Heartbeat disease classifier

In [1]:

```
# Pandas
import pandas as pd

# Scikit learn
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, accuracy_score, confusion_matrix
from sklearn.preprocessing import LabelEncoder
from sklearn.utils import shuffle
from sklearn.utils import class_weight

# Keras
from keras.models import Sequential
from keras.layers import Dense, Dropout, Activation, Flatten
from keras.layers import Convolution2D, Conv2D, MaxPooling2D, GlobalAveragePooling2D
from keras.utils import to_categorical

# Audio
import librosa
import librosa.display

# Plot
import matplotlib.pyplot as plt

# Utility
import os
import glob
import numpy as np
from tqdm import tqdm
import itertools
```

```
Using TensorFlow backend.
```

Build Dataset

In [2]:

```
%%time
dataset = []
for folder in ["../input/set_a/**","../input/set_b/**"]:
    for filename in glob.iglob(folder):
        if os.path.exists(filename):
            label = os.path.basename(filename).split("_")[0]
            # skip audio smaller than 4 secs
            if librosa.get_duration(filename=filename)>=4:
                if label not in ["Aunlabelledtest", "Bunlabelledtest"]:
                    dataset.append({
                        "filename": filename,
                        "label": label
                    })
dataset = pd.DataFrame(dataset)
dataset = shuffle(dataset, random_state=42)
```

```
CPU times: user 136 ms, sys: 156 ms, total: 292 ms
Wall time: 1.46 s
```

In [3]:

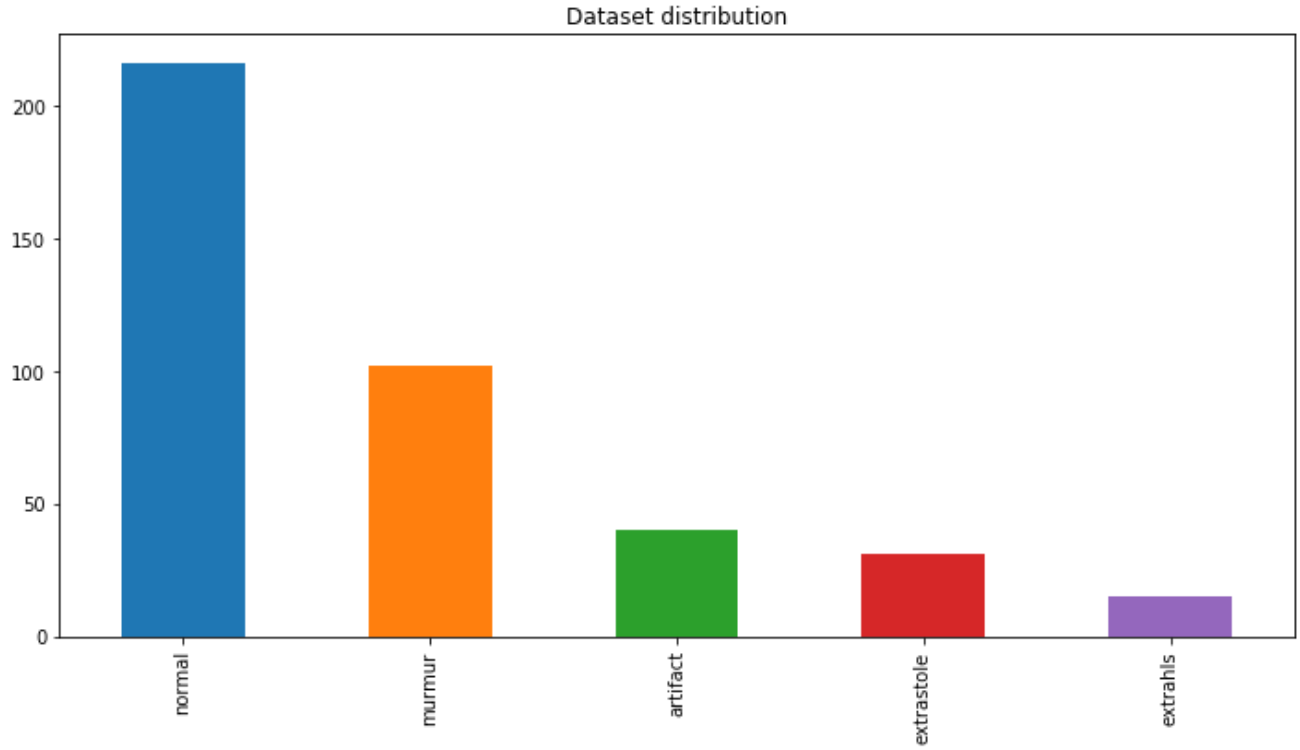
```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 404 entries, 70 to 102
Data columns (total 2 columns):
```

```
filename    404 non-null object
label       404 non-null object
dtypes: object(2)
memory usage: 9.5+ KB
```

In [4]:

```
plt.figure(figsize=(12,6))
dataset.label.value_counts().plot(kind='bar', title="Dataset distribution")
plt.show()
```



Split dataset in train and test

In [5]:

```
train, test = train_test_split(dataset, test_size=0.2, random_state=42)

print("Train: %i" % len(train))
print("Test: %i" % len(test))
```

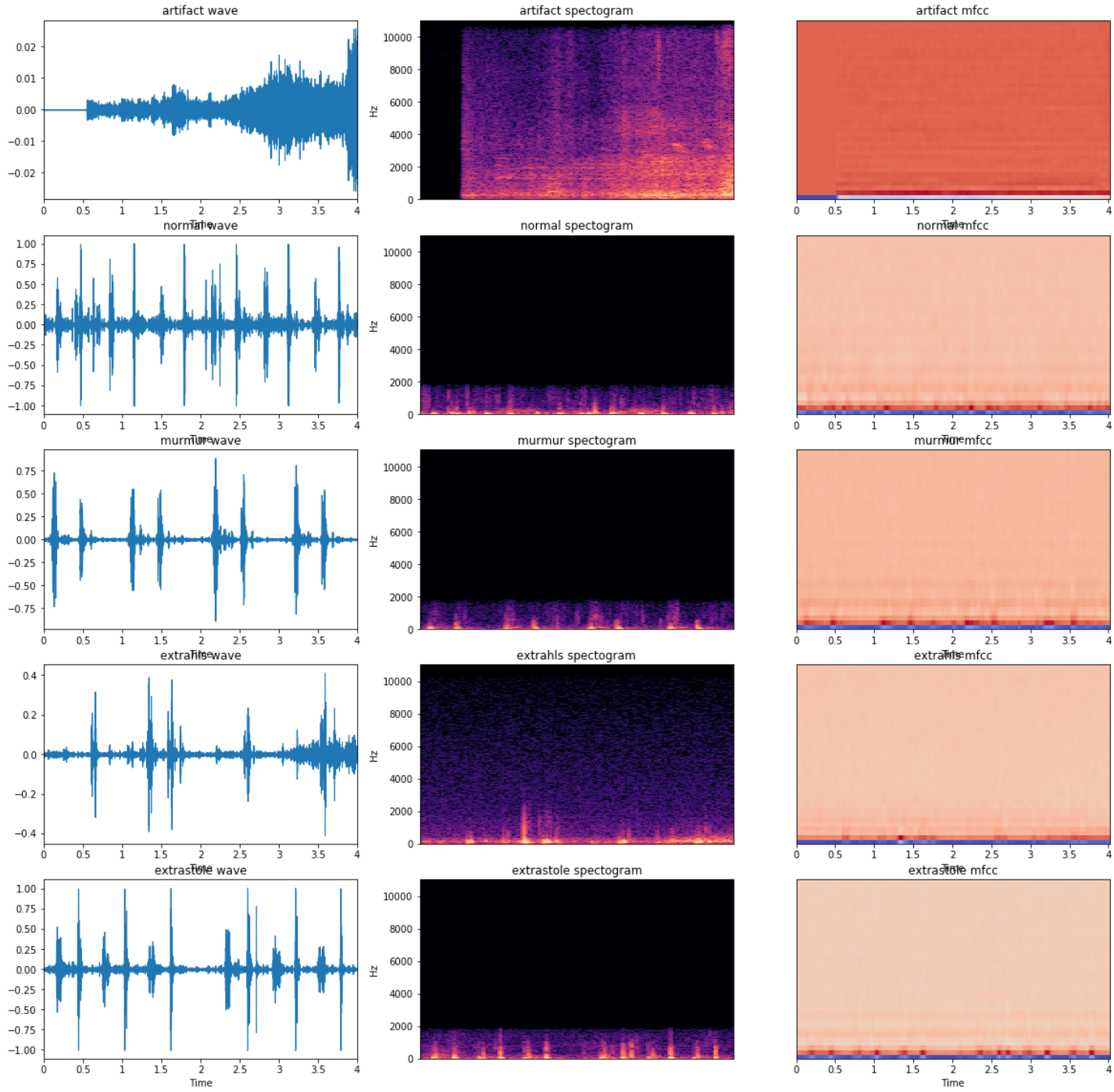
```
Train: 323
Test: 81
```

Show Audio info

In [6]:

```
%%time
plt.figure(figsize=(20,20))
idx = 0
for label in dataset.label.unique():
    y, sr = librosa.load(dataset[dataset.label==label].filename.iloc[0], duration=4)
    idx+=1
    plt.subplot(5, 3, idx)
    plt.title("%s wave" % label)
    librosa.display.waveplot(y, sr=sr)
    idx+=1
    plt.subplot(5, 3, idx)
    D = librosa.amplitude_to_db(np.abs(librosa.stft(y)), ref=np.max)
    librosa.display.specshow(D, y_axis='linear')
    plt.title("%s spectogram" % label)
    idx+=1
    mfccs = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40)
```

```
    plt.subplot(5, 3, idx)
    librosa.display.specshow(mfccs, x_axis='time')
    plt.title("%s mfcc" % label)
plt.show()
```




```
CPU times: user 5.95 s, sys: 2 s, total: 7.94 s
Wall time: 5.44 s
```

Extract features from audio

In [7]:

```
def extract_features(audio_path):
    y, sr = librosa.load(audio_path, duration=4)
    mfccs = librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40)
    return mfccs
```

