Epoch 126/200
21/21 [==============================] - 3s 149ms/step - loss: 10.0680 - accur
acy: 0.7920 - val_loss: 31.1679 - val_accuracy: 0.7741
Epoch 127/200
21/21 [==============================] - 3s 148ms/step - loss: 10.3211 - accur
acy: 0.7944 - val_loss: 30.4885 - val_accuracy: 0.7788
Epoch 128/200
21/21 [==============================] - 3s 149ms/step - loss: 8.9498 - accura
cy: 0.7940 - val_loss: 36.5441 - val_accuracy: 0.7656
Epoch 129/200
21/21 [==============================] - 3s 149ms/step - loss: 10.1383 - accur
acy: 0.7855 - val_loss: 30.9898 - val_accuracy: 0.7656
Epoch 130/200
21/21 [==============================] - 3s 154ms/step - loss: 8.7990 - accura
cy: 0.7973 - val_loss: 32.7716 - val_accuracy: 0.7702
Epoch 131/200
21/21 [==============================] - 3s 148ms/step - loss: 8.8020 - accura
cy: 0.7875 - val_loss: 38.3177 - val_accuracy: 0.7757
Epoch 132/200
21/21 [==============================] - 3s 148ms/step - loss: 12.4498 - accur
acy: 0.7968 - val_loss: 38.1873 - val_accuracy: 0.7648
Epoch 133/200
21/21 [==============================] - 3s 148ms/step - loss: 15.4009 - accur
acy: 0.7894 - val_loss: 32.6809 - val_accuracy: 0.7648
Epoch 134/200
21/21 [==============================] - 3s 155ms/step - loss: 12.5035 - accur
acy: 0.8020 - val_loss: 29.9088 - val_accuracy: 0.7718
Epoch 135/200
21/21 [==============================] - 3s 149ms/step - loss: 9.9192 - accura
cy: 0.8054 - val_loss: 37.4625 - val_accuracy: 0.7757
Epoch 136/200
21/21 [==============================] - 3s 148ms/step - loss: 11.0706 - accur
acy: 0.8014 - val_loss: 30.7776 - val_accuracy: 0.7734
Epoch 137/200
21/21 [==============================] - 3s 148ms/step - loss: 10.2467 - accur
acy: 0.7971 - val_loss: 32.0368 - val_accuracy: 0.7734
Epoch 138/200
21/21 [==============================] - 3s 149ms/step - loss: 8.9706 - accura
cy: 0.8033 - val_loss: 32.2719 - val_accuracy: 0.7687
Epoch 139/200
21/21 [==============================] - 3s 149ms/step - loss: 9.7576 - accura
cy: 0.7860 - val_loss: 29.8887 - val_accuracy: 0.7625
Epoch 140/200
21/21 [==============================] - 3s 149ms/step - loss: 7.9833 - accura
cy: 0.8025 - val_loss: 30.3887 - val_accuracy: 0.7710
```

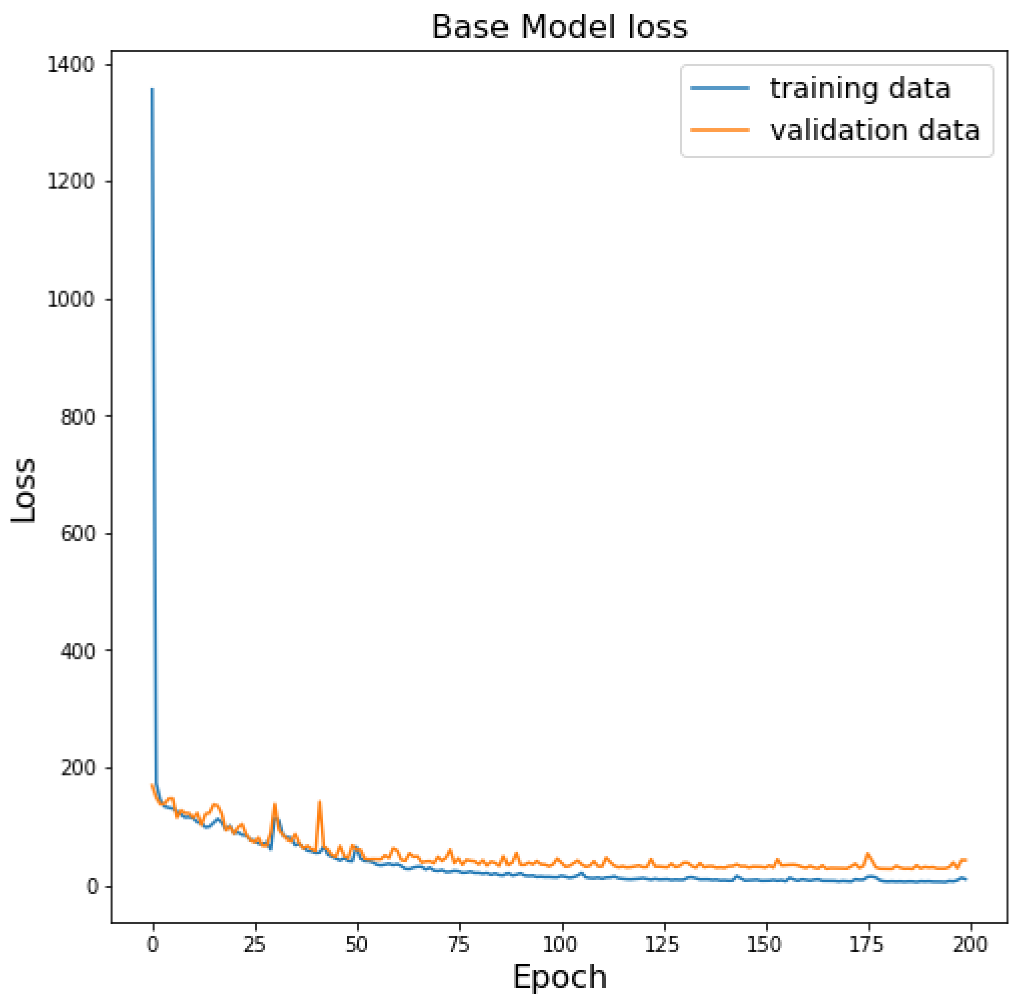
```
Epoch 141/200
21/21 [==============================] - 3s 156ms/step - loss: 8.0169 - accura
cy: 0.8009 - val_loss: 29.6066 - val_accuracy: 0.7726
Epoch 142/200
21/21 [==============================] - 3s 148ms/step - loss: 7.7768 - accura
cy: 0.8074 - val_loss: 32.1671 - val_accuracy: 0.7687
Epoch 143/200
21/21 [==============================] - 3s 149ms/step - loss: 8.0904 - accura
cy: 0.8082 - val_loss: 32.4838 - val_accuracy: 0.7726
Epoch 144/200
21/21 [==============================] - 3s 148ms/step - loss: 13.7366 - accur
acy: 0.7985 - val_loss: 35.4130 - val_accuracy: 0.7656
Epoch 145/200
21/21 [==============================] - 3s 148ms/step - loss: 12.5223 - accur
acy: 0.8101 - val_loss: 32.0769 - val_accuracy: 0.7765
Epoch 146/200
21/21 [==============================] - 3s 147ms/step - loss: 8.0312 - accura
cy: 0.8021 - val_loss: 32.3306 - val_accuracy: 0.7702
Epoch 147/200
21/21 [==============================] - 3s 148ms/step - loss: 8.2032 - accura
cy: 0.7979 - val_loss: 30.0255 - val_accuracy: 0.7812
Epoch 148/200
21/21 [==============================] - 3s 147ms/step - loss: 8.8611 - accura
cy: 0.8058 - val_loss: 31.6736 - val_accuracy: 0.7718
Epoch 149/200
21/21 [==============================] - 3s 149ms/step - loss: 9.1502 - accura
cy: 0.8017 - val_loss: 31.5741 - val_accuracy: 0.7843
Epoch 150/200
21/21 [==============================] - 3s 148ms/step - loss: 7.6370 - accura
cy: 0.8162 - val_loss: 31.3279 - val_accuracy: 0.7765
Epoch 151/200
21/21 [==============================] - 3s 149ms/step - loss: 7.4881 - accura
cy: 0.8044 - val_loss: 30.1257 - val_accuracy: 0.7734
Epoch 152/200
21/21 [==============================] - 3s 148ms/step - loss: 7.9366 - accura
cy: 0.8120 - val_loss: 33.4426 - val_accuracy: 0.7765
Epoch 153/200
21/21 [==============================] - 3s 149ms/step - loss: 8.9016 - accura
cy: 0.8106 - val_loss: 30.1685 - val_accuracy: 0.7749
Epoch 154/200
21/21 [==============================] - 3s 149ms/step - loss: 7.6138 - accura
cy: 0.8088 - val_loss: 44.0003 - val_accuracy: 0.7843
Epoch 155/200
21/21 [==============================] - 3s 154ms/step - loss: 10.2040 - accur
acy: 0.8052 - val_loss: 33.4994 - val_accuracy: 0.7718
Epoch 156/200
21/21 [==============================] - 3s 149ms/step - loss: 7.0969 - accura
cy: 0.8139 - val_loss: 34.1594 - val_accuracy: 0.7734
Epoch 157/200
21/21 [==============================] - 3s 149ms/step - loss: 13.0869 - accur
acy: 0.8116 - val_loss: 34.6451 - val_accuracy: 0.7687
Epoch 158/200
21/21 [==============================] - 3s 148ms/step - loss: 10.0705 - accur
acy: 0.7982 - val_loss: 34.9360 - val_accuracy: 0.7765
Epoch 159/200
21/21 [==============================] - 3s 149ms/step - loss: 8.0825 - accura
cy: 0.8124 - val_loss: 33.6247 - val_accuracy: 0.7702
Epoch 160/200
21/21 [==============================] - 3s 149ms/step - loss: 10.4545 - accur
acy: 0.8108 - val_loss: 31.1004 - val_accuracy: 0.7804
```

```
Epoch 161/200
21/21 [==============================] - 3s 150ms/step - loss: 9.3018 - accura
cy: 0.8117 - val_loss: 29.4437 - val_accuracy: 0.7788
Epoch 162/200
21/21 [==============================] - 3s 149ms/step - loss: 7.9394 - accura
cy: 0.8114 - val_loss: 32.6672 - val_accuracy: 0.7765
Epoch 163/200
21/21 [==============================] - 3s 148ms/step - loss: 10.6909 - accur
acy: 0.8032 - val_loss: 29.8098 - val_accuracy: 0.7788
Epoch 164/200
21/21 [==============================] - 3s 149ms/step - loss: 9.6178 - accura
cy: 0.8221 - val_loss: 28.5498 - val_accuracy: 0.7734
Epoch 165/200
21/21 [==============================] - 3s 148ms/step - loss: 7.5821 - accura
cy: 0.8147 - val_loss: 33.4852 - val_accuracy: 0.7765
Epoch 166/200
21/21 [==============================] - 3s 149ms/step - loss: 7.8393 - accura
cy: 0.8141 - val_loss: 27.8979 - val_accuracy: 0.7804
Epoch 167/200
21/21 [==============================] - 3s 149ms/step - loss: 6.6579 - accura
cy: 0.8089 - val_loss: 29.0397 - val_accuracy: 0.7819
Epoch 168/200
21/21 [==============================] - 3s 149ms/step - loss: 7.4536 - accura
cy: 0.8082 - val_loss: 29.0372 - val_accuracy: 0.7757
Epoch 169/200
21/21 [==============================] - 3s 148ms/step - loss: 6.4152 - accura
cy: 0.8125 - val_loss: 29.0764 - val_accuracy: 0.7757
Epoch 170/200
21/21 [==============================] - 3s 149ms/step - loss: 7.3021 - accura
cy: 0.8137 - val_loss: 28.6999 - val_accuracy: 0.7788
Epoch 171/200
21/21 [==============================] - 3s 148ms/step - loss: 6.7724 - accura
cy: 0.8166 - val_loss: 28.3183 - val_accuracy: 0.7819
Epoch 172/200
21/21 [==============================] - 3s 148ms/step - loss: 6.1268 - accura
cy: 0.8166 - val_loss: 32.0771 - val_accuracy: 0.7726
Epoch 173/200
21/21 [==============================] - 3s 149ms/step - loss: 10.6386 - accur
acy: 0.8192 - val_loss: 36.5264 - val_accuracy: 0.7702
Epoch 174/200
21/21 [==============================] - 3s 148ms/step - loss: 9.5481 - accura
cy: 0.8175 - val_loss: 29.2224 - val_accuracy: 0.7734
Epoch 175/200
21/21 [==============================] - 3s 149ms/step - loss: 9.3800 - accura
cy: 0.8173 - val_loss: 33.0013 - val_accuracy: 0.7773
Epoch 176/200
21/21 [==============================] - 3s 147ms/step - loss: 14.7732 - accur
acy: 0.8103 - val_loss: 53.8838 - val_accuracy: 0.7812
Epoch 177/200
21/21 [==============================] - 3s 149ms/step - loss: 17.9078 - accur
acy: 0.8111 - val_loss: 43.1395 - val_accuracy: 0.7780
Epoch 178/200
21/21 [==============================] - 3s 149ms/step - loss: 14.1981 - accur
acy: 0.8101 - val_loss: 30.4384 - val_accuracy: 0.7702
Epoch 179/200
21/21 [==============================] - 3s 148ms/step - loss: 8.6409 - accura
cy: 0.8126 - val_loss: 28.2008 - val_accuracy: 0.7827
Epoch 180/200
21/21 [==============================] - 3s 150ms/step - loss: 6.7538 - accura
cy: 0.8227 - val_loss: 28.0797 - val_accuracy: 0.7819
```

```
Epoch 181/200
21/21 [==============================] - 3s 149ms/step - loss: 5.9851 - accura
cy: 0.8302 - val_loss: 27.6790 - val_accuracy: 0.7835
Epoch 182/200
21/21 [==============================] - 3s 149ms/step - loss: 6.1750 - accura
cy: 0.8210 - val_loss: 27.7512 - val_accuracy: 0.7741
Epoch 183/200
21/21 [==============================] - 3s 149ms/step - loss: 6.0448 - accura
cy: 0.8227 - val_loss: 32.8339 - val_accuracy: 0.7710
Epoch 184/200
21/21 [==============================] - 3s 149ms/step - loss: 6.6809 - accura
cy: 0.8248 - val_loss: 29.6798 - val_accuracy: 0.7812
Epoch 185/200
21/21 [==============================] - 3s 154ms/step - loss: 6.0104 - accura
cy: 0.8323 - val_loss: 28.0861 - val_accuracy: 0.7788
Epoch 186/200
21/21 [==============================] - 3s 150ms/step - loss: 6.1436 - accura
cy: 0.8332 - val_loss: 28.5285 - val_accuracy: 0.7710
Epoch 187/200
21/21 [==============================] - 3s 148ms/step - loss: 6.5541 - accura
cy: 0.8221 - val_loss: 27.6350 - val_accuracy: 0.7796
Epoch 188/200
21/21 [==============================] - 3s 149ms/step - loss: 5.3638 - accura
cy: 0.8308 - val_loss: 33.7586 - val_accuracy: 0.7850
Epoch 189/200
21/21 [==============================] - 3s 149ms/step - loss: 7.0978 - accura
cy: 0.8295 - val_loss: 28.4404 - val_accuracy: 0.7757
Epoch 190/200
21/21 [==============================] - 3s 148ms/step - loss: 6.5188 - accura
cy: 0.8207 - val_loss: 31.2016 - val_accuracy: 0.7741
Epoch 191/200
21/21 [==============================] - 3s 149ms/step - loss: 6.6684 - accura
cy: 0.8209 - val_loss: 29.9703 - val_accuracy: 0.7804
Epoch 192/200
21/21 [==============================] - 3s 150ms/step - loss: 5.8475 - accura
cy: 0.8389 - val_loss: 30.7573 - val_accuracy: 0.7812
Epoch 193/200
21/21 [==============================] - 3s 148ms/step - loss: 5.8618 - accura
cy: 0.8466 - val_loss: 28.5384 - val_accuracy: 0.7741
Epoch 194/200
21/21 [==============================] - 3s 149ms/step - loss: 5.6436 - accura
cy: 0.8366 - val_loss: 28.3016 - val_accuracy: 0.7749
Epoch 195/200
21/21 [==============================] - 3s 150ms/step - loss: 5.0574 - accura
cy: 0.8277 - val_loss: 28.4410 - val_accuracy: 0.7741
Epoch 196/200
21/21 [==============================] - 3s 149ms/step - loss: 6.8590 - accura
cy: 0.8326 - val_loss: 30.6511 - val_accuracy: 0.7827
Epoch 197/200
21/21 [==============================] - 3s 148ms/step - loss: 7.0817 - accura
cy: 0.8350 - val_loss: 38.7126 - val_accuracy: 0.7780
Epoch 198/200
21/21 [==============================] - 3s 155ms/step - loss: 9.0134 - accura
cy: 0.8287 - val_loss: 28.6266 - val_accuracy: 0.7765
Epoch 199/200
21/21 [==============================] - 3s 147ms/step - loss: 12.0456 - accur
acy: 0.8251 - val_loss: 42.7015 - val_accuracy: 0.7835
Epoch 200/200
21/21 [==============================] - 3s 149ms/step - loss: 12.4244 - accur
acy: 0.8143 - val_loss: 42.5877 - val_accuracy: 0.7812
```

In [76]:
```
plt.figure(figsize=(8,8))
plt.plot(history_augment.history['loss'])
plt.plot(history_augment.history['val_loss'])
plt.title('Base Model loss',fontsize=16)
plt.ylabel('Loss',fontsize=16)
plt.xlabel('Epoch',fontsize=16)
plt.legend(['training data', 'validation data'], loc='upper right',fontsize=14)
```

Out[76]:
```
<matplotlib.legend.Legend at 0x7f1ee1b4b150>
```



Transfer Learning

Transfer learning uses pre-trained models and corresponding weights to augment our base model. This is useful:

- To overcome the small train data size we have even after augmentation.
- Provides better accuracy as the model has been trained already by several million images
  We have considered MobileNet as it suits our use-case and also due to it's light-weight nature. It also performed well compared to other models such as VGG16, InceptionV3 etc. based on our testing.

In [85]:
```
import tensorflow as tf
```