In [8]:

```
%%time
x_train, x_test = [], []
print("Extract features from TRAIN  and TEST dataset")
for idx in tqdm(range(len(train))):
    x_train.append(extract_features(train.filename.iloc[idx]))
```

```
for idx in tqdm(range(len(test))):
    x_test.append(extract_features(test.filename.iloc[idx]))


x_test = np.asarray(x_test)
x_train = np.asarray(x_train)

print("X train:", x_train.shape)
print("X test:", x_test.shape)
```

```
  0%|          | 0/323 [00:00<?, ?it/s]
```

```
Extract features from TRAIN  and TEST dataset
```

```
100%|██████████| 323/323 [01:01<00:00,  4.68it/s]
100%|██████████| 81/81 [00:16<00:00,  4.54it/s]
```

```
X train: (323, 40, 173)
X test: (81, 40, 173)
CPU times: user 1min 28s, sys: 35.2 s, total: 2min 4s
Wall time: 1min 17s
```

Encode labels

In [9]:

```
%%time
encoder = LabelEncoder()
encoder.fit(train.label)

y_train = encoder.transform(train.label)
y_test = encoder.transform(test.label)
```

```
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 655 µs
```

Compute class weights

In [10]:

```
class_weights = class_weight.compute_class_weight('balanced',
                                                  np.unique(y_train),
                                                  y_train)
```

Input shapes

In [11]:

```
x_train = x_train.reshape(x_train.shape[0], x_train.shape[1], x_train.shape[2], 1)
x_test = x_test.reshape(x_test.shape[0], x_test.shape[1], x_test.shape[2], 1)
y_train = to_categorical(y_train)
y_test = to_categorical(y_test)



print("X train:", x_train.shape)
print("Y train:", y_train.shape)
print()
print("X test:", x_test.shape)
print("Y test:", y_test.shape)
```

```
X train: (323, 40, 173, 1)
Y train: (323, 5)

X test: (81, 40, 173, 1)
Y test: (81, 5)
```

Build Model

In [12]:

```
model = Sequential()
model.add(Conv2D(filters=16, kernel_size=2, input_shape=(x_train.shape[1],x_train.shape[
2],x_train.shape[3]), activation='relu'))
model.add(MaxPooling2D(pool_size=2))
model.add(Dropout(0.2))

model.add(Conv2D(filters=32, kernel_size=2, activation='relu'))
model.add(MaxPooling2D(pool_size=2))
model.add(Dropout(0.2))

model.add(Conv2D(filters=64, kernel_size=2, activation='relu'))
model.add(MaxPooling2D(pool_size=2))
model.add(Dropout(0.2))

model.add(Conv2D(filters=128, kernel_size=2, activation='relu'))
model.add(MaxPooling2D(pool_size=2))
model.add(Dropout(0.5))
model.add(GlobalAveragePooling2D())

model.add(Dense(len(encoder.classes_), activation='softmax'))
model.summary()
```

```
WARNING:tensorflow:From /opt/conda/lib/python3.6/site-packages/tensorflow/python/framewor
k/op_def_library.py:263: colocate_with (from tensorflow.python.framework.ops) is deprecat
ed and will be removed in a future version.
Instructions for updating:
Colocations handled automatically by placer.
WARNING:tensorflow:From /opt/conda/lib/python3.6/site-packages/keras/backend/tensorflow_b
ackend.py:3445: calling dropout (from tensorflow.python.ops.nn_ops) with keep_prob is dep
recated and will be removed in a future version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 39, 172, 16)       80
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 19, 86, 16)        0
_________________________________________________________________
dropout_1 (Dropout)          (None, 19, 86, 16)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 18, 85, 32)        2080
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 9, 42, 32)         0
_________________________________________________________________
dropout_2 (Dropout)          (None, 9, 42, 32)         0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 8, 41, 64)         8256
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 4, 20, 64)         0
_________________________________________________________________
dropout_3 (Dropout)          (None, 4, 20, 64)         0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 3, 19, 128)        32896
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 1, 9, 128)         0
_________________________________________________________________
dropout_4 (Dropout)          (None, 1, 9, 128)         0
_________________________________________________________________
global_average_pooling2d_1 ( (None, 128)               0
_________________________________________________________________
dense_1 (Dense)              (None, 5)                 645
=================================================================
Total params: 43,957
Trainable params: 43,957
Non-trainable params: 0
```

_____

Compile model

In [13]:

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

Fit model

In [14]:

```
%%time
history = model.fit(x_train, y_train,
              batch_size=128,
              epochs=300,
              validation_data=(x_test, y_test),
              class_weight=class_weights,
              shuffle=True)
```

```
WARNING:tensorflow:From /opt/conda/lib/python3.6/site-packages/tensorflow/python/ops/math
_ops.py:3066: to_int32 (from tensorflow.python.ops.math_ops) is deprecated and will be re
moved in a future version.
Instructions for updating:
Use tf.cast instead.
Train on 323 samples, validate on 81 samples
Epoch 1/300
323/323 [==============================] - 3s 9ms/step - loss: 8.0145 - acc: 0.3591 - val
_loss: 7.0338 - val_acc: 0.5062
Epoch 2/300
323/323 [==============================] - 0s 137us/step - loss: 7.2088 - acc: 0.5418 - v
al_loss: 6.9138 - val_acc: 0.5062
Epoch 3/300
323/323 [==============================] - 0s 127us/step - loss: 7.2224 - acc: 0.5418 - v
al_loss: 6.0484 - val_acc: 0.5062
Epoch 4/300
323/323 [==============================] - 0s 122us/step - loss: 6.8072 - acc: 0.5449 - v
al_loss: 4.8574 - val_acc: 0.6173
Epoch 5/300
323/323 [==============================] - 0s 122us/step - loss: 7.1333 - acc: 0.5108 - v
al_loss: 5.8427 - val_acc: 0.5309
Epoch 6/300
323/323 [==============================] - 0s 123us/step - loss: 6.8571 - acc: 0.5542 - v
al_loss: 5.6974 - val_acc: 0.5309
Epoch 7/300
323/323 [==============================] - 0s 124us/step - loss: 6.8464 - acc: 0.5573 - v
al_loss: 4.7642 - val_acc: 0.6420
Epoch 8/300
323/323 [==============================] - 0s 124us/step - loss: 6.7641 - acc: 0.5201 - v
al_loss: 4.7291 - val_acc: 0.6543
Epoch 9/300
323/323 [==============================] - 0s 125us/step - loss: 6.6175 - acc: 0.5697 - v
al_loss: 4.9089 - val_acc: 0.6543
Epoch 10/300
323/323 [==============================] - 0s 125us/step - loss: 6.6555 - acc: 0.5666 - v
al_loss: 4.9271 - val_acc: 0.6420
Epoch 11/300
323/323 [==============================] - 0s 125us/step - loss: 6.5995 - acc: 0.5728 - v
al_loss: 4.6531 - val_acc: 0.6543
Epoch 12/300
323/323 [==============================] - 0s 124us/step - loss: 6.3939 - acc: 0.5851 - v
al_loss: 4.6826 - val_acc: 0.6790
Epoch 13/300
323/323 [==============================] - 0s 122us/step - loss: 6.5727 - acc: 0.5635 - v
al_loss: 4.6112 - val_acc: 0.6790
Epoch 14/300
323/323 [==============================] - 0s 121us/step - loss: 6.5231 - acc: 0.5820 - v
al_loss: 5.1777 - val_acc: 0.5926
Epoch 15/300
323/323 [==============================] - 0s 127us/step - loss: 6.6886 - acc: 0.5697 - v
```

```
al_loss: 5.0225 - val_acc: 0.5926
Epoch 16/300
323/323 [==============================] - 0s 122us/step - loss: 6.6567 - acc: 0.5728 - v
al_loss: 4.6099 - val_acc: 0.6543
Epoch 17/300
323/323 [==============================] - 0s 120us/step - loss: 6.3924 - acc: 0.5851 - v
al_loss: 4.6122 - val_acc: 0.6790
Epoch 18/300
323/323 [==============================] - 0s 124us/step - loss: 6.4775 - acc: 0.5697 - v
al_loss: 4.5408 - val_acc: 0.6914
Epoch 19/300
323/323 [==============================] - 0s 123us/step - loss: 6.4514 - acc: 0.5851 - v
al_loss: 4.5258 - val_acc: 0.6790
Epoch 20/300
323/323 [==============================] - 0s 120us/step - loss: 6.4706 - acc: 0.5820 - v
al_loss: 4.5059 - val_acc: 0.6790
Epoch 21/300
323/323 [==============================] - 0s 126us/step - loss: 6.4388 - acc: 0.5913 - v
al_loss: 4.4854 - val_acc: 0.6667
Epoch 22/300
323/323 [==============================] - 0s 124us/step - loss: 6.4219 - acc: 0.5851 - v
al_loss: 4.5018 - val_acc: 0.6667
Epoch 23/300
323/323 [==============================] - 0s 122us/step - loss: 6.3478 - acc: 0.5944 - v
al_loss: 4.4786 - val_acc: 0.6667
Epoch 24/300
323/323 [==============================] - 0s 122us/step - loss: 6.3345 - acc: 0.5944 - v
al_loss: 4.4264 - val_acc: 0.6790
Epoch 25/300
323/323 [==============================] - 0s 122us/step - loss: 6.3912 - acc: 0.5882 - v
al_loss: 4.3955 - val_acc: 0.6914
Epoch 26/300
323/323 [==============================] - 0s 124us/step - loss: 6.3153 - acc: 0.6037 - v
al_loss: 4.3484 - val_acc: 0.7160
Epoch 27/300
323/323 [==============================] - 0s 123us/step - loss: 6.3464 - acc: 0.5944 - v
al_loss: 4.3391 - val_acc: 0.7037
Epoch 28/300
323/323 [==============================] - 0s 122us/step - loss: 6.3726 - acc: 0.5820 - v
al_loss: 4.4082 - val_acc: 0.6667
Epoch 29/300
323/323 [==============================] - 0s 127us/step - loss: 6.4479 - acc: 0.5851 - v
al_loss: 4.4610 - val_acc: 0.6543
Epoch 30/300
323/323 [==============================] - 0s 124us/step - loss: 6.3625 - acc: 0.5913 - v
al_loss: 4.2446 - val_acc: 0.6667
Epoch 31/300
323/323 [==============================] - 0s 128us/step - loss: 6.3891 - acc: 0.5820 - v
al_loss: 4.0362 - val_acc: 0.7160
Epoch 32/300
323/323 [==============================] - 0s 125us/step - loss: 6.3656 - acc: 0.5944 - v
al_loss: 4.0923 - val_acc: 0.6790
Epoch 33/300
323/323 [==============================] - 0s 128us/step - loss: 6.3237 - acc: 0.5944 - v
al_loss: 4.2163 - val_acc: 0.6420
Epoch 34/300
323/323 [==============================] - 0s 125us/step - loss: 6.3151 - acc: 0.5944 - v
al_loss: 4.1105 - val_acc: 0.6543
Epoch 35/300
323/323 [==============================] - 0s 123us/step - loss: 6.3346 - acc: 0.5851 - v
al_loss: 3.7839 - val_acc: 0.6914
Epoch 36/300
323/323 [==============================] - 0s 126us/step - loss: 6.3588 - acc: 0.5882 - v
al_loss: 3.5905 - val_acc: 0.7160
Epoch 37/300
323/323 [==============================] - 0s 124us/step - loss: 6.3462 - acc: 0.5851 - v
al_loss: 3.4991 - val_acc: 0.6790
Epoch 38/300
323/323 [==============================] - 0s 126us/step - loss: 6.3169 - acc: 0.5882 - v
al_loss: 3.3245 - val_acc: 0.6667
Epoch 39/300
323/323 [==============================] - 0s 126us/step - loss: 6.2728 - acc: 0.5913 - v
```