```
from tensorflow.python.keras import Sequential
from tensorflow.keras import layers, optimizers, applications, callbacks
from tensorflow.keras.applications import DenseNet121
from tensorflow.keras.applications.resnet50 import ResNet50
from tensorflow.keras.layers import *
from tensorflow.keras.models import Model, load_model
from tensorflow.keras.initializers import glorot_uniform
from tensorflow.keras.utils import plot_model
from tensorflow.keras.callbacks import ReduceLROnPlateau, EarlyStopping, ModelC
from IPython.display import display
from tensorflow.keras import backend as K
import tensorflow.keras.preprocessing.image as tf_image
from sklearn.model_selection import train_test_split

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import cv2
```

In [98]:

```
model_transfer_learning = Sequential()

# Use MobileNet pre-trainined model due to it's lightweight computations and su
pretrained_model = applications.MobileNet(input_shape=(96, 96, 3), include_top=
pretrained_model.trainable = True

# Define input layers
model_transfer_learning.add(layers.Convolution2D(3, (1, 1), padding='same', use
model_transfer_learning.add(layers.LeakyReLU(alpha = 0.1))

# Introduce pre_trainined model as a layer inbetween by dropping it's existing
model_transfer_learning.add(pretrained_model)
model_transfer_learning.add(layers.GlobalAveragePooling2D())
model_transfer_learning.add(layers.Dropout(0.3))
model_transfer_learning.add(layers.Dense(30,activation='elu'))

model_transfer_learning.summary()

model_transfer_learning.compile(optimizer='adam',
              loss='mean_squared_error',
              metrics=['accuracy'])
```

```
WARNING:tensorflow:`input_shape` is undefined or non-square, or `rows` is not
in [128, 160, 192, 224]. Weights for input shape (224, 224) will be loaded as
the default.
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_4 (Conv2D)            (None, 96, 96, 3)         3
_________________________________________________________________
leaky_re_lu_4 (LeakyReLU)    (None, 96, 96, 3)         0
_________________________________________________________________
mobilenet_1.00_224 (Function (None, 3, 3, 1024)        3228864
_________________________________________________________________
global_average_pooling2d_4 ( (None, 1024)              0
_________________________________________________________________
dropout_4 (Dropout)          (None, 1024)              0
_________________________________________________________________
dense_4 (Dense)              (None, 30)                30750
=================================================================
Total params: 3,259,617
Trainable params: 3,237,729
Non-trainable params: 21,888
_________________________________________________________________
```

In [99]:
```
history_transfer_learning_augment = model_transfer_learning.fit(X_train,y_trair
```

```
Epoch 1/250
21/21 [==============================] - 7s 217ms/step - loss: 1828.1873 - acc
uracy: 0.1129 - val_loss: 892.9673 - val_accuracy: 0.1269
Epoch 2/250
21/21 [==============================] - 4s 192ms/step - loss: 570.9426 - accu
racy: 0.2401 - val_loss: 594.0733 - val_accuracy: 0.2453
Epoch 3/250
21/21 [==============================] - 4s 196ms/step - loss: 222.9855 - accu
racy: 0.4077 - val_loss: 391.2330 - val_accuracy: 0.2944
Epoch 4/250
21/21 [==============================] - 4s 199ms/step - loss: 111.0810 - accu
racy: 0.4745 - val_loss: 186.7026 - val_accuracy: 0.4673
Epoch 5/250
21/21 [==============================] - 4s 196ms/step - loss: 52.3300 - accur
acy: 0.5243 - val_loss: 69.1961 - val_accuracy: 0.6752
Epoch 6/250
21/21 [==============================] - 4s 192ms/step - loss: 26.1767 - accur
acy: 0.6081 - val_loss: 44.6618 - val_accuracy: 0.6783
Epoch 7/250
21/21 [==============================] - 4s 198ms/step - loss: 23.6152 - accur
acy: 0.6301 - val_loss: 35.7036 - val_accuracy: 0.6729
Epoch 8/250
21/21 [==============================] - 4s 195ms/step - loss: 23.7307 - accur
acy: 0.6427 - val_loss: 36.3608 - val_accuracy: 0.7002
Epoch 9/250
21/21 [==============================] - 4s 192ms/step - loss: 22.8926 - accur
acy: 0.6361 - val_loss: 24.6952 - val_accuracy: 0.6760
Epoch 10/250
21/21 [==============================] - 4s 194ms/step - loss: 21.3373 - accur
acy: 0.6375 - val_loss: 14.5246 - val_accuracy: 0.7017
Epoch 11/250
21/21 [==============================] - 4s 196ms/step - loss: 20.2263 - accur
acy: 0.6338 - val_loss: 10.6737 - val_accuracy: 0.7017
Epoch 12/250
21/21 [==============================] - 4s 198ms/step - loss: 20.0453 - accur
acy: 0.6474 - val_loss: 27.6694 - val_accuracy: 0.7033
Epoch 13/250
21/21 [==============================] - 4s 197ms/step - loss: 19.9834 - accur
acy: 0.6534 - val_loss: 29.8590 - val_accuracy: 0.7033
Epoch 14/250
21/21 [==============================] - 4s 196ms/step - loss: 19.2074 - accur
acy: 0.6536 - val_loss: 29.0000 - val_accuracy: 0.7033
Epoch 15/250
21/21 [==============================] - 4s 195ms/step - loss: 19.4962 - accur
acy: 0.6579 - val_loss: 36.5536 - val_accuracy: 0.7033
Epoch 16/250
21/21 [==============================] - 4s 199ms/step - loss: 19.3883 - accur
acy: 0.6519 - val_loss: 24.6329 - val_accuracy: 0.7033
Epoch 17/250
21/21 [==============================] - 4s 198ms/step - loss: 17.6359 - accur
acy: 0.6468 - val_loss: 18.7979 - val_accuracy: 0.7033
Epoch 18/250
21/21 [==============================] - 4s 195ms/step - loss: 16.9594 - accur
acy: 0.6505 - val_loss: 17.2143 - val_accuracy: 0.7033
Epoch 19/250
21/21 [==============================] - 4s 198ms/step - loss: 16.3199 - accur
acy: 0.6581 - val_loss: 13.0957 - val_accuracy: 0.7033
Epoch 20/250
21/21 [==============================] - 4s 195ms/step - loss: 16.3373 - accur
acy: 0.6544 - val_loss: 13.3185 - val_accuracy: 0.7033
```

```
Epoch 21/250
21/21 [==============================] - 4s 195ms/step - loss: 15.6942 - accur
acy: 0.6579 - val_loss: 22.8801 - val_accuracy: 0.7033
Epoch 22/250
21/21 [==============================] - 4s 196ms/step - loss: 15.5860 - accur
acy: 0.6678 - val_loss: 13.6133 - val_accuracy: 0.7033
Epoch 23/250
21/21 [==============================] - 4s 198ms/step - loss: 15.3138 - accur
acy: 0.6690 - val_loss: 9.1061 - val_accuracy: 0.7064
Epoch 24/250
21/21 [==============================] - 4s 195ms/step - loss: 15.6777 - accur
acy: 0.6597 - val_loss: 10.1769 - val_accuracy: 0.7118
Epoch 25/250
21/21 [==============================] - 4s 195ms/step - loss: 15.4017 - accur
acy: 0.6684 - val_loss: 8.6312 - val_accuracy: 0.7087
Epoch 26/250
21/21 [==============================] - 4s 192ms/step - loss: 15.0528 - accur
acy: 0.6735 - val_loss: 10.4565 - val_accuracy: 0.7064
Epoch 27/250
21/21 [==============================] - 4s 196ms/step - loss: 14.7547 - accur
acy: 0.6663 - val_loss: 9.5511 - val_accuracy: 0.7181
Epoch 28/250
21/21 [==============================] - 4s 196ms/step - loss: 15.1999 - accur
acy: 0.6676 - val_loss: 8.6386 - val_accuracy: 0.7220
Epoch 29/250
21/21 [==============================] - 4s 197ms/step - loss: 14.5286 - accur
acy: 0.6762 - val_loss: 11.9285 - val_accuracy: 0.7165
Epoch 30/250
21/21 [==============================] - 4s 193ms/step - loss: 16.3826 - accur
acy: 0.6682 - val_loss: 18.8681 - val_accuracy: 0.7072
Epoch 31/250
21/21 [==============================] - 4s 194ms/step - loss: 18.3637 - accur
acy: 0.6791 - val_loss: 18.3253 - val_accuracy: 0.7033
Epoch 32/250
21/21 [==============================] - 4s 195ms/step - loss: 16.4504 - accur
acy: 0.6680 - val_loss: 13.3732 - val_accuracy: 0.7048
Epoch 33/250
21/21 [==============================] - 4s 190ms/step - loss: 15.2415 - accur
acy: 0.6649 - val_loss: 11.7405 - val_accuracy: 0.7056
Epoch 34/250
21/21 [==============================] - 4s 198ms/step - loss: 14.2405 - accur
acy: 0.6760 - val_loss: 14.2569 - val_accuracy: 0.7056
Epoch 35/250
21/21 [==============================] - 4s 192ms/step - loss: 13.3025 - accur
acy: 0.6674 - val_loss: 12.2522 - val_accuracy: 0.7064
Epoch 36/250
21/21 [==============================] - 4s 193ms/step - loss: 13.0637 - accur
acy: 0.6713 - val_loss: 11.1964 - val_accuracy: 0.7056
Epoch 37/250
21/21 [==============================] - 4s 195ms/step - loss: 12.6842 - accur
acy: 0.6727 - val_loss: 8.5540 - val_accuracy: 0.7095
Epoch 38/250
21/21 [==============================] - 4s 194ms/step - loss: 12.7595 - accur
acy: 0.6768 - val_loss: 9.8671 - val_accuracy: 0.7103
Epoch 39/250
21/21 [==============================] - 4s 196ms/step - loss: 12.3932 - accur
acy: 0.6698 - val_loss: 7.6134 - val_accuracy: 0.7103
Epoch 40/250
21/21 [==============================] - 4s 193ms/step - loss: 12.1157 - accur
acy: 0.6902 - val_loss: 8.7867 - val_accuracy: 0.7126
```

```
Epoch 41/250
21/21 [==============================] - 4s 193ms/step - loss: 11.5075 - accur
acy: 0.6819 - val_loss: 7.6487 - val_accuracy: 0.7009
Epoch 42/250
21/21 [==============================] - 4s 196ms/step - loss: 12.1285 - accur
acy: 0.6982 - val_loss: 10.8747 - val_accuracy: 0.7009
Epoch 43/250
21/21 [==============================] - 4s 194ms/step - loss: 11.2369 - accur
acy: 0.6900 - val_loss: 11.4789 - val_accuracy: 0.7220
Epoch 44/250
21/21 [==============================] - 4s 200ms/step - loss: 10.5561 - accur
acy: 0.6908 - val_loss: 10.2375 - val_accuracy: 0.7274
Epoch 45/250
21/21 [==============================] - 4s 195ms/step - loss: 10.7042 - accur
acy: 0.6963 - val_loss: 9.4854 - val_accuracy: 0.7150
Epoch 46/250
21/21 [==============================] - 4s 196ms/step - loss: 9.9928 - accura
cy: 0.7062 - val_loss: 8.4953 - val_accuracy: 0.7118
Epoch 47/250
21/21 [==============================] - 4s 195ms/step - loss: 9.0752 - accura
cy: 0.7035 - val_loss: 10.5346 - val_accuracy: 0.7259
Epoch 48/250
21/21 [==============================] - 4s 194ms/step - loss: 8.9243 - accura
cy: 0.7023 - val_loss: 12.7516 - val_accuracy: 0.7227
Epoch 49/250
21/21 [==============================] - 4s 195ms/step - loss: 8.0987 - accura
cy: 0.7140 - val_loss: 11.1421 - val_accuracy: 0.7274
Epoch 50/250
21/21 [==============================] - 4s 196ms/step - loss: 7.6872 - accura
cy: 0.7136 - val_loss: 8.2459 - val_accuracy: 0.7329
Epoch 51/250
21/21 [==============================] - 4s 195ms/step - loss: 7.4981 - accura
cy: 0.7233 - val_loss: 9.7465 - val_accuracy: 0.7344
Epoch 52/250
21/21 [==============================] - 4s 194ms/step - loss: 7.2095 - accura
cy: 0.7294 - val_loss: 6.7235 - val_accuracy: 0.7422
Epoch 53/250
21/21 [==============================] - 4s 193ms/step - loss: 7.0817 - accura
cy: 0.7305 - val_loss: 8.7139 - val_accuracy: 0.7438
Epoch 54/250
21/21 [==============================] - 4s 195ms/step - loss: 6.6973 - accura
cy: 0.7311 - val_loss: 6.9726 - val_accuracy: 0.7391
Epoch 55/250
21/21 [==============================] - 4s 195ms/step - loss: 6.4060 - accura
cy: 0.7309 - val_loss: 8.1225 - val_accuracy: 0.7469
Epoch 56/250
21/21 [==============================] - 4s 197ms/step - loss: 6.1352 - accura
cy: 0.7344 - val_loss: 9.4980 - val_accuracy: 0.7539
Epoch 57/250
21/21 [==============================] - 4s 196ms/step - loss: 5.9273 - accura
cy: 0.7397 - val_loss: 9.3200 - val_accuracy: 0.7391
Epoch 58/250
21/21 [==============================] - 4s 196ms/step - loss: 5.8391 - accura
cy: 0.7422 - val_loss: 7.0134 - val_accuracy: 0.7547
Epoch 59/250
21/21 [==============================] - 4s 195ms/step - loss: 5.6453 - accura
cy: 0.7407 - val_loss: 7.6353 - val_accuracy: 0.7461
Epoch 60/250
21/21 [==============================] - 4s 195ms/step - loss: 5.4960 - accura
cy: 0.7523 - val_loss: 5.2954 - val_accuracy: 0.7593
```

```
Epoch 61/250
21/21 [==============================] - 4s 197ms/step - loss: 5.2027 - accura
cy: 0.7465 - val_loss: 5.6118 - val_accuracy: 0.7656
Epoch 62/250
21/21 [==============================] - 4s 195ms/step - loss: 5.0118 - accura
cy: 0.7498 - val_loss: 3.4441 - val_accuracy: 0.7671
Epoch 63/250
21/21 [==============================] - 4s 194ms/step - loss: 4.8438 - accura
cy: 0.7539 - val_loss: 3.3442 - val_accuracy: 0.7734
Epoch 64/250
21/21 [==============================] - 4s 194ms/step - loss: 4.8582 - accura
cy: 0.7545 - val_loss: 5.7208 - val_accuracy: 0.7757
Epoch 65/250
21/21 [==============================] - 4s 196ms/step - loss: 5.3463 - accura
cy: 0.7576 - val_loss: 7.7015 - val_accuracy: 0.7422
Epoch 66/250
21/21 [==============================] - 4s 193ms/step - loss: 4.7671 - accura
cy: 0.7555 - val_loss: 8.4707 - val_accuracy: 0.7523
Epoch 67/250
21/21 [==============================] - 4s 192ms/step - loss: 4.4830 - accura
cy: 0.7701 - val_loss: 3.3629 - val_accuracy: 0.7625
Epoch 68/250
21/21 [==============================] - 4s 194ms/step - loss: 4.5508 - accura
cy: 0.7601 - val_loss: 5.9733 - val_accuracy: 0.7640
Epoch 69/250
21/21 [==============================] - 4s 193ms/step - loss: 4.3176 - accura
cy: 0.7802 - val_loss: 4.5173 - val_accuracy: 0.7757
Epoch 70/250
21/21 [==============================] - 4s 191ms/step - loss: 4.2575 - accura
cy: 0.7771 - val_loss: 3.7200 - val_accuracy: 0.7796
Epoch 71/250
21/21 [==============================] - 4s 193ms/step - loss: 4.2746 - accura
cy: 0.7749 - val_loss: 3.7539 - val_accuracy: 0.8053
Epoch 72/250
21/21 [==============================] - 4s 194ms/step - loss: 4.0976 - accura
cy: 0.7845 - val_loss: 3.7128 - val_accuracy: 0.8006
Epoch 73/250
21/21 [==============================] - 4s 196ms/step - loss: 4.1376 - accura
cy: 0.7812 - val_loss: 3.8842 - val_accuracy: 0.8146
Epoch 74/250
21/21 [==============================] - 4s 193ms/step - loss: 4.2668 - accura
cy: 0.7905 - val_loss: 2.8278 - val_accuracy: 0.8006
Epoch 75/250
21/21 [==============================] - 4s 192ms/step - loss: 4.1243 - accura
cy: 0.7868 - val_loss: 3.1967 - val_accuracy: 0.8107
Epoch 76/250
21/21 [==============================] - 4s 197ms/step - loss: 4.1541 - accura
cy: 0.7903 - val_loss: 4.2456 - val_accuracy: 0.8162
Epoch 77/250
21/21 [==============================] - 4s 194ms/step - loss: 4.0625 - accura
cy: 0.8000 - val_loss: 3.9065 - val_accuracy: 0.8170
Epoch 78/250
21/21 [==============================] - 4s 201ms/step - loss: 3.9940 - accura
cy: 0.8004 - val_loss: 5.5183 - val_accuracy: 0.8248
Epoch 79/250
21/21 [==============================] - 4s 196ms/step - loss: 4.0454 - accura
cy: 0.8035 - val_loss: 6.0485 - val_accuracy: 0.7905
Epoch 80/250
21/21 [==============================] - 4s 198ms/step - loss: 3.9718 - accura
cy: 0.8096 - val_loss: 3.1002 - val_accuracy: 0.7967
```