al_loss: 3.0363 - val_acc: 0.6543
Epoch 40/300
323/323 [==============================] - 0s 124us/step - loss: 6.1780 - acc: 0.5944 - v
al_loss: 2.3290 - val_acc: 0.6296
Epoch 41/300
323/323 [==============================] - 0s 130us/step - loss: 5.4679 - acc: 0.5046 - v
al_loss: 1.2547 - val_acc: 0.6667
Epoch 42/300
323/323 [==============================] - 0s 125us/step - loss: 2.8303 - acc: 0.3808 - v
al_loss: 1.0837 - val_acc: 0.4444
Epoch 43/300
323/323 [==============================] - 0s 124us/step - loss: 2.2433 - acc: 0.4427 - v
al_loss: 0.9310 - val_acc: 0.6790
Epoch 44/300
323/323 [==============================] - 0s 118us/step - loss: 2.1170 - acc: 0.5913 - v
al_loss: 0.8748 - val_acc: 0.6914
Epoch 45/300
323/323 [==============================] - 0s 122us/step - loss: 1.3136 - acc: 0.5511 - v
al_loss: 1.0671 - val_acc: 0.7037
Epoch 46/300
323/323 [==============================] - 0s 125us/step - loss: 1.4124 - acc: 0.3591 - v
al_loss: 1.1321 - val_acc: 0.7037
Epoch 47/300
323/323 [==============================] - 0s 125us/step - loss: 1.2213 - acc: 0.4396 - v
al_loss: 1.0802 - val_acc: 0.6914
Epoch 48/300
323/323 [==============================] - 0s 123us/step - loss: 1.0708 - acc: 0.5511 - v
al_loss: 1.0557 - val_acc: 0.7037
Epoch 49/300
323/323 [==============================] - 0s 123us/step - loss: 1.0227 - acc: 0.6130 - v
al_loss: 1.0689 - val_acc: 0.7037
Epoch 50/300
323/323 [==============================] - 0s 126us/step - loss: 1.0514 - acc: 0.5851 - v
al_loss: 1.1123 - val_acc: 0.6914
Epoch 51/300
323/323 [==============================] - 0s 123us/step - loss: 1.0122 - acc: 0.5851 - v
al_loss: 1.1452 - val_acc: 0.6667
Epoch 52/300
323/323 [==============================] - 0s 126us/step - loss: 0.9725 - acc: 0.5697 - v
al_loss: 1.1546 - val_acc: 0.6914
Epoch 53/300
323/323 [==============================] - 0s 126us/step - loss: 0.9821 - acc: 0.5387 - v
al_loss: 1.1448 - val_acc: 0.7160
Epoch 54/300
323/323 [==============================] - 0s 123us/step - loss: 0.9616 - acc: 0.5387 - v
al_loss: 1.1171 - val_acc: 0.6667
Epoch 55/300
323/323 [==============================] - 0s 125us/step - loss: 0.9585 - acc: 0.5604 - v
al_loss: 1.0789 - val_acc: 0.6543
Epoch 56/300
323/323 [==============================] - 0s 128us/step - loss: 0.9613 - acc: 0.5944 - v
al_loss: 1.0491 - val_acc: 0.6543
Epoch 57/300
323/323 [==============================] - 0s 123us/step - loss: 0.9199 - acc: 0.6223 - v
al_loss: 1.0244 - val_acc: 0.6790
Epoch 58/300
323/323 [==============================] - 0s 124us/step - loss: 0.9130 - acc: 0.6006 - v
al_loss: 1.0013 - val_acc: 0.6914
Epoch 59/300
323/323 [==============================] - 0s 124us/step - loss: 0.9052 - acc: 0.6161 - v
al_loss: 0.9844 - val_acc: 0.7037
Epoch 60/300
323/323 [==============================] - 0s 124us/step - loss: 0.9020 - acc: 0.6130 - v
al_loss: 0.9854 - val_acc: 0.7407
Epoch 61/300
323/323 [==============================] - 0s 126us/step - loss: 0.9172 - acc: 0.5728 - v
al_loss: 0.9858 - val_acc: 0.7407
Epoch 62/300
323/323 [==============================] - 0s 132us/step - loss: 0.8853 - acc: 0.6347 - v
al_loss: 0.9801 - val_acc: 0.7531
Epoch 63/300
323/323 [==============================] - 0s 126us/step - loss: 0.8851 - acc: 0.6471 - v

al_loss: 0.9606 - val_acc: 0.7407
Epoch 64/300
323/323 [==============================] - 0s 134us/step - loss: 0.8834 - acc: 0.6254 - v
al_loss: 0.9414 - val_acc: 0.7407
Epoch 65/300
323/323 [==============================] - 0s 123us/step - loss: 0.8758 - acc: 0.6254 - v
al_loss: 0.9289 - val_acc: 0.7407
Epoch 66/300
323/323 [==============================] - 0s 127us/step - loss: 0.8669 - acc: 0.6409 - v
al_loss: 0.9087 - val_acc: 0.7654
Epoch 67/300
323/323 [==============================] - 0s 122us/step - loss: 0.8740 - acc: 0.6440 - v
al_loss: 0.8878 - val_acc: 0.7531
Epoch 68/300
323/323 [==============================] - 0s 122us/step - loss: 0.8714 - acc: 0.6533 - v
al_loss: 0.8722 - val_acc: 0.7531
Epoch 69/300
323/323 [==============================] - 0s 121us/step - loss: 0.8648 - acc: 0.6502 - v
al_loss: 0.8598 - val_acc: 0.7531
Epoch 70/300
323/323 [==============================] - 0s 121us/step - loss: 0.8500 - acc: 0.6502 - v
al_loss: 0.8578 - val_acc: 0.7531
Epoch 71/300
323/323 [==============================] - 0s 120us/step - loss: 0.8271 - acc: 0.6594 - v
al_loss: 0.8653 - val_acc: 0.7407
Epoch 72/300
323/323 [==============================] - 0s 125us/step - loss: 0.8312 - acc: 0.6440 - v
al_loss: 0.8617 - val_acc: 0.7654
Epoch 73/300
323/323 [==============================] - 0s 126us/step - loss: 0.8291 - acc: 0.6780 - v
al_loss: 0.8301 - val_acc: 0.7407
Epoch 74/300
323/323 [==============================] - 0s 128us/step - loss: 0.8305 - acc: 0.6533 - v
al_loss: 0.8205 - val_acc: 0.7407
Epoch 75/300
323/323 [==============================] - 0s 135us/step - loss: 0.8121 - acc: 0.6656 - v
al_loss: 0.8250 - val_acc: 0.7531
Epoch 76/300
323/323 [==============================] - 0s 124us/step - loss: 0.7918 - acc: 0.6904 - v
al_loss: 0.8166 - val_acc: 0.7531
Epoch 77/300
323/323 [==============================] - 0s 125us/step - loss: 0.7899 - acc: 0.6718 - v
al_loss: 0.8045 - val_acc: 0.7407
Epoch 78/300
323/323 [==============================] - 0s 121us/step - loss: 0.8092 - acc: 0.6718 - v
al_loss: 0.7818 - val_acc: 0.7407
Epoch 79/300
323/323 [==============================] - 0s 123us/step - loss: 0.7909 - acc: 0.6718 - v
al_loss: 0.7842 - val_acc: 0.7531
Epoch 80/300
323/323 [==============================] - 0s 126us/step - loss: 0.7916 - acc: 0.6904 - v
al_loss: 0.7894 - val_acc: 0.7778
Epoch 81/300
323/323 [==============================] - 0s 125us/step - loss: 0.7838 - acc: 0.6997 - v
al_loss: 0.7709 - val_acc: 0.7531
Epoch 82/300
323/323 [==============================] - 0s 121us/step - loss: 0.7621 - acc: 0.6904 - v
al_loss: 0.7476 - val_acc: 0.7531
Epoch 83/300
323/323 [==============================] - 0s 133us/step - loss: 0.7503 - acc: 0.6997 - v
al_loss: 0.7426 - val_acc: 0.7531
Epoch 84/300
323/323 [==============================] - 0s 124us/step - loss: 0.7452 - acc: 0.6997 - v
al_loss: 0.7398 - val_acc: 0.7654
Epoch 85/300
323/323 [==============================] - 0s 122us/step - loss: 0.7364 - acc: 0.6718 - v
al_loss: 0.7380 - val_acc: 0.7654
Epoch 86/300
323/323 [==============================] - 0s 127us/step - loss: 0.7218 - acc: 0.6904 - v
al_loss: 0.7200 - val_acc: 0.7654
Epoch 87/300
323/323 [==============================] - 0s 125us/step - loss: 0.7631 - acc: 0.6842 - v

al_loss: 0.7054 - val_acc: 0.7531
Epoch 88/300
323/323 [==============================] - 0s 130us/step - loss: 0.7614 - acc: 0.6966 - v
al_loss: 0.7121 - val_acc: 0.7654
Epoch 89/300
323/323 [==============================] - 0s 120us/step - loss: 0.7390 - acc: 0.7028 - v
al_loss: 0.7373 - val_acc: 0.7778
Epoch 90/300
323/323 [==============================] - 0s 121us/step - loss: 0.7103 - acc: 0.7028 - v
al_loss: 0.7495 - val_acc: 0.7654
Epoch 91/300
323/323 [==============================] - 0s 120us/step - loss: 0.7135 - acc: 0.7121 - v
al_loss: 0.7134 - val_acc: 0.7778
Epoch 92/300
323/323 [==============================] - 0s 123us/step - loss: 0.7096 - acc: 0.7307 - v
al_loss: 0.7003 - val_acc: 0.7778
Epoch 93/300
323/323 [==============================] - 0s 122us/step - loss: 0.7753 - acc: 0.7028 - v
al_loss: 0.7258 - val_acc: 0.7778
Epoch 94/300
323/323 [==============================] - 0s 121us/step - loss: 0.7336 - acc: 0.6842 - v
al_loss: 0.7363 - val_acc: 0.7778
Epoch 95/300
323/323 [==============================] - 0s 127us/step - loss: 0.7088 - acc: 0.7121 - v
al_loss: 0.6966 - val_acc: 0.7901
Epoch 96/300
323/323 [==============================] - 0s 126us/step - loss: 0.7005 - acc: 0.7276 - v
al_loss: 0.6868 - val_acc: 0.7778
Epoch 97/300
323/323 [==============================] - 0s 127us/step - loss: 0.7039 - acc: 0.7183 - v
al_loss: 0.7171 - val_acc: 0.7778
Epoch 98/300
323/323 [==============================] - 0s 122us/step - loss: 0.7130 - acc: 0.7090 - v
al_loss: 0.7046 - val_acc: 0.7531
Epoch 99/300
323/323 [==============================] - 0s 126us/step - loss: 0.7027 - acc: 0.7121 - v
al_loss: 0.6808 - val_acc: 0.7778
Epoch 100/300
323/323 [==============================] - 0s 127us/step - loss: 0.7141 - acc: 0.7059 - v
al_loss: 0.6843 - val_acc: 0.7778
Epoch 101/300
323/323 [==============================] - 0s 121us/step - loss: 0.6799 - acc: 0.7307 - v
al_loss: 0.7041 - val_acc: 0.7531
Epoch 102/300
323/323 [==============================] - 0s 121us/step - loss: 0.6908 - acc: 0.7059 - v
al_loss: 0.6980 - val_acc: 0.7778
Epoch 103/300
323/323 [==============================] - 0s 123us/step - loss: 0.7106 - acc: 0.7307 - v
al_loss: 0.7119 - val_acc: 0.7901
Epoch 104/300
323/323 [==============================] - 0s 124us/step - loss: 0.6688 - acc: 0.7121 - v
al_loss: 0.7420 - val_acc: 0.7654
Epoch 105/300
323/323 [==============================] - 0s 122us/step - loss: 0.7128 - acc: 0.7121 - v
al_loss: 0.7193 - val_acc: 0.7654
Epoch 106/300
323/323 [==============================] - 0s 125us/step - loss: 0.7076 - acc: 0.7276 - v
al_loss: 0.7145 - val_acc: 0.7654
Epoch 107/300
323/323 [==============================] - 0s 123us/step - loss: 0.6668 - acc: 0.7430 - v
al_loss: 0.7275 - val_acc: 0.7531
Epoch 108/300
323/323 [==============================] - 0s 124us/step - loss: 0.6897 - acc: 0.7183 - v
al_loss: 0.7084 - val_acc: 0.7901
Epoch 109/300
323/323 [==============================] - 0s 122us/step - loss: 0.6838 - acc: 0.7276 - v
al_loss: 0.7161 - val_acc: 0.7901
Epoch 110/300
323/323 [==============================] - 0s 120us/step - loss: 0.7109 - acc: 0.7121 - v
al_loss: 0.6978 - val_acc: 0.7654
Epoch 111/300
323/323 [==============================] - 0s 126us/step - loss: 0.6883 - acc: 0.7028 - v

```
al_loss: 0.6627 - val_acc: 0.7901
Epoch 112/300
323/323 [==============================] - 0s 128us/step - loss: 0.6544 - acc: 0.7307 - v
al_loss: 0.6569 - val_acc: 0.7654
Epoch 113/300
323/323 [==============================] - 0s 125us/step - loss: 0.6668 - acc: 0.7245 - v
al_loss: 0.6771 - val_acc: 0.7901
Epoch 114/300
323/323 [==============================] - 0s 122us/step - loss: 0.6719 - acc: 0.7152 - v
al_loss: 0.6656 - val_acc: 0.7901
Epoch 115/300
323/323 [==============================] - 0s 124us/step - loss: 0.6445 - acc: 0.7430 - v
al_loss: 0.6562 - val_acc: 0.7778
Epoch 116/300
323/323 [==============================] - 0s 125us/step - loss: 0.6807 - acc: 0.7276 - v
al_loss: 0.6772 - val_acc: 0.7778
Epoch 117/300
323/323 [==============================] - 0s 124us/step - loss: 0.6366 - acc: 0.7523 - v
al_loss: 0.6909 - val_acc: 0.7654
Epoch 118/300
323/323 [==============================] - 0s 124us/step - loss: 0.6411 - acc: 0.7337 - v
al_loss: 0.6972 - val_acc: 0.7654
Epoch 119/300
323/323 [==============================] - 0s 122us/step - loss: 0.6675 - acc: 0.7183 - v
al_loss: 0.6552 - val_acc: 0.7531
Epoch 120/300
323/323 [==============================] - 0s 125us/step - loss: 0.6746 - acc: 0.7276 - v
al_loss: 0.6793 - val_acc: 0.7654
Epoch 121/300
323/323 [==============================] - 0s 122us/step - loss: 0.6848 - acc: 0.7368 - v
al_loss: 0.7049 - val_acc: 0.7778
Epoch 122/300
323/323 [==============================] - 0s 124us/step - loss: 0.6833 - acc: 0.7028 - v
al_loss: 0.6508 - val_acc: 0.8148
Epoch 123/300
323/323 [==============================] - 0s 125us/step - loss: 0.6958 - acc: 0.7183 - v
al_loss: 0.6712 - val_acc: 0.7901
Epoch 124/300
323/323 [==============================] - 0s 127us/step - loss: 0.6614 - acc: 0.7307 - v
al_loss: 0.6984 - val_acc: 0.7407
Epoch 125/300
323/323 [==============================] - 0s 122us/step - loss: 0.6145 - acc: 0.7276 - v
al_loss: 0.6989 - val_acc: 0.7531
Epoch 126/300
323/323 [==============================] - 0s 120us/step - loss: 0.6715 - acc: 0.7245 - v
al_loss: 0.6992 - val_acc: 0.7407
Epoch 127/300
323/323 [==============================] - 0s 123us/step - loss: 0.6416 - acc: 0.7430 - v
al_loss: 0.6988 - val_acc: 0.7407
Epoch 128/300
323/323 [==============================] - 0s 123us/step - loss: 0.6231 - acc: 0.7307 - v
al_loss: 0.6695 - val_acc: 0.7654
Epoch 129/300
323/323 [==============================] - 0s 124us/step - loss: 0.6456 - acc: 0.7399 - v
al_loss: 0.6471 - val_acc: 0.7901
Epoch 130/300
323/323 [==============================] - 0s 125us/step - loss: 0.6358 - acc: 0.7399 - v
al_loss: 0.6723 - val_acc: 0.7531
Epoch 131/300
323/323 [==============================] - 0s 121us/step - loss: 0.6123 - acc: 0.7523 - v
al_loss: 0.6540 - val_acc: 0.7901
Epoch 132/300
323/323 [==============================] - 0s 123us/step - loss: 0.6037 - acc: 0.7523 - v
al_loss: 0.6598 - val_acc: 0.7654
Epoch 133/300
323/323 [==============================] - 0s 121us/step - loss: 0.6446 - acc: 0.7492 - v
al_loss: 0.6770 - val_acc: 0.7531
Epoch 134/300
323/323 [==============================] - 0s 119us/step - loss: 0.6441 - acc: 0.7307 - v
al_loss: 0.6967 - val_acc: 0.7407
Epoch 135/300
```