```
Epoch 81/250
21/21 [==============================] - 4s 194ms/step - loss: 3.7534 - accura
cy: 0.8024 - val_loss: 2.6807 - val_accuracy: 0.8162
Epoch 82/250
21/21 [==============================] - 4s 197ms/step - loss: 3.9037 - accura
cy: 0.8043 - val_loss: 4.2753 - val_accuracy: 0.8115
Epoch 83/250
21/21 [==============================] - 4s 193ms/step - loss: 3.6999 - accura
cy: 0.8080 - val_loss: 2.7708 - val_accuracy: 0.8240
Epoch 84/250
21/21 [==============================] - 4s 197ms/step - loss: 3.6734 - accura
cy: 0.8141 - val_loss: 2.8426 - val_accuracy: 0.8302
Epoch 85/250
21/21 [==============================] - 4s 196ms/step - loss: 3.7235 - accura
cy: 0.8063 - val_loss: 3.5200 - val_accuracy: 0.8248
Epoch 86/250
21/21 [==============================] - 4s 195ms/step - loss: 3.6173 - accura
cy: 0.8181 - val_loss: 2.6603 - val_accuracy: 0.8294
Epoch 87/250
21/21 [==============================] - 4s 192ms/step - loss: 3.5464 - accura
cy: 0.8141 - val_loss: 2.5881 - val_accuracy: 0.8326
Epoch 88/250
21/21 [==============================] - 4s 190ms/step - loss: 3.6204 - accura
cy: 0.8139 - val_loss: 2.7652 - val_accuracy: 0.7780
Epoch 89/250
21/21 [==============================] - 4s 196ms/step - loss: 3.6377 - accura
cy: 0.8141 - val_loss: 2.6079 - val_accuracy: 0.8022
Epoch 90/250
21/21 [==============================] - 4s 199ms/step - loss: 3.4659 - accura
cy: 0.8115 - val_loss: 2.3958 - val_accuracy: 0.8419
Epoch 91/250
21/21 [==============================] - 4s 193ms/step - loss: 3.4908 - accura
cy: 0.8187 - val_loss: 2.5040 - val_accuracy: 0.8388
Epoch 92/250
21/21 [==============================] - 4s 195ms/step - loss: 3.4145 - accura
cy: 0.8254 - val_loss: 2.4326 - val_accuracy: 0.8481
Epoch 93/250
21/21 [==============================] - 4s 196ms/step - loss: 3.4725 - accura
cy: 0.8166 - val_loss: 2.5762 - val_accuracy: 0.8427
Epoch 94/250
21/21 [==============================] - 4s 198ms/step - loss: 3.5214 - accura
cy: 0.8230 - val_loss: 2.7416 - val_accuracy: 0.8419
Epoch 95/250
21/21 [==============================] - 4s 196ms/step - loss: 3.4160 - accura
cy: 0.8259 - val_loss: 3.1408 - val_accuracy: 0.8318
Epoch 96/250
21/21 [==============================] - 4s 196ms/step - loss: 3.3543 - accura
cy: 0.8250 - val_loss: 3.0752 - val_accuracy: 0.8263
Epoch 97/250
21/21 [==============================] - 4s 195ms/step - loss: 3.4129 - accura
cy: 0.8304 - val_loss: 4.8643 - val_accuracy: 0.8349
Epoch 98/250
21/21 [==============================] - 4s 194ms/step - loss: 3.3947 - accura
cy: 0.8267 - val_loss: 3.5054 - val_accuracy: 0.8294
Epoch 99/250
21/21 [==============================] - 4s 194ms/step - loss: 3.2181 - accura
cy: 0.8248 - val_loss: 2.9405 - val_accuracy: 0.8302
Epoch 100/250
21/21 [==============================] - 4s 193ms/step - loss: 3.1378 - accura
cy: 0.8326 - val_loss: 2.1656 - val_accuracy: 0.8357
```

```
Epoch 101/250
21/21 [==============================] - 4s 190ms/step - loss: 3.1903 - accura
cy: 0.8316 - val_loss: 2.8281 - val_accuracy: 0.8178
Epoch 102/250
21/21 [==============================] - 4s 198ms/step - loss: 3.2216 - accura
cy: 0.8298 - val_loss: 2.3707 - val_accuracy: 0.8419
Epoch 103/250
21/21 [==============================] - 4s 196ms/step - loss: 3.1841 - accura
cy: 0.8353 - val_loss: 2.4503 - val_accuracy: 0.8294
Epoch 104/250
21/21 [==============================] - 4s 197ms/step - loss: 3.2265 - accura
cy: 0.8400 - val_loss: 2.3659 - val_accuracy: 0.8302
Epoch 105/250
21/21 [==============================] - 4s 197ms/step - loss: 3.1952 - accura
cy: 0.8322 - val_loss: 2.3378 - val_accuracy: 0.8357
Epoch 106/250
21/21 [==============================] - 4s 198ms/step - loss: 3.0805 - accura
cy: 0.8318 - val_loss: 2.3641 - val_accuracy: 0.8279
Epoch 107/250
21/21 [==============================] - 4s 195ms/step - loss: 3.0838 - accura
cy: 0.8388 - val_loss: 2.4636 - val_accuracy: 0.8030
Epoch 108/250
21/21 [==============================] - 4s 194ms/step - loss: 3.0550 - accura
cy: 0.8333 - val_loss: 2.3111 - val_accuracy: 0.8341
Epoch 109/250
21/21 [==============================] - 4s 195ms/step - loss: 3.1794 - accura
cy: 0.8384 - val_loss: 2.3106 - val_accuracy: 0.8419
Epoch 110/250
21/21 [==============================] - 4s 193ms/step - loss: 3.0917 - accura
cy: 0.8390 - val_loss: 2.6187 - val_accuracy: 0.8045
Epoch 111/250
21/21 [==============================] - 4s 196ms/step - loss: 3.0958 - accura
cy: 0.8327 - val_loss: 3.4481 - val_accuracy: 0.8224
Epoch 112/250
21/21 [==============================] - 4s 192ms/step - loss: 3.1190 - accura
cy: 0.8386 - val_loss: 2.3054 - val_accuracy: 0.8006
Epoch 113/250
21/21 [==============================] - 4s 199ms/step - loss: 2.9804 - accura
cy: 0.8444 - val_loss: 2.6496 - val_accuracy: 0.8193
Epoch 114/250
21/21 [==============================] - 4s 192ms/step - loss: 3.0624 - accura
cy: 0.8413 - val_loss: 2.0676 - val_accuracy: 0.8123
Epoch 115/250
21/21 [==============================] - 4s 193ms/step - loss: 3.0664 - accura
cy: 0.8353 - val_loss: 1.9608 - val_accuracy: 0.8318
Epoch 116/250
21/21 [==============================] - 4s 191ms/step - loss: 3.0724 - accura
cy: 0.8396 - val_loss: 2.3434 - val_accuracy: 0.8458
Epoch 117/250
21/21 [==============================] - 4s 193ms/step - loss: 3.0512 - accura
cy: 0.8438 - val_loss: 2.2655 - val_accuracy: 0.8154
Epoch 118/250
21/21 [==============================] - 4s 198ms/step - loss: 2.9898 - accura
cy: 0.8374 - val_loss: 2.1689 - val_accuracy: 0.8364
Epoch 119/250
21/21 [==============================] - 4s 194ms/step - loss: 3.0174 - accura
cy: 0.8456 - val_loss: 1.9892 - val_accuracy: 0.8100
Epoch 120/250
21/21 [==============================] - 4s 198ms/step - loss: 3.0468 - accura
cy: 0.8400 - val_loss: 2.2472 - val_accuracy: 0.8427
```

```
Epoch 121/250
21/21 [==============================] - 4s 194ms/step - loss: 3.0294 - accura
cy: 0.8400 - val_loss: 2.3973 - val_accuracy: 0.8185
Epoch 122/250
21/21 [==============================] - 4s 195ms/step - loss: 2.8889 - accura
cy: 0.8489 - val_loss: 2.4060 - val_accuracy: 0.8497
Epoch 123/250
21/21 [==============================] - 4s 198ms/step - loss: 2.9463 - accura
cy: 0.8380 - val_loss: 1.9525 - val_accuracy: 0.8419
Epoch 124/250
21/21 [==============================] - 4s 204ms/step - loss: 2.9663 - accura
cy: 0.8376 - val_loss: 1.8586 - val_accuracy: 0.8458
Epoch 125/250
21/21 [==============================] - 4s 194ms/step - loss: 2.9199 - accura
cy: 0.8460 - val_loss: 2.1744 - val_accuracy: 0.8458
Epoch 126/250
21/21 [==============================] - 4s 194ms/step - loss: 2.8616 - accura
cy: 0.8470 - val_loss: 2.1235 - val_accuracy: 0.8474
Epoch 127/250
21/21 [==============================] - 4s 193ms/step - loss: 2.9510 - accura
cy: 0.8427 - val_loss: 1.7804 - val_accuracy: 0.8497
Epoch 128/250
21/21 [==============================] - 4s 193ms/step - loss: 2.7941 - accura
cy: 0.8505 - val_loss: 2.4365 - val_accuracy: 0.8326
Epoch 129/250
21/21 [==============================] - 4s 195ms/step - loss: 2.8826 - accura
cy: 0.8345 - val_loss: 2.2957 - val_accuracy: 0.8450
Epoch 130/250
21/21 [==============================] - 4s 202ms/step - loss: 2.8555 - accura
cy: 0.8458 - val_loss: 2.7638 - val_accuracy: 0.8450
Epoch 131/250
21/21 [==============================] - 4s 197ms/step - loss: 2.9226 - accura
cy: 0.8522 - val_loss: 2.8523 - val_accuracy: 0.8357
Epoch 132/250
21/21 [==============================] - 4s 197ms/step - loss: 2.8669 - accura
cy: 0.8509 - val_loss: 2.0704 - val_accuracy: 0.8380
Epoch 133/250
21/21 [==============================] - 4s 194ms/step - loss: 2.8928 - accura
cy: 0.8528 - val_loss: 2.3831 - val_accuracy: 0.8255
Epoch 134/250
21/21 [==============================] - 4s 196ms/step - loss: 2.8345 - accura
cy: 0.8487 - val_loss: 2.3360 - val_accuracy: 0.8411
Epoch 135/250
21/21 [==============================] - 4s 198ms/step - loss: 2.8055 - accura
cy: 0.8524 - val_loss: 1.8777 - val_accuracy: 0.8372
Epoch 136/250
21/21 [==============================] - 4s 194ms/step - loss: 2.7415 - accura
cy: 0.8518 - val_loss: 2.0324 - val_accuracy: 0.8193
Epoch 137/250
21/21 [==============================] - 4s 195ms/step - loss: 2.8235 - accura
cy: 0.8505 - val_loss: 2.4816 - val_accuracy: 0.8217
Epoch 138/250
21/21 [==============================] - 4s 195ms/step - loss: 2.7778 - accura
cy: 0.8427 - val_loss: 2.0563 - val_accuracy: 0.8341
Epoch 139/250
21/21 [==============================] - 4s 196ms/step - loss: 2.8324 - accura
cy: 0.8435 - val_loss: 1.9210 - val_accuracy: 0.8411
Epoch 140/250
21/21 [==============================] - 4s 191ms/step - loss: 2.7331 - accura
cy: 0.8452 - val_loss: 1.9951 - val_accuracy: 0.8442
```