al_loss: 0.6622 - val_acc: 0.7778
Epoch 136/300
323/323 [==============================] - 0s 130us/step - loss: 0.6207 - acc: 0.7276 - v
al_loss: 0.6574 - val_acc: 0.7778
Epoch 137/300
323/323 [==============================] - 0s 133us/step - loss: 0.6301 - acc: 0.7245 - v
al_loss: 0.6311 - val_acc: 0.7901
Epoch 138/300
323/323 [==============================] - 0s 123us/step - loss: 0.6146 - acc: 0.7709 - v
al_loss: 0.6068 - val_acc: 0.7901
Epoch 139/300
323/323 [==============================] - 0s 125us/step - loss: 0.6161 - acc: 0.7554 - v
al_loss: 0.6463 - val_acc: 0.7531
Epoch 140/300
323/323 [==============================] - 0s 125us/step - loss: 0.6448 - acc: 0.7152 - v
al_loss: 0.6568 - val_acc: 0.7531
Epoch 141/300
323/323 [==============================] - 0s 124us/step - loss: 0.6133 - acc: 0.7399 - v
al_loss: 0.6351 - val_acc: 0.7531
Epoch 142/300
323/323 [==============================] - 0s 121us/step - loss: 0.6126 - acc: 0.7492 - v
al_loss: 0.6451 - val_acc: 0.7407
Epoch 143/300
323/323 [==============================] - 0s 128us/step - loss: 0.5890 - acc: 0.7492 - v
al_loss: 0.6372 - val_acc: 0.7531
Epoch 144/300
323/323 [==============================] - 0s 130us/step - loss: 0.6305 - acc: 0.7430 - v
al_loss: 0.6832 - val_acc: 0.7407
Epoch 145/300
323/323 [==============================] - 0s 125us/step - loss: 0.6327 - acc: 0.7368 - v
al_loss: 0.6509 - val_acc: 0.7284
Epoch 146/300
323/323 [==============================] - 0s 128us/step - loss: 0.5987 - acc: 0.7399 - v
al_loss: 0.6334 - val_acc: 0.7531
Epoch 147/300
323/323 [==============================] - 0s 125us/step - loss: 0.6295 - acc: 0.7276 - v
al_loss: 0.6480 - val_acc: 0.7778
Epoch 148/300
323/323 [==============================] - 0s 124us/step - loss: 0.6268 - acc: 0.7152 - v
al_loss: 0.6524 - val_acc: 0.7778
Epoch 149/300
323/323 [==============================] - 0s 125us/step - loss: 0.5905 - acc: 0.7616 - v
al_loss: 0.6273 - val_acc: 0.7901
Epoch 150/300
323/323 [==============================] - 0s 124us/step - loss: 0.6301 - acc: 0.7368 - v
al_loss: 0.6376 - val_acc: 0.7654
Epoch 151/300
323/323 [==============================] - 0s 128us/step - loss: 0.5970 - acc: 0.7647 - v
al_loss: 0.6530 - val_acc: 0.7531
Epoch 152/300
323/323 [==============================] - 0s 124us/step - loss: 0.5725 - acc: 0.7647 - v
al_loss: 0.6521 - val_acc: 0.7654
Epoch 153/300
323/323 [==============================] - 0s 122us/step - loss: 0.5978 - acc: 0.7523 - v
al_loss: 0.6783 - val_acc: 0.7531
Epoch 154/300
323/323 [==============================] - 0s 127us/step - loss: 0.5913 - acc: 0.7368 - v
al_loss: 0.6618 - val_acc: 0.7531
Epoch 155/300
323/323 [==============================] - 0s 125us/step - loss: 0.5696 - acc: 0.7678 - v
al_loss: 0.6401 - val_acc: 0.7778
Epoch 156/300
323/323 [==============================] - 0s 129us/step - loss: 0.5614 - acc: 0.7554 - v
al_loss: 0.6353 - val_acc: 0.7531
Epoch 157/300
323/323 [==============================] - 0s 131us/step - loss: 0.5858 - acc: 0.7523 - v
al_loss: 0.6289 - val_acc: 0.7654
Epoch 158/300
323/323 [==============================] - 0s 124us/step - loss: 0.6183 - acc: 0.7307 - v
al_loss: 0.6429 - val_acc: 0.7654
Epoch 159/300
323/323 [==============================] - 0s 125us/step - loss: 0.5790 - acc: 0.7864 - v

al_loss: 0.6577 - val_acc: 0.7531
Epoch 160/300
323/323 [==============================] - 0s 137us/step - loss: 0.6157 - acc: 0.7276 - v
al_loss: 0.6177 - val_acc: 0.7778
Epoch 161/300
323/323 [==============================] - 0s 126us/step - loss: 0.6076 - acc: 0.7461 - v
al_loss: 0.6485 - val_acc: 0.7778
Epoch 162/300
323/323 [==============================] - 0s 122us/step - loss: 0.6008 - acc: 0.7430 - v
al_loss: 0.6803 - val_acc: 0.7284
Epoch 163/300
323/323 [==============================] - 0s 123us/step - loss: 0.5958 - acc: 0.7368 - v
al_loss: 0.6095 - val_acc: 0.7778
Epoch 164/300
323/323 [==============================] - 0s 123us/step - loss: 0.6059 - acc: 0.7368 - v
al_loss: 0.6454 - val_acc: 0.7531
Epoch 165/300
323/323 [==============================] - 0s 121us/step - loss: 0.5813 - acc: 0.7647 - v
al_loss: 0.6549 - val_acc: 0.7778
Epoch 166/300
323/323 [==============================] - 0s 121us/step - loss: 0.5547 - acc: 0.7616 - v
al_loss: 0.6131 - val_acc: 0.7778
Epoch 167/300
323/323 [==============================] - 0s 129us/step - loss: 0.5872 - acc: 0.7709 - v
al_loss: 0.6134 - val_acc: 0.7654
Epoch 168/300
323/323 [==============================] - 0s 123us/step - loss: 0.5581 - acc: 0.7740 - v
al_loss: 0.6468 - val_acc: 0.7531
Epoch 169/300
323/323 [==============================] - 0s 120us/step - loss: 0.5540 - acc: 0.7771 - v
al_loss: 0.6644 - val_acc: 0.7654
Epoch 170/300
323/323 [==============================] - 0s 128us/step - loss: 0.5628 - acc: 0.7740 - v
al_loss: 0.6475 - val_acc: 0.7654
Epoch 171/300
323/323 [==============================] - 0s 127us/step - loss: 0.5680 - acc: 0.7771 - v
al_loss: 0.6334 - val_acc: 0.7531
Epoch 172/300
323/323 [==============================] - 0s 129us/step - loss: 0.5437 - acc: 0.7740 - v
al_loss: 0.6429 - val_acc: 0.7654
Epoch 173/300
323/323 [==============================] - 0s 121us/step - loss: 0.5224 - acc: 0.7957 - v
al_loss: 0.6296 - val_acc: 0.7654
Epoch 174/300
323/323 [==============================] - 0s 125us/step - loss: 0.5542 - acc: 0.7709 - v
al_loss: 0.6285 - val_acc: 0.7778
Epoch 175/300
323/323 [==============================] - 0s 123us/step - loss: 0.5634 - acc: 0.7709 - v
al_loss: 0.6335 - val_acc: 0.7654
Epoch 176/300
323/323 [==============================] - 0s 126us/step - loss: 0.5508 - acc: 0.7833 - v
al_loss: 0.6438 - val_acc: 0.7531
Epoch 177/300
323/323 [==============================] - 0s 123us/step - loss: 0.5867 - acc: 0.7709 - v
al_loss: 0.6488 - val_acc: 0.7531
Epoch 178/300
323/323 [==============================] - 0s 123us/step - loss: 0.5634 - acc: 0.7740 - v
al_loss: 0.6420 - val_acc: 0.7654
Epoch 179/300
323/323 [==============================] - 0s 125us/step - loss: 0.5363 - acc: 0.7740 - v
al_loss: 0.6344 - val_acc: 0.7531
Epoch 180/300
323/323 [==============================] - 0s 126us/step - loss: 0.5145 - acc: 0.7771 - v
al_loss: 0.6492 - val_acc: 0.7531
Epoch 181/300
323/323 [==============================] - 0s 122us/step - loss: 0.5628 - acc: 0.7523 - v
al_loss: 0.6661 - val_acc: 0.7531
Epoch 182/300
323/323 [==============================] - 0s 122us/step - loss: 0.5435 - acc: 0.7771 - v
al_loss: 0.6808 - val_acc: 0.7284
Epoch 183/300
323/323 [==============================] - 0s 127us/step - loss: 0.5573 - acc: 0.7585 - v

```
al_loss: 0.6171 - val_acc: 0.7778
Epoch 184/300
323/323 [==============================] - 0s 129us/step - loss: 0.5431 - acc: 0.7616 - v
al_loss: 0.6049 - val_acc: 0.7778
Epoch 185/300
323/323 [==============================] - 0s 130us/step - loss: 0.5537 - acc: 0.7833 - v
al_loss: 0.6859 - val_acc: 0.7284
Epoch 186/300
323/323 [==============================] - 0s 127us/step - loss: 0.5592 - acc: 0.7895 - v
al_loss: 0.6779 - val_acc: 0.7654
Epoch 187/300
323/323 [==============================] - 0s 126us/step - loss: 0.5467 - acc: 0.7771 - v
al_loss: 0.6541 - val_acc: 0.7531
Epoch 188/300
323/323 [==============================] - 0s 120us/step - loss: 0.5702 - acc: 0.7647 - v
al_loss: 0.6618 - val_acc: 0.7407
Epoch 189/300
323/323 [==============================] - 0s 120us/step - loss: 0.5438 - acc: 0.7647 - v
al_loss: 0.6800 - val_acc: 0.7531
Epoch 190/300
323/323 [==============================] - 0s 131us/step - loss: 0.4828 - acc: 0.8050 - v
al_loss: 0.6861 - val_acc: 0.7654
Epoch 191/300
323/323 [==============================] - 0s 132us/step - loss: 0.5250 - acc: 0.7616 - v
al_loss: 0.7257 - val_acc: 0.7284
Epoch 192/300
323/323 [==============================] - 0s 125us/step - loss: 0.5722 - acc: 0.7678 - v
al_loss: 0.7038 - val_acc: 0.7407
Epoch 193/300
323/323 [==============================] - 0s 125us/step - loss: 0.5451 - acc: 0.7523 - v
al_loss: 0.6919 - val_acc: 0.7531
Epoch 194/300
323/323 [==============================] - 0s 126us/step - loss: 0.5298 - acc: 0.7957 - v
al_loss: 0.6408 - val_acc: 0.7654
Epoch 195/300
323/323 [==============================] - 0s 121us/step - loss: 0.5279 - acc: 0.7709 - v
al_loss: 0.6357 - val_acc: 0.7654
Epoch 196/300
323/323 [==============================] - 0s 129us/step - loss: 0.5595 - acc: 0.7988 - v
al_loss: 0.6287 - val_acc: 0.7654
Epoch 197/300
323/323 [==============================] - 0s 124us/step - loss: 0.5257 - acc: 0.7647 - v
al_loss: 0.6924 - val_acc: 0.7407
Epoch 198/300
323/323 [==============================] - 0s 121us/step - loss: 0.5588 - acc: 0.7616 - v
al_loss: 0.7579 - val_acc: 0.6914
Epoch 199/300
323/323 [==============================] - 0s 125us/step - loss: 0.5465 - acc: 0.7709 - v
al_loss: 0.7032 - val_acc: 0.7160
Epoch 200/300
323/323 [==============================] - 0s 124us/step - loss: 0.5266 - acc: 0.7833 - v
al_loss: 0.6633 - val_acc: 0.7531
Epoch 201/300
323/323 [==============================] - 0s 128us/step - loss: 0.5161 - acc: 0.7771 - v
al_loss: 0.6606 - val_acc: 0.7284
Epoch 202/300
323/323 [==============================] - 0s 124us/step - loss: 0.5828 - acc: 0.7647 - v
al_loss: 0.6213 - val_acc: 0.7778
Epoch 203/300
323/323 [==============================] - 0s 127us/step - loss: 0.5570 - acc: 0.7709 - v
al_loss: 0.5979 - val_acc: 0.7778
Epoch 204/300
323/323 [==============================] - 0s 128us/step - loss: 0.5487 - acc: 0.7678 - v
al_loss: 0.6628 - val_acc: 0.7407
Epoch 205/300
323/323 [==============================] - 0s 128us/step - loss: 0.5160 - acc: 0.8111 - v
al_loss: 0.6656 - val_acc: 0.7407
Epoch 206/300
323/323 [==============================] - 0s 122us/step - loss: 0.5007 - acc: 0.7709 - v
al_loss: 0.6496 - val_acc: 0.7531
Epoch 207/300
323/323 [==============================] - 0s 123us/step - loss: 0.5209 - acc: 0.7802 - v
```