```
Epoch 141/250
21/21 [==============================] - 4s 194ms/step - loss: 2.7291 - accura
cy: 0.8573 - val_loss: 1.6693 - val_accuracy: 0.8551
Epoch 142/250
21/21 [==============================] - 4s 193ms/step - loss: 2.7254 - accura
cy: 0.8540 - val_loss: 1.8644 - val_accuracy: 0.8357
Epoch 143/250
21/21 [==============================] - 4s 193ms/step - loss: 2.8309 - accura
cy: 0.8427 - val_loss: 1.8162 - val_accuracy: 0.8388
Epoch 144/250
21/21 [==============================] - 4s 194ms/step - loss: 2.8852 - accura
cy: 0.8477 - val_loss: 2.0211 - val_accuracy: 0.8512
Epoch 145/250
21/21 [==============================] - 4s 202ms/step - loss: 2.7570 - accura
cy: 0.8462 - val_loss: 2.4722 - val_accuracy: 0.8326
Epoch 146/250
21/21 [==============================] - 4s 193ms/step - loss: 2.7284 - accura
cy: 0.8497 - val_loss: 2.5915 - val_accuracy: 0.8006
Epoch 147/250
21/21 [==============================] - 4s 195ms/step - loss: 2.7304 - accura
cy: 0.8466 - val_loss: 1.8719 - val_accuracy: 0.8466
Epoch 148/250
21/21 [==============================] - 4s 197ms/step - loss: 2.6123 - accura
cy: 0.8583 - val_loss: 1.6755 - val_accuracy: 0.8435
Epoch 149/250
21/21 [==============================] - 4s 196ms/step - loss: 2.7260 - accura
cy: 0.8608 - val_loss: 1.7667 - val_accuracy: 0.8489
Epoch 150/250
21/21 [==============================] - 4s 195ms/step - loss: 2.6656 - accura
cy: 0.8598 - val_loss: 2.2188 - val_accuracy: 0.8388
Epoch 151/250
21/21 [==============================] - 4s 196ms/step - loss: 2.6535 - accura
cy: 0.8549 - val_loss: 2.1025 - val_accuracy: 0.8435
Epoch 152/250
21/21 [==============================] - 4s 193ms/step - loss: 2.6113 - accura
cy: 0.8483 - val_loss: 1.8920 - val_accuracy: 0.8536
Epoch 153/250
21/21 [==============================] - 4s 204ms/step - loss: 2.5775 - accura
cy: 0.8565 - val_loss: 1.9808 - val_accuracy: 0.8512
Epoch 154/250
21/21 [==============================] - 4s 196ms/step - loss: 2.6314 - accura
cy: 0.8620 - val_loss: 2.3471 - val_accuracy: 0.8583
Epoch 155/250
21/21 [==============================] - 4s 196ms/step - loss: 2.6117 - accura
cy: 0.8588 - val_loss: 1.7991 - val_accuracy: 0.8489
Epoch 156/250
21/21 [==============================] - 4s 194ms/step - loss: 2.6175 - accura
cy: 0.8581 - val_loss: 1.6474 - val_accuracy: 0.8419
Epoch 157/250
21/21 [==============================] - 4s 192ms/step - loss: 2.6793 - accura
cy: 0.8664 - val_loss: 2.2094 - val_accuracy: 0.8497
Epoch 158/250
21/21 [==============================] - 4s 193ms/step - loss: 2.6209 - accura
cy: 0.8585 - val_loss: 2.7192 - val_accuracy: 0.8505
Epoch 159/250
21/21 [==============================] - 4s 196ms/step - loss: 2.5960 - accura
cy: 0.8602 - val_loss: 1.8656 - val_accuracy: 0.8450
Epoch 160/250
21/21 [==============================] - 4s 194ms/step - loss: 2.5596 - accura
cy: 0.8536 - val_loss: 1.8234 - val_accuracy: 0.8551
```

```
Epoch 161/250
21/21 [==============================] - 4s 202ms/step - loss: 2.6377 - accura
cy: 0.8596 - val_loss: 2.5717 - val_accuracy: 0.8590
Epoch 162/250
21/21 [==============================] - 4s 204ms/step - loss: 2.7200 - accura
cy: 0.8544 - val_loss: 2.2471 - val_accuracy: 0.8372
Epoch 163/250
21/21 [==============================] - 4s 195ms/step - loss: 2.5697 - accura
cy: 0.8621 - val_loss: 2.4742 - val_accuracy: 0.8302
Epoch 164/250
21/21 [==============================] - 4s 204ms/step - loss: 2.6160 - accura
cy: 0.8499 - val_loss: 1.9475 - val_accuracy: 0.8364
Epoch 165/250
21/21 [==============================] - 4s 194ms/step - loss: 2.6792 - accura
cy: 0.8565 - val_loss: 2.0960 - val_accuracy: 0.8450
Epoch 166/250
21/21 [==============================] - 4s 194ms/step - loss: 2.7415 - accura
cy: 0.8567 - val_loss: 2.8363 - val_accuracy: 0.8466
Epoch 167/250
21/21 [==============================] - 4s 196ms/step - loss: 2.6183 - accura
cy: 0.8618 - val_loss: 2.5018 - val_accuracy: 0.8567
Epoch 168/250
21/21 [==============================] - 4s 203ms/step - loss: 2.5799 - accura
cy: 0.8585 - val_loss: 3.0861 - val_accuracy: 0.8653
Epoch 169/250
21/21 [==============================] - 4s 195ms/step - loss: 2.6588 - accura
cy: 0.8577 - val_loss: 1.8809 - val_accuracy: 0.8380
Epoch 170/250
21/21 [==============================] - 4s 198ms/step - loss: 2.6805 - accura
cy: 0.8524 - val_loss: 3.0463 - val_accuracy: 0.8567
Epoch 171/250
21/21 [==============================] - 4s 194ms/step - loss: 2.6260 - accura
cy: 0.8588 - val_loss: 3.1217 - val_accuracy: 0.8536
Epoch 172/250
21/21 [==============================] - 4s 198ms/step - loss: 2.6031 - accura
cy: 0.8602 - val_loss: 2.5766 - val_accuracy: 0.8497
Epoch 173/250
21/21 [==============================] - 4s 198ms/step - loss: 2.5124 - accura
cy: 0.8524 - val_loss: 1.8490 - val_accuracy: 0.8458
Epoch 174/250
21/21 [==============================] - 4s 193ms/step - loss: 2.4855 - accura
cy: 0.8707 - val_loss: 1.9932 - val_accuracy: 0.8419
Epoch 175/250
21/21 [==============================] - 4s 195ms/step - loss: 2.5071 - accura
cy: 0.8657 - val_loss: 1.8161 - val_accuracy: 0.8567
Epoch 176/250
21/21 [==============================] - 4s 193ms/step - loss: 2.4862 - accura
cy: 0.8586 - val_loss: 1.5567 - val_accuracy: 0.8512
Epoch 177/250
21/21 [==============================] - 4s 195ms/step - loss: 2.4718 - accura
cy: 0.8588 - val_loss: 1.9289 - val_accuracy: 0.8520
Epoch 178/250
21/21 [==============================] - 4s 197ms/step - loss: 2.6043 - accura
cy: 0.8612 - val_loss: 1.5969 - val_accuracy: 0.8489
Epoch 179/250
21/21 [==============================] - 4s 193ms/step - loss: 2.5080 - accura
cy: 0.8563 - val_loss: 1.7125 - val_accuracy: 0.8544
Epoch 180/250
21/21 [==============================] - 4s 194ms/step - loss: 2.5571 - accura
cy: 0.8590 - val_loss: 1.6832 - val_accuracy: 0.8528
```