al_loss: 0.6378 - val_acc: 0.7531
Epoch 208/300
323/323 [==============================] - 0s 124us/step - loss: 0.5002 - acc: 0.7988 - v
al_loss: 0.6174 - val_acc: 0.7654
Epoch 209/300
323/323 [==============================] - 0s 131us/step - loss: 0.5054 - acc: 0.7926 - v
al_loss: 0.6389 - val_acc: 0.7284
Epoch 210/300
323/323 [==============================] - 0s 125us/step - loss: 0.5214 - acc: 0.7740 - v
al_loss: 0.6766 - val_acc: 0.7037
Epoch 211/300
323/323 [==============================] - 0s 124us/step - loss: 0.5252 - acc: 0.7740 - v
al_loss: 0.6919 - val_acc: 0.7407
Epoch 212/300
323/323 [==============================] - 0s 123us/step - loss: 0.5023 - acc: 0.7895 - v
al_loss: 0.6979 - val_acc: 0.7654
Epoch 213/300
323/323 [==============================] - 0s 120us/step - loss: 0.5122 - acc: 0.7802 - v
al_loss: 0.6836 - val_acc: 0.7531
Epoch 214/300
323/323 [==============================] - 0s 121us/step - loss: 0.5021 - acc: 0.7802 - v
al_loss: 0.6327 - val_acc: 0.7654
Epoch 215/300
323/323 [==============================] - 0s 121us/step - loss: 0.5075 - acc: 0.7926 - v
al_loss: 0.6316 - val_acc: 0.7654
Epoch 216/300
323/323 [==============================] - 0s 126us/step - loss: 0.5206 - acc: 0.8050 - v
al_loss: 0.6579 - val_acc: 0.7531
Epoch 217/300
323/323 [==============================] - 0s 120us/step - loss: 0.4993 - acc: 0.7771 - v
al_loss: 0.6453 - val_acc: 0.7531
Epoch 218/300
323/323 [==============================] - 0s 126us/step - loss: 0.4855 - acc: 0.8019 - v
al_loss: 0.6569 - val_acc: 0.7407
Epoch 219/300
323/323 [==============================] - 0s 125us/step - loss: 0.4921 - acc: 0.7833 - v
al_loss: 0.6460 - val_acc: 0.7407
Epoch 220/300
323/323 [==============================] - 0s 127us/step - loss: 0.4932 - acc: 0.7895 - v
al_loss: 0.6319 - val_acc: 0.7407
Epoch 221/300
323/323 [==============================] - 0s 124us/step - loss: 0.5226 - acc: 0.7864 - v
al_loss: 0.6566 - val_acc: 0.7407
Epoch 222/300
323/323 [==============================] - 0s 124us/step - loss: 0.5274 - acc: 0.7678 - v
al_loss: 0.6622 - val_acc: 0.7160
Epoch 223/300
323/323 [==============================] - 0s 123us/step - loss: 0.4663 - acc: 0.8080 - v
al_loss: 0.6080 - val_acc: 0.7407
Epoch 224/300
323/323 [==============================] - 0s 131us/step - loss: 0.5380 - acc: 0.7678 - v
al_loss: 0.6346 - val_acc: 0.7407
Epoch 225/300
323/323 [==============================] - 0s 128us/step - loss: 0.5104 - acc: 0.7740 - v
al_loss: 0.6917 - val_acc: 0.7160
Epoch 226/300
323/323 [==============================] - 0s 125us/step - loss: 0.4876 - acc: 0.7988 - v
al_loss: 0.6691 - val_acc: 0.7284
Epoch 227/300
323/323 [==============================] - 0s 124us/step - loss: 0.5102 - acc: 0.7895 - v
al_loss: 0.6808 - val_acc: 0.6790
Epoch 228/300
323/323 [==============================] - 0s 125us/step - loss: 0.4878 - acc: 0.8111 - v
al_loss: 0.6672 - val_acc: 0.7407
Epoch 229/300
323/323 [==============================] - 0s 125us/step - loss: 0.4981 - acc: 0.7833 - v
al_loss: 0.6265 - val_acc: 0.7531
Epoch 230/300
323/323 [==============================] - 0s 120us/step - loss: 0.4967 - acc: 0.7864 - v
al_loss: 0.5946 - val_acc: 0.7531
Epoch 231/300
323/323 [==============================] - 0s 126us/step - loss: 0.4912 - acc: 0.7740 - v

al_loss: 0.6189 - val_acc: 0.7407
Epoch 232/300
323/323 [==============================] - 0s 124us/step - loss: 0.4829 - acc: 0.7678 - v
al_loss: 0.6128 - val_acc: 0.7654
Epoch 233/300
323/323 [==============================] - 0s 130us/step - loss: 0.5416 - acc: 0.7771 - v
al_loss: 0.6077 - val_acc: 0.7531
Epoch 234/300
323/323 [==============================] - 0s 124us/step - loss: 0.4623 - acc: 0.7926 - v
al_loss: 0.6325 - val_acc: 0.7284
Epoch 235/300
323/323 [==============================] - 0s 127us/step - loss: 0.4631 - acc: 0.8111 - v
al_loss: 0.6571 - val_acc: 0.7407
Epoch 236/300
323/323 [==============================] - 0s 126us/step - loss: 0.4687 - acc: 0.8173 - v
al_loss: 0.6393 - val_acc: 0.7407
Epoch 237/300
323/323 [==============================] - 0s 130us/step - loss: 0.4585 - acc: 0.8019 - v
al_loss: 0.6311 - val_acc: 0.7407
Epoch 238/300
323/323 [==============================] - 0s 124us/step - loss: 0.4858 - acc: 0.7926 - v
al_loss: 0.6314 - val_acc: 0.7407
Epoch 239/300
323/323 [==============================] - 0s 126us/step - loss: 0.4784 - acc: 0.8019 - v
al_loss: 0.6625 - val_acc: 0.7407
Epoch 240/300
323/323 [==============================] - 0s 128us/step - loss: 0.4615 - acc: 0.8080 - v
al_loss: 0.7009 - val_acc: 0.7037
Epoch 241/300
323/323 [==============================] - 0s 129us/step - loss: 0.4721 - acc: 0.7988 - v
al_loss: 0.7230 - val_acc: 0.7037
Epoch 242/300
323/323 [==============================] - 0s 130us/step - loss: 0.5147 - acc: 0.7895 - v
al_loss: 0.6722 - val_acc: 0.7037
Epoch 243/300
323/323 [==============================] - 0s 125us/step - loss: 0.4522 - acc: 0.8266 - v
al_loss: 0.6371 - val_acc: 0.7407
Epoch 244/300
323/323 [==============================] - 0s 130us/step - loss: 0.4825 - acc: 0.7771 - v
al_loss: 0.6225 - val_acc: 0.7407
Epoch 245/300
323/323 [==============================] - 0s 123us/step - loss: 0.4709 - acc: 0.7833 - v
al_loss: 0.6370 - val_acc: 0.7407
Epoch 246/300
323/323 [==============================] - 0s 125us/step - loss: 0.4535 - acc: 0.8452 - v
al_loss: 0.6282 - val_acc: 0.7284
Epoch 247/300
323/323 [==============================] - 0s 124us/step - loss: 0.4671 - acc: 0.7926 - v
al_loss: 0.6176 - val_acc: 0.7284
Epoch 248/300
323/323 [==============================] - 0s 125us/step - loss: 0.4555 - acc: 0.7926 - v
al_loss: 0.6357 - val_acc: 0.7407
Epoch 249/300
323/323 [==============================] - 0s 128us/step - loss: 0.4695 - acc: 0.7895 - v
al_loss: 0.6539 - val_acc: 0.7160
Epoch 250/300
323/323 [==============================] - 0s 124us/step - loss: 0.4344 - acc: 0.8204 - v
al_loss: 0.6398 - val_acc: 0.7160
Epoch 251/300
323/323 [==============================] - 0s 121us/step - loss: 0.4613 - acc: 0.7926 - v
al_loss: 0.6101 - val_acc: 0.7407
Epoch 252/300
323/323 [==============================] - 0s 127us/step - loss: 0.4508 - acc: 0.7926 - v
al_loss: 0.5917 - val_acc: 0.7531
Epoch 253/300
323/323 [==============================] - 0s 123us/step - loss: 0.4328 - acc: 0.8080 - v
al_loss: 0.5942 - val_acc: 0.7654
Epoch 254/300
323/323 [==============================] - 0s 121us/step - loss: 0.4468 - acc: 0.8080 - v
al_loss: 0.5973 - val_acc: 0.7654
Epoch 255/300
323/323 [==============================] - 0s 124us/step - loss: 0.4604 - acc: 0.8111 - v