```
Epoch 181/250
21/21 [==============================] - 4s 201ms/step - loss: 2.5403 - accura
cy: 0.8635 - val_loss: 1.7405 - val_accuracy: 0.8497
Epoch 182/250
21/21 [==============================] - 4s 196ms/step - loss: 2.4846 - accura
cy: 0.8676 - val_loss: 2.6865 - val_accuracy: 0.8481
Epoch 183/250
21/21 [==============================] - 4s 195ms/step - loss: 2.5146 - accura
cy: 0.8653 - val_loss: 2.3621 - val_accuracy: 0.8466
Epoch 184/250
21/21 [==============================] - 4s 195ms/step - loss: 2.5846 - accura
cy: 0.8594 - val_loss: 2.2488 - val_accuracy: 0.8567
Epoch 185/250
21/21 [==============================] - 4s 194ms/step - loss: 2.5541 - accura
cy: 0.8579 - val_loss: 2.1017 - val_accuracy: 0.8567
Epoch 186/250
21/21 [==============================] - 4s 195ms/step - loss: 2.4623 - accura
cy: 0.8725 - val_loss: 2.6967 - val_accuracy: 0.8458
Epoch 187/250
21/21 [==============================] - 4s 194ms/step - loss: 2.5479 - accura
cy: 0.8598 - val_loss: 1.6604 - val_accuracy: 0.8466
Epoch 188/250
21/21 [==============================] - 4s 194ms/step - loss: 2.5277 - accura
cy: 0.8606 - val_loss: 1.9929 - val_accuracy: 0.8380
Epoch 189/250
21/21 [==============================] - 4s 203ms/step - loss: 2.5071 - accura
cy: 0.8620 - val_loss: 3.3648 - val_accuracy: 0.8489
Epoch 190/250
21/21 [==============================] - 4s 196ms/step - loss: 2.5894 - accura
cy: 0.8666 - val_loss: 2.5202 - val_accuracy: 0.8559
Epoch 191/250
21/21 [==============================] - 4s 193ms/step - loss: 2.6144 - accura
cy: 0.8633 - val_loss: 1.9467 - val_accuracy: 0.8349
Epoch 192/250
21/21 [==============================] - 4s 193ms/step - loss: 2.4997 - accura
cy: 0.8588 - val_loss: 1.8071 - val_accuracy: 0.8590
Epoch 193/250
21/21 [==============================] - 4s 197ms/step - loss: 2.5087 - accura
cy: 0.8585 - val_loss: 1.7678 - val_accuracy: 0.8481
Epoch 194/250
21/21 [==============================] - 4s 196ms/step - loss: 2.5455 - accura
cy: 0.8549 - val_loss: 1.8101 - val_accuracy: 0.8450
Epoch 195/250
21/21 [==============================] - 4s 197ms/step - loss: 2.4624 - accura
cy: 0.8676 - val_loss: 1.9167 - val_accuracy: 0.8435
Epoch 196/250
21/21 [==============================] - 4s 193ms/step - loss: 2.5458 - accura
cy: 0.8674 - val_loss: 2.0296 - val_accuracy: 0.8544
Epoch 197/250
21/21 [==============================] - 4s 193ms/step - loss: 2.4873 - accura
cy: 0.8651 - val_loss: 1.5523 - val_accuracy: 0.8474
Epoch 198/250
21/21 [==============================] - 4s 195ms/step - loss: 2.4285 - accura
cy: 0.8620 - val_loss: 1.5382 - val_accuracy: 0.8505
Epoch 199/250
21/21 [==============================] - 4s 196ms/step - loss: 2.4773 - accura
cy: 0.8633 - val_loss: 1.7506 - val_accuracy: 0.8450
Epoch 200/250
21/21 [==============================] - 4s 197ms/step - loss: 2.4620 - accura
cy: 0.8651 - val_loss: 2.5263 - val_accuracy: 0.8536
```

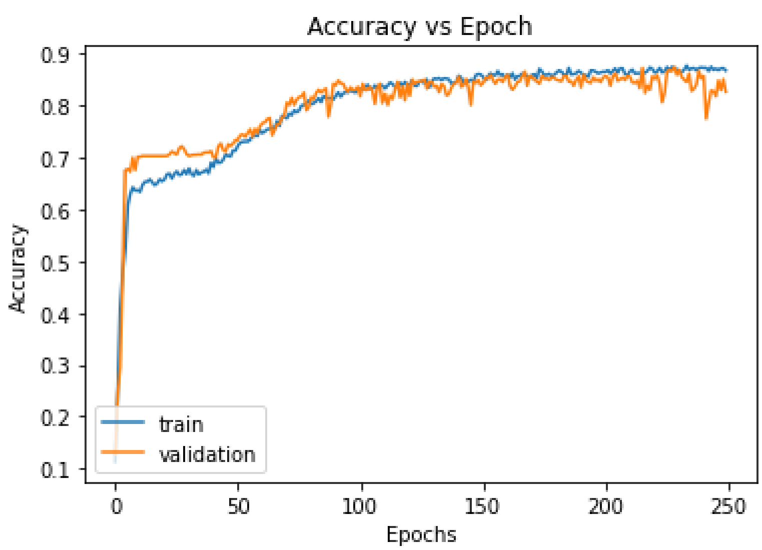
```
Epoch 201/250
21/21 [==============================] - 4s 194ms/step - loss: 2.5687 - accura
cy: 0.8647 - val_loss: 2.2440 - val_accuracy: 0.8567
Epoch 202/250
21/21 [==============================] - 4s 199ms/step - loss: 2.4676 - accura
cy: 0.8680 - val_loss: 4.3226 - val_accuracy: 0.8536
Epoch 203/250
21/21 [==============================] - 4s 195ms/step - loss: 2.6698 - accura
cy: 0.8618 - val_loss: 3.5973 - val_accuracy: 0.8380
Epoch 204/250
21/21 [==============================] - 4s 193ms/step - loss: 2.5465 - accura
cy: 0.8666 - val_loss: 3.4044 - val_accuracy: 0.8567
Epoch 205/250
21/21 [==============================] - 4s 195ms/step - loss: 2.5480 - accura
cy: 0.8715 - val_loss: 4.8325 - val_accuracy: 0.8614
Epoch 206/250
21/21 [==============================] - 4s 191ms/step - loss: 2.4912 - accura
cy: 0.8627 - val_loss: 2.7258 - val_accuracy: 0.8474
Epoch 207/250
21/21 [==============================] - 4s 195ms/step - loss: 2.4457 - accura
cy: 0.8717 - val_loss: 2.1473 - val_accuracy: 0.8505
Epoch 208/250
21/21 [==============================] - 4s 198ms/step - loss: 2.4554 - accura
cy: 0.8717 - val_loss: 1.6604 - val_accuracy: 0.8536
Epoch 209/250
21/21 [==============================] - 4s 194ms/step - loss: 2.4250 - accura
cy: 0.8608 - val_loss: 1.8312 - val_accuracy: 0.8489
Epoch 210/250
21/21 [==============================] - 4s 202ms/step - loss: 2.4648 - accura
cy: 0.8585 - val_loss: 1.7223 - val_accuracy: 0.8575
Epoch 211/250
21/21 [==============================] - 4s 197ms/step - loss: 2.4260 - accura
cy: 0.8657 - val_loss: 1.5606 - val_accuracy: 0.8551
Epoch 212/250
21/21 [==============================] - 4s 198ms/step - loss: 2.3748 - accura
cy: 0.8692 - val_loss: 1.4755 - val_accuracy: 0.8442
Epoch 213/250
21/21 [==============================] - 4s 195ms/step - loss: 2.3189 - accura
cy: 0.8618 - val_loss: 1.9535 - val_accuracy: 0.8435
Epoch 214/250
21/21 [==============================] - 4s 198ms/step - loss: 2.4394 - accura
cy: 0.8620 - val_loss: 1.6083 - val_accuracy: 0.8474
Epoch 215/250
21/21 [==============================] - 4s 194ms/step - loss: 2.6083 - accura
cy: 0.8623 - val_loss: 1.7066 - val_accuracy: 0.8333
Epoch 216/250
21/21 [==============================] - 4s 194ms/step - loss: 2.5191 - accura
cy: 0.8594 - val_loss: 1.5373 - val_accuracy: 0.8723
Epoch 217/250
21/21 [==============================] - 4s 194ms/step - loss: 2.4456 - accura
cy: 0.8637 - val_loss: 2.5561 - val_accuracy: 0.8240
Epoch 218/250
21/21 [==============================] - 4s 202ms/step - loss: 2.4316 - accura
cy: 0.8666 - val_loss: 2.5309 - val_accuracy: 0.8380
Epoch 219/250
21/21 [==============================] - 4s 200ms/step - loss: 2.4126 - accura
cy: 0.8734 - val_loss: 1.5793 - val_accuracy: 0.8302
Epoch 220/250
21/21 [==============================] - 4s 194ms/step - loss: 2.5096 - accura
cy: 0.8703 - val_loss: 1.5808 - val_accuracy: 0.8559
```

```
Epoch 221/250
21/21 [==============================] - 4s 197ms/step - loss: 2.5243 - accura
cy: 0.8629 - val_loss: 1.5774 - val_accuracy: 0.8512
Epoch 222/250
21/21 [==============================] - 4s 196ms/step - loss: 2.5037 - accura
cy: 0.8660 - val_loss: 1.9393 - val_accuracy: 0.8583
Epoch 223/250
21/21 [==============================] - 4s 188ms/step - loss: 2.4420 - accura
cy: 0.8727 - val_loss: 2.5860 - val_accuracy: 0.8419
Epoch 224/250
21/21 [==============================] - 4s 196ms/step - loss: 2.4801 - accura
cy: 0.8635 - val_loss: 2.0790 - val_accuracy: 0.8061
Epoch 225/250
21/21 [==============================] - 4s 202ms/step - loss: 2.3692 - accura
cy: 0.8660 - val_loss: 2.0822 - val_accuracy: 0.8240
Epoch 226/250
21/21 [==============================] - 4s 196ms/step - loss: 2.3770 - accura
cy: 0.8742 - val_loss: 2.0561 - val_accuracy: 0.8637
Epoch 227/250
21/21 [==============================] - 4s 195ms/step - loss: 2.6061 - accura
cy: 0.8674 - val_loss: 3.1190 - val_accuracy: 0.8684
Epoch 228/250
21/21 [==============================] - 4s 191ms/step - loss: 2.5567 - accura
cy: 0.8744 - val_loss: 2.5652 - val_accuracy: 0.8660
Epoch 229/250
21/21 [==============================] - 4s 194ms/step - loss: 2.4930 - accura
cy: 0.8723 - val_loss: 2.3971 - val_accuracy: 0.8738
Epoch 230/250
21/21 [==============================] - 4s 195ms/step - loss: 2.3376 - accura
cy: 0.8694 - val_loss: 2.8456 - val_accuracy: 0.8590
Epoch 231/250
21/21 [==============================] - 4s 193ms/step - loss: 2.4812 - accura
cy: 0.8668 - val_loss: 1.9561 - val_accuracy: 0.8684
Epoch 232/250
21/21 [==============================] - 4s 195ms/step - loss: 2.3914 - accura
cy: 0.8639 - val_loss: 1.6831 - val_accuracy: 0.8551
Epoch 233/250
21/21 [==============================] - 4s 197ms/step - loss: 2.4993 - accura
cy: 0.8690 - val_loss: 2.1012 - val_accuracy: 0.8497
Epoch 234/250
21/21 [==============================] - 4s 196ms/step - loss: 2.3688 - accura
cy: 0.8771 - val_loss: 2.4687 - val_accuracy: 0.8450
Epoch 235/250
21/21 [==============================] - 4s 193ms/step - loss: 2.4237 - accura
cy: 0.8684 - val_loss: 1.5338 - val_accuracy: 0.8614
Epoch 236/250
21/21 [==============================] - 4s 198ms/step - loss: 2.3833 - accura
cy: 0.8715 - val_loss: 1.9599 - val_accuracy: 0.8279
Epoch 237/250
21/21 [==============================] - 4s 196ms/step - loss: 2.3829 - accura
cy: 0.8723 - val_loss: 2.1328 - val_accuracy: 0.8318
Epoch 238/250
21/21 [==============================] - 4s 194ms/step - loss: 2.4078 - accura
cy: 0.8674 - val_loss: 1.6384 - val_accuracy: 0.8403
Epoch 239/250
21/21 [==============================] - 4s 197ms/step - loss: 2.3539 - accura
cy: 0.8723 - val_loss: 1.5748 - val_accuracy: 0.8660
Epoch 240/250
21/21 [==============================] - 4s 198ms/step - loss: 2.4025 - accura
cy: 0.8732 - val_loss: 1.4063 - val_accuracy: 0.8520
```

```
Epoch 241/250
21/21 [==============================] - 4s 198ms/step - loss: 2.3747 - accura
cy: 0.8725 - val_loss: 1.8593 - val_accuracy: 0.8567
Epoch 242/250
21/21 [==============================] - 4s 199ms/step - loss: 2.5624 - accura
cy: 0.8727 - val_loss: 3.0753 - val_accuracy: 0.7741
Epoch 243/250
21/21 [==============================] - 4s 195ms/step - loss: 2.5967 - accura
cy: 0.8664 - val_loss: 1.9770 - val_accuracy: 0.8037
Epoch 244/250
21/21 [==============================] - 4s 194ms/step - loss: 2.4302 - accura
cy: 0.8748 - val_loss: 2.4588 - val_accuracy: 0.8287
Epoch 245/250
21/21 [==============================] - 4s 196ms/step - loss: 2.3474 - accura
cy: 0.8686 - val_loss: 1.9550 - val_accuracy: 0.8271
Epoch 246/250
21/21 [==============================] - 4s 197ms/step - loss: 2.3512 - accura
cy: 0.8711 - val_loss: 2.2096 - val_accuracy: 0.8178
Epoch 247/250
21/21 [==============================] - 4s 194ms/step - loss: 2.3468 - accura
cy: 0.8680 - val_loss: 2.4699 - val_accuracy: 0.8481
Epoch 248/250
21/21 [==============================] - 4s 195ms/step - loss: 2.5293 - accura
cy: 0.8715 - val_loss: 2.1987 - val_accuracy: 0.8294
Epoch 249/250
21/21 [==============================] - 4s 195ms/step - loss: 2.4251 - accura
cy: 0.8721 - val_loss: 1.7820 - val_accuracy: 0.8505
Epoch 250/250
21/21 [==============================] - 4s 195ms/step - loss: 2.5375 - accura
cy: 0.8674 - val_loss: 2.8170 - val_accuracy: 0.8263
```

In [108…

```
plt.plot(history_transfer_learning_augment.history['accuracy'])
plt.plot(history_transfer_learning_augment.history['val_accuracy'])
plt.title('Accuracy vs Epoch')
plt.ylabel('Accuracy')
plt.xlabel('Epochs')
plt.legend(['train', 'validation'], loc='lower left')
plt.show()
```



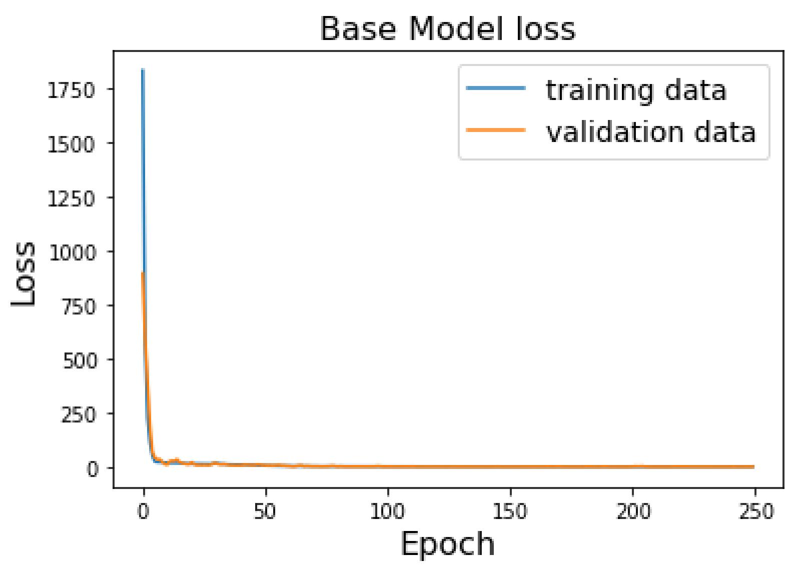
In [101…

```
#plt.figure(figsize=(8,8))
plt.plot(history_transfer_learning_augment.history['loss'])
plt.plot(history_transfer_learning_augment.history['val_loss'])
```

```
plt.title('Base Model loss',fontsize=16)
plt.ylabel('Loss',fontsize=16)
plt.xlabel('Epoch',fontsize=16)
plt.legend(['training data', 'validation data'], loc='upper right',fontsize=14)
```

Out[101]:

```
<matplotlib.legend.Legend at 0x7f1ec92da3d0>
```



Test Data

Here we visualize the predictions on the test data using the model that performed the best. In our case, this was "model_transfer_learning", which was the model where we dropped data that had missing values and increased the size of the training data size using data augmentation methods and later applied pre-trained neural nets.

In [102…

```
# Laod test data
df_test = pd.read_csv('test.csv')
df_test.shape
```

Out[102]:

```
(1783, 2)
```

In [103…

```
#Convert test data images to numpy array
test_images = convert_data_to_image(df_test)
```

In [104…

```
%%time
test_preds = model_transfer_learning.predict(test_images)
```

```
CPU times: user 664 ms, sys: 14.2 ms, total: 678 ms
Wall time: 694 ms
```

In [105…

```
test_preds
```

Out[105]:
```
array([[66.58177 , 35.0434  , 30.411098, ..., 70.49301 , 51.594803,
        83.45485 ],
       [66.56351 , 35.00365 , 30.397743, ..., 70.44331 , 51.608395,
        83.421715],
       [66.550674, 35.03766 , 30.406645, ..., 70.44105 , 51.64615 ,
        83.32666 ],
       ...,
       [66.579765, 34.99425 , 30.397264, ..., 70.432014, 51.605526,
        83.42317 ],
       [66.61526 , 35.055393, 30.41927 , ..., 70.48153 , 51.555717,
        83.43022 ],
       [66.562004, 34.96918 , 30.399776, ..., 70.48713 , 51.609028,
        83.432365]], dtype=float32)
```

In [107…
```
# Let's see if everything looks good !
fig = plt.figure(figsize=(20,20))
for i in range(20):
    axis = fig.add_subplot(4, 5, i+1, xticks=[], yticks=[])
    plot_sample(test_images[i], test_preds[i], axis, "")
plt.show()
```

Thus, for facial detection data augmentation coupled with transfer learning proved to be the best approach. As a future work, we can add more augmentation to our training data that, we believe, will improve our accuracy even more. We were successful in beating the previous record of accuracy in this project by a significant amount.

```
In [ ]:
```