al_loss: 0.6198 - val_acc: 0.7654
Epoch 256/300
323/323 [==============================] - 0s 121us/step - loss: 0.4539 - acc: 0.8173 - v
al_loss: 0.6350 - val_acc: 0.7531
Epoch 257/300
323/323 [==============================] - 0s 127us/step - loss: 0.4523 - acc: 0.7957 - v
al_loss: 0.6540 - val_acc: 0.7284
Epoch 258/300
323/323 [==============================] - 0s 125us/step - loss: 0.4620 - acc: 0.8204 - v
al_loss: 0.6158 - val_acc: 0.7407
Epoch 259/300
323/323 [==============================] - 0s 125us/step - loss: 0.4649 - acc: 0.7833 - v
al_loss: 0.5940 - val_acc: 0.7407
Epoch 260/300
323/323 [==============================] - 0s 122us/step - loss: 0.4655 - acc: 0.8080 - v
al_loss: 0.6050 - val_acc: 0.7160
Epoch 261/300
323/323 [==============================] - 0s 123us/step - loss: 0.4740 - acc: 0.8080 - v
al_loss: 0.5972 - val_acc: 0.7531
Epoch 262/300
323/323 [==============================] - 0s 123us/step - loss: 0.4256 - acc: 0.7988 - v
al_loss: 0.6965 - val_acc: 0.7037
Epoch 263/300
323/323 [==============================] - 0s 126us/step - loss: 0.4819 - acc: 0.7988 - v
al_loss: 0.6429 - val_acc: 0.7284
Epoch 264/300
323/323 [==============================] - 0s 131us/step - loss: 0.4546 - acc: 0.8019 - v
al_loss: 0.6419 - val_acc: 0.7284
Epoch 265/300
323/323 [==============================] - 0s 124us/step - loss: 0.4408 - acc: 0.8142 - v
al_loss: 0.7281 - val_acc: 0.6543
Epoch 266/300
323/323 [==============================] - 0s 127us/step - loss: 0.4351 - acc: 0.8359 - v
al_loss: 0.6063 - val_acc: 0.7531
Epoch 267/300
323/323 [==============================] - 0s 128us/step - loss: 0.4952 - acc: 0.7895 - v
al_loss: 0.6063 - val_acc: 0.7407
Epoch 268/300
323/323 [==============================] - 0s 123us/step - loss: 0.4491 - acc: 0.8204 - v
al_loss: 0.6461 - val_acc: 0.6914
Epoch 269/300
323/323 [==============================] - 0s 125us/step - loss: 0.4704 - acc: 0.7802 - v
al_loss: 0.5752 - val_acc: 0.7654
Epoch 270/300
323/323 [==============================] - 0s 123us/step - loss: 0.4526 - acc: 0.7864 - v
al_loss: 0.6002 - val_acc: 0.7531
Epoch 271/300
323/323 [==============================] - 0s 124us/step - loss: 0.4502 - acc: 0.7926 - v
al_loss: 0.6411 - val_acc: 0.7407
Epoch 272/300
323/323 [==============================] - 0s 123us/step - loss: 0.4600 - acc: 0.8142 - v
al_loss: 0.6251 - val_acc: 0.7407
Epoch 273/300
323/323 [==============================] - 0s 121us/step - loss: 0.4363 - acc: 0.8173 - v
al_loss: 0.6217 - val_acc: 0.7407
Epoch 274/300
323/323 [==============================] - 0s 124us/step - loss: 0.4439 - acc: 0.7957 - v
al_loss: 0.6116 - val_acc: 0.7284
Epoch 275/300
323/323 [==============================] - 0s 126us/step - loss: 0.4475 - acc: 0.8142 - v
al_loss: 0.5703 - val_acc: 0.7531
Epoch 276/300
323/323 [==============================] - 0s 132us/step - loss: 0.4179 - acc: 0.8235 - v
al_loss: 0.5540 - val_acc: 0.7778
Epoch 277/300
323/323 [==============================] - 0s 128us/step - loss: 0.4450 - acc: 0.8204 - v
al_loss: 0.5463 - val_acc: 0.7778
Epoch 278/300
323/323 [==============================] - 0s 127us/step - loss: 0.3898 - acc: 0.8359 - v
al_loss: 0.5615 - val_acc: 0.7654
Epoch 279/300
323/323 [==============================] - 0s 126us/step - loss: 0.4648 - acc: 0.8204 - v

```
al_loss: 0.5603 - val_acc: 0.7531
Epoch 280/300
323/323 [==============================] - 0s 129us/step - loss: 0.4050 - acc: 0.8359 - v
al_loss: 0.6089 - val_acc: 0.7407
Epoch 281/300
323/323 [==============================] - 0s 131us/step - loss: 0.4294 - acc: 0.8266 - v
al_loss: 0.6379 - val_acc: 0.7160
Epoch 282/300
323/323 [==============================] - 0s 123us/step - loss: 0.4134 - acc: 0.8297 - v
al_loss: 0.6403 - val_acc: 0.7160
Epoch 283/300
323/323 [==============================] - 0s 133us/step - loss: 0.4247 - acc: 0.8235 - v
al_loss: 0.6405 - val_acc: 0.7284
Epoch 284/300
323/323 [==============================] - 0s 126us/step - loss: 0.4207 - acc: 0.8204 - v
al_loss: 0.6314 - val_acc: 0.7284
Epoch 285/300
323/323 [==============================] - 0s 124us/step - loss: 0.4243 - acc: 0.8050 - v
al_loss: 0.6086 - val_acc: 0.7160
Epoch 286/300
323/323 [==============================] - 0s 121us/step - loss: 0.3893 - acc: 0.8390 - v
al_loss: 0.5855 - val_acc: 0.7407
Epoch 287/300
323/323 [==============================] - 0s 126us/step - loss: 0.3754 - acc: 0.8204 - v
al_loss: 0.5887 - val_acc: 0.7407
Epoch 288/300
323/323 [==============================] - 0s 120us/step - loss: 0.4308 - acc: 0.8173 - v
al_loss: 0.6052 - val_acc: 0.7531
Epoch 289/300
323/323 [==============================] - 0s 125us/step - loss: 0.4241 - acc: 0.8173 - v
al_loss: 0.5261 - val_acc: 0.7654
Epoch 290/300
323/323 [==============================] - 0s 125us/step - loss: 0.3897 - acc: 0.8421 - v
al_loss: 0.5350 - val_acc: 0.7778
Epoch 291/300
323/323 [==============================] - 0s 125us/step - loss: 0.3894 - acc: 0.8483 - v
al_loss: 0.5703 - val_acc: 0.7654
Epoch 292/300
323/323 [==============================] - 0s 129us/step - loss: 0.4223 - acc: 0.8111 - v
al_loss: 0.5838 - val_acc: 0.7531
Epoch 293/300
323/323 [==============================] - 0s 125us/step - loss: 0.4301 - acc: 0.8266 - v
al_loss: 0.5565 - val_acc: 0.7531
Epoch 294/300
323/323 [==============================] - 0s 126us/step - loss: 0.4149 - acc: 0.8173 - v
al_loss: 0.5855 - val_acc: 0.7531
Epoch 295/300
323/323 [==============================] - 0s 124us/step - loss: 0.4138 - acc: 0.8204 - v
al_loss: 0.6246 - val_acc: 0.7531
Epoch 296/300
323/323 [==============================] - 0s 125us/step - loss: 0.4306 - acc: 0.8421 - v
al_loss: 0.6121 - val_acc: 0.7407
Epoch 297/300
323/323 [==============================] - 0s 127us/step - loss: 0.3968 - acc: 0.8359 - v
al_loss: 0.5806 - val_acc: 0.7654
Epoch 298/300
323/323 [==============================] - 0s 126us/step - loss: 0.3778 - acc: 0.8452 - v
al_loss: 0.6454 - val_acc: 0.7160
Epoch 299/300
323/323 [==============================] - 0s 126us/step - loss: 0.4297 - acc: 0.8111 - v
al_loss: 0.5511 - val_acc: 0.7654
Epoch 300/300
323/323 [==============================] - 0s 122us/step - loss: 0.3965 - acc: 0.8297 - v
al_loss: 0.5282 - val_acc: 0.7778
CPU times: user 15.8 s, sys: 2.95 s, total: 18.8 s
Wall time: 16 s
```

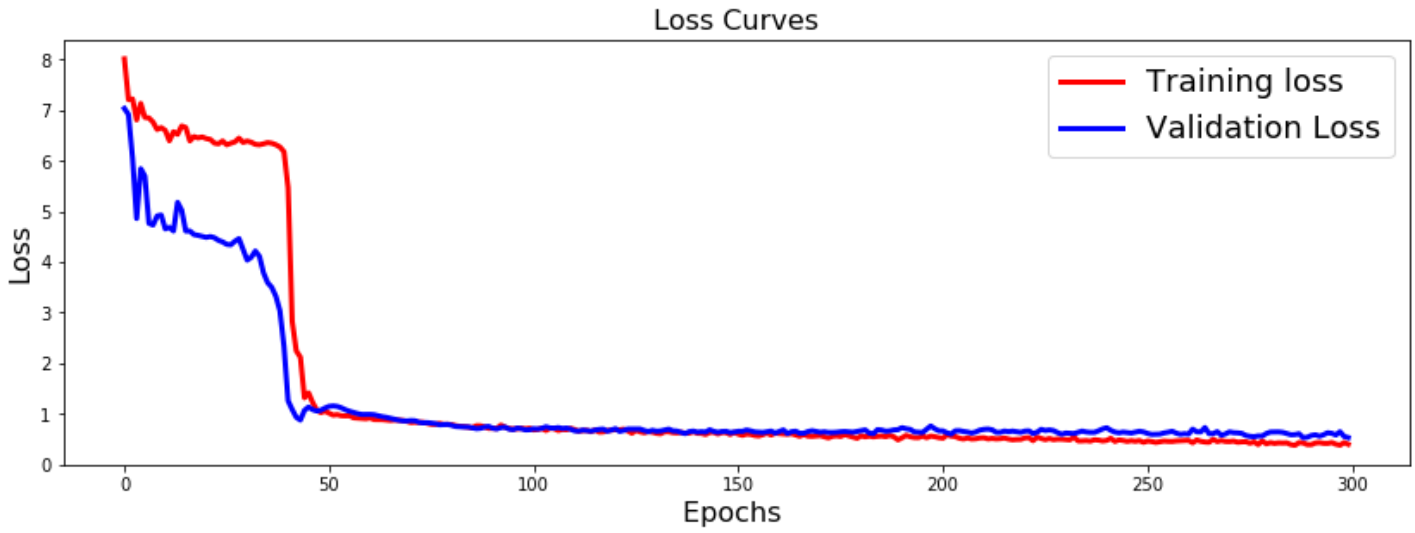
In [15]:

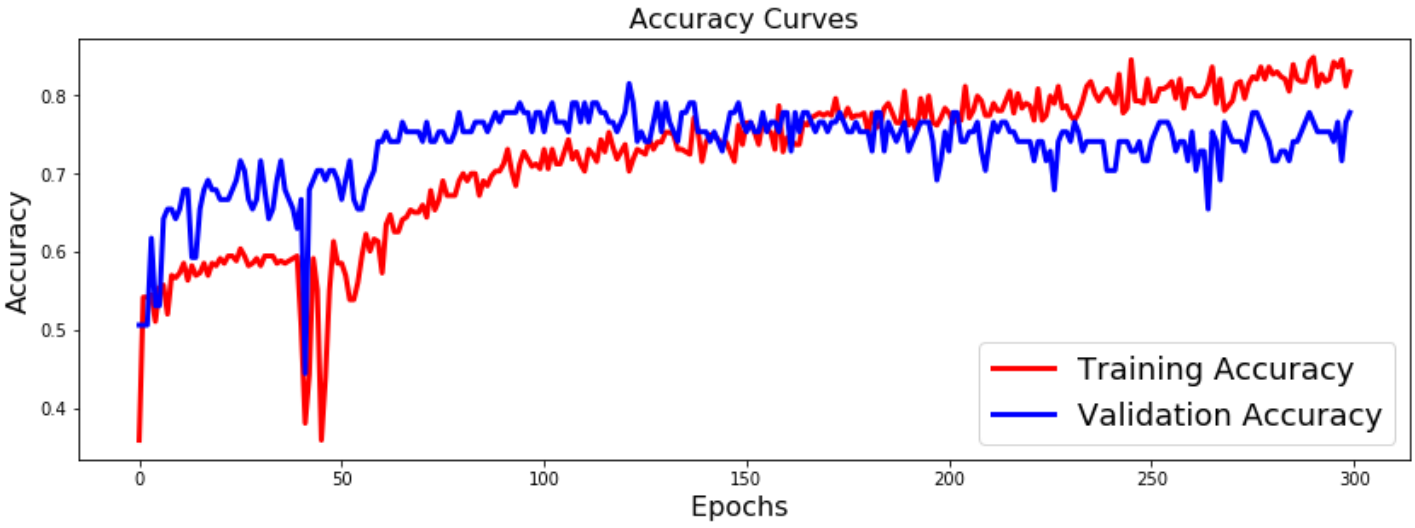
```
# Loss Curves
plt.figure(figsize=[14,10])
```

```
plt.subplot(211)
plt.plot(history.history['loss'],'r',linewidth=3.0)
plt.plot(history.history['val_loss'],'b',linewidth=3.0)
plt.legend(['Training loss', 'Validation Loss'],fontsize=18)
plt.xlabel('Epochs ',fontsize=16)
plt.ylabel('Loss',fontsize=16)
plt.title('Loss Curves',fontsize=16)

# Accuracy Curves
plt.figure(figsize=[14,10])
plt.subplot(212)
plt.plot(history.history['acc'],'r',linewidth=3.0)
plt.plot(history.history['val_acc'],'b',linewidth=3.0)
plt.legend(['Training Accuracy', 'Validation Accuracy'],fontsize=18)
plt.xlabel('Epochs ',fontsize=16)
plt.ylabel('Accuracy',fontsize=16)
plt.title('Accuracy Curves',fontsize=16)
```

Out[15]:

```
Text(0.5, 1.0, 'Accuracy Curves')
```





Save model

In [16]:

```
# Save model and weights
model_name = "heartbeat_disease.h5"
model.save(model_name)
print('Saved trained model at %s ' % model_name)
```

```
Saved trained model at heartbeat_disease.h5
```

Evaluate model

Evaluate model

In [17]:

```
scores = model.evaluate(x_test, y_test, verbose=1)
print('Test loss:', scores[0])
print('Test accuracy:', scores[1])
```

```
81/81 [==============================] - 0s 335us/step
Test loss: 0.5282161000334187
Test accuracy: 0.7777777807212171
```

Classification Report

In [18]:

```
predictions = model.predict(x_test, verbose=1)
```

```
81/81 [==============================] - 0s 1ms/step
```

In [19]:

```
y_true, y_pred = [],[]
classes = encoder.classes_
for idx, prediction in enumerate(predictions):
    y_true.append(classes[np.argmax(y_test[idx])])
    y_pred.append(classes[np.argmax(prediction)])
```

In [20]:

```
print(classification_report(y_pred, y_true))
```

```
              precision    recall  f1-score   support

    artifact       0.88      1.00      0.94        15
    extrahls       0.67      0.33      0.44         6
  extrastole       0.00      0.00      0.00         1
      murmur       0.78      0.70      0.74        20
      normal       0.78      0.82      0.80        39

   micro avg       0.78      0.78      0.78        81
   macro avg       0.62      0.57      0.58        81
weighted avg       0.78      0.78      0.77        81
```

Confusion Matrix

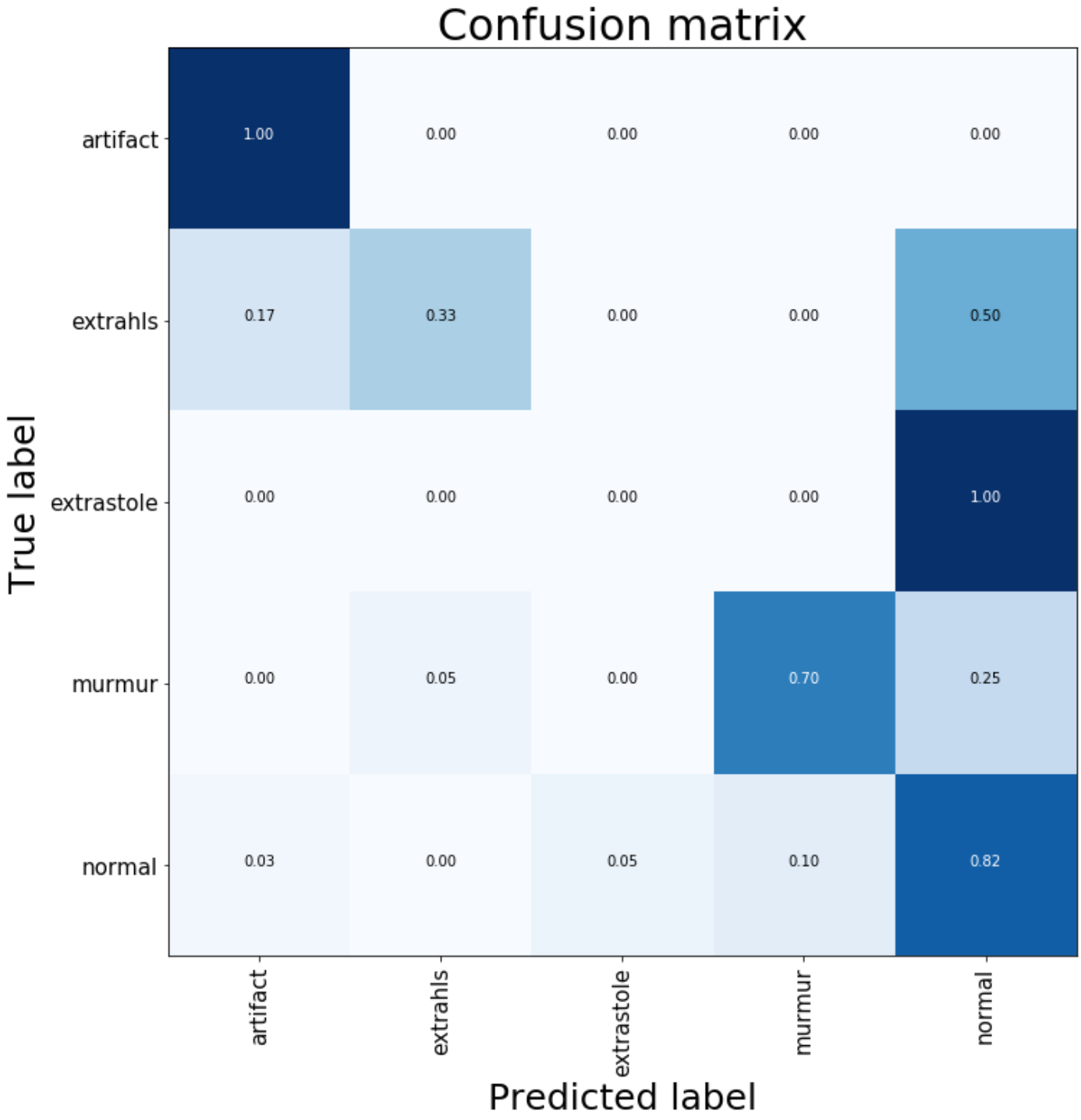
In [21]:

```
def plot_confusion_matrix(cm, classes,
                          title='Confusion matrix',
                          cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    plt.figure(figsize=(11, 11))
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title, fontsize=30)
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=90, fontsize=15)
    plt.yticks(tick_marks, classes, fontsize=15)

    fmt = '.2f'
    thresh = cm.max() / 2.
    for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
        plt.text(j, i, format(cm[i, j], fmt),
                 horizontalalignment="center",
                 color="white" if cm[i, j] > thresh else "black")
```

```
    plt.ylabel('True label', fontsize=25)
    plt.xlabel('Predicted label', fontsize=25)
    plt.tight_layout()

    plt.show()
```

In [22]:

```
cnf_matrix = confusion_matrix(y_pred, y_true)
cnf_matrix = cnf_matrix.astype(float) / cnf_matrix.sum(axis=1)[:, np.newaxis]
plot_confusion_matrix(cnf_matrix, classes)
```



In [23]: