In [1]:
```python
# Работающая нейросеть

# Импорт
from __future__ import absolute_import, division, print_function, unicode_literals
import numpy as np
import pandas as pd
import tensorflow as tf
from tensorflow import feature_column
from tensorflow.keras import layers
```

In [2]:
```python
# Пути для тренировочных файлов
path1 = r"./har_dataset/train/file.txt"
path2 = r"./har_dataset/train/outfile.txt"
path3 = r"./har_dataset/train/new_train.txt" # Итоговый файл с тренировочными данными

# Пути для тестовых данных
path4 = r"./har_dataset/test/file.txt"
path5 = r"./har_dataset/test/outfile.txt"
path6 = r"./har_dataset/test/new_test.txt" # Итоговый файл с тестовыми данными

# Путь до списка признаков
path_features_file = r"./har_dataset/features.txt"

# Пути к файлам лейблов
path_train_labels = r"./har_dataset/train/y_train.txt"
path_test_labels = r"./har_dataset/test/y_test.txt"
final_train_labels = r"./har_dataset/train/y_train_fin.txt"
final_test_labels = r"./har_dataset/test/y_test_fin.txt"
```

In [3]:
```python
# Дополнительные функции для обработки файлов

# Функция удаления лишних символов из входного файла
def cleaning_func(path_file, out_path_file, new_path_file):
    fin = open(path_file, 'r')
    fout = open(out_path_file, 'w')
    lines = fin.readlines()
    for line in lines:
        fout.write(line[2:])
    fout.close()
    fin.close()
    fout = open(out_path_file, 'r')
    new = open(new_path_file, 'w')
    lines = fout.readlines()
    for line in lines:
        new.write(" ".join(line.split()) + "\n")
    fout.close()
    new.close()

# Функция, печатающая список признаков из файла
def features_string(path_features_file):
    features_dict = {}
    with open(path_features_file) as file:
        for line in file:
            key, *value = line.split()
            features_dict[key] = value
    string = ""
    for i in range(1, 562):
        string = string + (features_dict[str(i)])[0] + " "
    return(string)

# Функция уменьшения лейблов
# (переход от системы 1-6 к 0-5 системе)
def labels_decrease(in_lbl, out_lbl):
    f_in = open(in_lbl, 'r')
    f_out = open(out_lbl, 'w')
    lines = f_in.readlines()
    for line in lines:
        f_out.write(str(int(line)-1) + '\n')
    f_out.close()
    f_in.close()
```

In [22]:
```python
# Вызов функций с аргументами путей до файлов
cleaning_func(path1, path2, path3)
cleaning_func(path4, path5, path6)
print(features_print_func(path_features_file))
labels_decrease(path_train_labels, final_train_labels)
labels_decrease(path_test_labels, final_test_labels)
```

```
tBodyAcc-mean()-X tBodyAcc-mean()-Y tBodyAcc-mean()-Z tBodyAcc-std()-X tBodyAcc-std()-Y tBodyAcc-std()-Z tBodyAcc-mad()-X tBody
Acc-mad()-Y tBodyAcc-mad()-Z tBodyAcc-max()-X tBodyAcc-max()-Y tBodyAcc-max()-Z tBodyAcc-min()-X tBodyAcc-min()-Y tBodyAcc-min
()-Z tBodyAcc-sma() tBodyAcc-energy()-X tBodyAcc-energy()-Y tBodyAcc-energy()-Z tBodyAcc-iqr()-X tBodyAcc-iqr()-Y tBodyAcc-iqr
()-Z tBodyAcc-entropy()-X tBodyAcc-entropy()-Y tBodyAcc-entropy()-Z tBodyAcc-arCoeff()-X,1 tBodyAcc-arCoeff()-X,2 tBodyAcc-arCo
eff()-X,3 tBodyAcc-arCoeff()-X,4 tBodyAcc-arCoeff()-Y,1 tBodyAcc-arCoeff()-Y,2 tBodyAcc-arCoeff()-Y,3 tBodyAcc-arCoeff()-Y,4 tB
odyAcc-arCoeff()-Z,1 tBodyAcc-arCoeff()-Z,2 tBodyAcc-arCoeff()-Z,3 tBodyAcc-arCoeff()-Z,4 tBodyAcc-correlation()-X,Y tBodyAcc-c
orrelation()-X,Z tBodyAcc-correlation()-Y,Z tGravityAcc-mean()-X tGravityAcc-mean()-Y tGravityAcc-mean()-Z tGravityAcc-std()-X
tGravityAcc-std()-Y tGravityAcc-std()-Z tGravityAcc-mad()-X tGravityAcc-mad()-Y tGravityAcc-mad()-Z tGravityAcc-max()-X tGravit
yAcc-max()-Y tGravityAcc-max()-Z tGravityAcc-min()-X tGravityAcc-min()-Y tGravityAcc-min()-Z tGravityAcc-sma() tGravityAcc-ener
gy()-X tGravityAcc-energy()-Y tGravityAcc-energy()-Z tGravityAcc-iqr()-X tGravityAcc-iqr()-Y tGravityAcc-iqr()-Z tGravityAcc-en
tropy()-X tGravityAcc-entropy()-Y tGravityAcc-entropy()-Z tGravityAcc-arCoeff()-X,1 tGravityAcc-arCoeff()-X,2 tGravityAcc-arCoe
ff()-X,3 tGravityAcc-arCoeff()-X,4 tGravityAcc-arCoeff()-Y,1 tGravityAcc-arCoeff()-Y,2 tGravityAcc-arCoeff()-Y,3 tGravityAcc-ar
Coeff()-Y,4 tGravityAcc-arCoeff()-Z,1 tGravityAcc-arCoeff()-Z,2 tGravityAcc-arCoeff()-Z,3 tGravityAcc-arCoeff()-Z,4 tGravityAcc
-correlation()-X,Y tGravityAcc-correlation()-X,Z tGravityAcc-correlation()-Y,Z tBodyAccJerk-mean()-X tBodyAccJerk-mean()-Y tBod
yAccJerk-mean()-Z tBodyAccJerk-std()-X tBodyAccJerk-std()-Y tBodyAccJerk-std()-Z tBodyAccJerk-mad()-X tBodyAccJerk-mad()-Y tBod
yAccJerk-mad()-Z tBodyAccJerk-max()-X tBodyAccJerk-max()-Y tBodyAccJerk-max()-Z tBodyAccJerk-min()-X tBodyAccJerk-min()-Y tBody
AccJerk-min()-Z tBodyAccJerk-sma() tBodyAccJerk-energy()-X tBodyAccJerk-energy()-Y tBodyAccJerk-energy()-Z tBodyAccJerk-iqr()-X
tBodyAccJerk-iqr()-Y tBodyAccJerk-iqr()-Z tBodyAccJerk-entropy()-X tBodyAccJerk-entropy()-Y tBodyAccJerk-entropy()-Z tBodyAccJe
rk-arCoeff()-X,1 tBodyAccJerk-arCoeff()-X,2 tBodyAccJerk-arCoeff()-X,3 tBodyAccJerk-arCoeff()-X,4 tBodyAccJerk-arCoeff()-Y,1 tB
odyAccJerk-arCoeff()-Y,2 tBodyAccJerk-arCoeff()-Y,3 tBodyAccJerk-arCoeff()-Y,4 tBodyAccJerk-arCoeff()-Z,1 tBodyAccJerk-arCoeff
()-Z,2 tBodyAccJerk-arCoeff()-Z,3 tBodyAccJerk-arCoeff()-Z,4 tBodyAccJerk-correlation()-X,Y tBodyAccJerk-correlation()-X,Z tBod
yAccJerk-correlation()-Y,Z tBodyGyro-mean()-X tBodyGyro-mean()-Y tBodyGyro-mean()-Z tBodyGyro-std()-X tBodyGyro-std()-Y tBodyGy
ro-std()-Z tBodyGyro-mad()-X tBodyGyro-mad()-Y tBodyGyro-mad()-Z tBodyGyro-max()-X tBodyGyro-max()-Y tBodyGyro-max()-Z tBodyGyr
o-min()-X tBodyGyro-min()-Y tBodyGyro-min()-Z tBodyGyro-sma() tBodyGyro-energy()-X tBodyGyro-energy()-Y tBodyGyro-energy()-Z tB
odyGyro-iqr()-X tBodyGyro-iqr()-Y tBodyGyro-iqr()-Z tBodyGyro-entropy()-X tBodyGyro-entropy()-Y tBodyGyro-entropy()-Z tBodyGyro
-arCoeff()-X,1 tBodyGyro-arCoeff()-X,2 tBodyGyro-arCoeff()-X,3 tBodyGyro-arCoeff()-X,4 tBodyGyro-arCoeff()-Y,1 tBodyGyro-arCoef
f()-Y,2 tBodyGyro-arCoeff()-Y,3 tBodyGyro-arCoeff()-Y,4 tBodyGyro-arCoeff()-Z,1 tBodyGyro-arCoeff()-Z,2 tBodyGyro-arCoeff()-Z,3
tBodyGyro-arCoeff()-Z,4 tBodyGyro-correlation()-X,Y tBodyGyro-correlation()-X,Z tBodyGyro-correlation()-Y,Z tBodyGyroJerk-mean
()-X tBodyGyroJerk-mean()-Y tBodyGyroJerk-mean()-Z tBodyGyroJerk-std()-X tBodyGyroJerk-std()-Y tBodyGyroJerk-std()-Z tBodyGyroJ
erk-mad()-X tBodyGyroJerk-mad()-Y tBodyGyroJerk-mad()-Z tBodyGyroJerk-max()-X tBodyGyroJerk-max()-Y tBodyGyroJerk-max()-Z tBody
GyroJerk-min()-X tBodyGyroJerk-min()-Y tBodyGyroJerk-min()-Z tBodyGyroJerk-sma() tBodyGyroJerk-energy()-X tBodyGyroJerk-energy
()-Y tBodyGyroJerk-energy()-Z tBodyGyroJerk-iqr()-X tBodyGyroJerk-iqr()-Y tBodyGyroJerk-iqr()-Z tBodyGyroJerk-entropy()-X tBody
GyroJerk-entropy()-Y tBodyGyroJerk-entropy()-Z tBodyGyroJerk-arCoeff()-X,1 tBodyGyroJerk-arCoeff()-X,2 tBodyGyroJerk-arCoeff()-
X,3 tBodyGyroJerk-arCoeff()-X,4 tBodyGyroJerk-arCoeff()-Y,1 tBodyGyroJerk-arCoeff()-Y,2 tBodyGyroJerk-arCoeff()-Y,3 tBodyGyroJe
rk-arCoeff()-Y,4 tBodyGyroJerk-arCoeff()-Z,1 tBodyGyroJerk-arCoeff()-Z,2 tBodyGyroJerk-arCoeff()-Z,3 tBodyGyroJerk-arCoeff()-Z,
4 tBodyGyroJerk-correlation()-X,Y tBodyGyroJerk-correlation()-X,Z tBodyGyroJerk-correlation()-Y,Z tBodyAccMag-mean() tBodyAccMa
g-std() tBodyAccMag-mad() tBodyAccMag-max() tBodyAccMag-min() tBodyAccMag-sma() tBodyAccMag-energy() tBodyAccMag-iqr() tBodyAcc
Mag-entropy() tBodyAccMag-arCoeff()1 tBodyAccMag-arCoeff()2 tBodyAccMag-arCoeff()3 tBodyAccMag-arCoeff()4 tGravityAccMag-mean()
tGravityAccMag-std() tGravityAccMag-mad() tGravityAccMag-max() tGravityAccMag-min() tGravityAccMag-sma() tGravityAccMag-energy
() tGravityAccMag-iqr() tGravityAccMag-entropy() tGravityAccMag-arCoeff()1 tGravityAccMag-arCoeff()2 tGravityAccMag-arCoeff()3
tGravityAccMag-arCoeff()4 tBodyAccJerkMag-mean() tBodyAccJerkMag-std() tBodyAccJerkMag-mad() tBodyAccJerkMag-max() tBodyAccJerk
Mag-min() tBodyAccJerkMag-sma() tBodyAccJerkMag-energy() tBodyAccJerkMag-iqr() tBodyAccJerkMag-entropy() tBodyAccJerkMag-arCoef
f()1 tBodyAccJerkMag-arCoeff()2 tBodyAccJerkMag-arCoeff()3 tBodyAccJerkMag-arCoeff()4 tBodyGyroMag-mean() tBodyGyroMag-std() tB
odyGyroMag-mad() tBodyGyroMag-max() tBodyGyroMag-min() tBodyGyroMag-sma() tBodyGyroMag-energy() tBodyGyroMag-iqr() tBodyGyroMag
-entropy() tBodyGyroMag-arCoeff()1 tBodyGyroMag-arCoeff()2 tBodyGyroMag-arCoeff()3 tBodyGyroMag-arCoeff()4 tBodyGyroJerkMag-mea
n() tBodyGyroJerkMag-std() tBodyGyroJerkMag-mad() tBodyGyroJerkMag-max() tBodyGyroJerkMag-min() tBodyGyroJerkMag-sma() tBodyGyr
oJerkMag-energy() tBodyGyroJerkMag-iqr() tBodyGyroJerkMag-entropy() tBodyGyroJerkMag-arCoeff()1 tBodyGyroJerkMag-arCoeff()2 tBo
dyGyroJerkMag-arCoeff()3 tBodyGyroJerkMag-arCoeff()4 fBodyAcc-mean()-X fBodyAcc-mean()-Y fBodyAcc-mean()-Z fBodyAcc-std()-X fBo
dyAcc-std()-Y fBodyAcc-std()-Z fBodyAcc-mad()-X fBodyAcc-mad()-Y fBodyAcc-mad()-Z fBodyAcc-max()-X fBodyAcc-max()-Y fBodyAcc-ma
x()-Z fBodyAcc-min()-X fBodyAcc-min()-Y fBodyAcc-min()-Z fBodyAcc-sma() fBodyAcc-energy()-X fBodyAcc-energy()-Y fBodyAcc-energy
()-Z fBodyAcc-iqr()-X fBodyAcc-iqr()-Y fBodyAcc-iqr()-Z fBodyAcc-entropy()-X fBodyAcc-entropy()-Y fBodyAcc-entropy()-Z fBodyAcc
-maxInds-X fBodyAcc-maxInds-Y fBodyAcc-maxInds-Z fBodyAcc-meanFreq()-X fBodyAcc-meanFreq()-Y fBodyAcc-meanFreq()-Z fBodyAcc-ske
wness()-X fBodyAcc-kurtosis()-X fBodyAcc-skewness()-Y fBodyAcc-kurtosis()-Y fBodyAcc-skewness()-Z fBodyAcc-kurtosis()-Z fBodyAc
c-bandsEnergy1()-1,8 fBodyAcc-bandsEnergy1()-9,16 fBodyAcc-bandsEnergy1()-17,24 fBodyAcc-bandsEnergy1()-25,32 fBodyAcc-bandsEne
rgy1()-33,40 fBodyAcc-bandsEnergy1()-41,48 fBodyAcc-bandsEnergy1()-49,56 fBodyAcc-bandsEnergy1()-57,64 fBodyAcc-bandsEnergy1()-
1,16 fBodyAcc-bandsEnergy1()-17,32 fBodyAcc-bandsEnergy1()-33,48 fBodyAcc-bandsEnergy1()-49,64 fBodyAcc-bandsEnergy1()-1,24 fBo
dyAcc-bandsEnergy1()-25,48 fBodyAcc-bandsEnergy2()-1,8 fBodyAcc-bandsEnergy2()-9,16 fBodyAcc-bandsEnergy2()-17,24 fBodyAcc-band
sEnergy2()-25,32 fBodyAcc-bandsEnergy2()-33,40 fBodyAcc-bandsEnergy2()-41,48 fBodyAcc-bandsEnergy2()-49,56 fBodyAcc-bandsEnergy
2()-57,64 fBodyAcc-bandsEnergy2()-1,16 fBodyAcc-bandsEnergy2()-17,32 fBodyAcc-bandsEnergy2()-33,48 fBodyAcc-bandsEnergy2()-49,6
4 fBodyAcc-bandsEnergy2()-1,24 fBodyAcc-bandsEnergy2()-25,48 fBodyAcc-bandsEnergy3()-1,8 fBodyAcc-bandsEnergy3()-9,16 fBodyAcc-
bandsEnergy3()-17,24 fBodyAcc-bandsEnergy3()-25,32 fBodyAcc-bandsEnergy3()-33,40 fBodyAcc-bandsEnergy3()-41,48 fBodyAcc-bandsEn
ergy3()-49,56 fBodyAcc-bandsEnergy3()-57,64 fBodyAcc-bandsEnergy3()-1,16 fBodyAcc-bandsEnergy3()-17,32 fBodyAcc-bandsEnergy3()-
33,48 fBodyAcc-bandsEnergy3()-49,64 fBodyAcc-bandsEnergy3()-1,24 fBodyAcc-bandsEnergy3()-25,48 fBodyAccJerk-mean()-X fBodyAccJe
rk-mean()-Y fBodyAccJerk-mean()-Z fBodyAccJerk-std()-X fBodyAccJerk-std()-Y fBodyAccJerk-std()-Z fBodyAccJerk-mad()-X fBodyAccJ
erk-mad()-Y fBodyAccJerk-mad()-Z fBodyAccJerk-max()-X fBodyAccJerk-max()-Y fBodyAccJerk-max()-Z fBodyAccJerk-min()-X fBodyAccJe
rk-min()-Y fBodyAccJerk-min()-Z fBodyAccJerk-sma() fBodyAccJerk-energy()-X fBodyAccJerk-energy()-Y fBodyAccJerk-energy()-Z fBod
yAccJerk-iqr()-X fBodyAccJerk-iqr()-Y fBodyAccJerk-iqr()-Z fBodyAccJerk-entropy()-X fBodyAccJerk-entropy()-Y fBodyAccJerk-entro
py()-Z fBodyAccJerk-maxInds-X fBodyAccJerk-maxInds-Y fBodyAccJerk-maxInds-Z fBodyAccJerk-meanFreq()-X fBodyAccJerk-meanFreq()-Y
fBodyAccJerk-meanFreq()-Z fBodyAccJerk-skewness()-X fBodyAccJerk-kurtosis()-X fBodyAccJerk-skewness()-Y fBodyAccJerk-kurtosis()
-Y fBodyAccJerk-skewness()-Z fBodyAccJerk-kurtosis()-Z fBodyAccJerk-bandsEnergy1()-1,8 fBodyAccJerk-bandsEnergy1()-9,16 fBodyAc
cJerk-bandsEnergy1()-17,24 fBodyAccJerk-bandsEnergy1()-25,32 fBodyAccJerk-bandsEnergy1()-33,40 fBodyAccJerk-bandsEnergy1()-41,4
8 fBodyAccJerk-bandsEnergy1()-49,56 fBodyAccJerk-bandsEnergy1()-57,64 fBodyAccJerk-bandsEnergy1()-1,16 fBodyAccJerk-bandsEnergy
1()-17,32 fBodyAccJerk-bandsEnergy1()-33,48 fBodyAccJerk-bandsEnergy1()-49,64 fBodyAccJerk-bandsEnergy1()-1,24 fBodyAccJerk-ban
dsEnergy1()-25,48 fBodyAccJerk-bandsEnergy2()-1,8 fBodyAccJerk-bandsEnergy2()-9,16 fBodyAccJerk-bandsEnergy2()-17,24 fBodyAccJe
rk-bandsEnergy2()-25,32 fBodyAccJerk-bandsEnergy2()-33,40 fBodyAccJerk-bandsEnergy2()-41,48 fBodyAccJerk-bandsEnergy2()-49,56 f
BodyAccJerk-bandsEnergy2()-57,64 fBodyAccJerk-bandsEnergy2()-1,16 fBodyAccJerk-bandsEnergy2()-17,32 fBodyAccJerk-bandsEnergy2()
-33,48 fBodyAccJerk-bandsEnergy2()-49,64 fBodyAccJerk-bandsEnergy2()-1,24 fBodyAccJerk-bandsEnergy2()-25,48 fBodyAccJerk-bandsE
nergy3()-1,8 fBodyAccJerk-bandsEnergy3()-9,16 fBodyAccJerk-bandsEnergy3()-17,24 fBodyAccJerk-bandsEnergy3()-25,32 fBodyAccJerk-
bandsEnergy3()-33,40 fBodyAccJerk-bandsEnergy3()-41,48 fBodyAccJerk-bandsEnergy3()-49,56 fBodyAccJerk-bandsEnergy3()-57,64 fBod
yAccJerk-bandsEnergy3()-1,16 fBodyAccJerk-bandsEnergy3()-17,32 fBodyAccJerk-bandsEnergy3()-33,48 fBodyAccJerk-bandsEnergy3()-4
9,64 fBodyAccJerk-bandsEnergy3()-1,24 fBodyAccJerk-bandsEnergy3()-25,48 fBodyGyro-mean()-X fBodyGyro-mean()-Y fBodyGyro-mean()-
Z fBodyGyro-std()-X fBodyGyro-std()-Y fBodyGyro-std()-Z fBodyGyro-mad()-X fBodyGyro-mad()-Y fBodyGyro-mad()-Z fBodyGyro-max()-X
fBodyGyro-max()-Y fBodyGyro-max()-Z fBodyGyro-min()-X fBodyGyro-min()-Y fBodyGyro-min()-Z fBodyGyro-sma() fBodyGyro-energy()-X
fBodyGyro-energy()-Y fBodyGyro-energy()-Z fBodyGyro-iqr()-X fBodyGyro-iqr()-Y fBodyGyro-iqr()-Z fBodyGyro-entropy()-X fBodyGyro
-entropy()-Y fBodyGyro-entropy()-Z fBodyGyro-maxInds-X fBodyGyro-maxInds-Y fBodyGyro-maxInds-Z fBodyGyro-meanFreq()-X fBodyGyro
-meanFreq()-Y fBodyGyro-meanFreq()-Z fBodyGyro-skewness()-X fBodyGyro-kurtosis()-X fBodyGyro-skewness()-Y fBodyGyro-kurtosis()-
Y fBodyGyro-skewness()-Z fBodyGyro-kurtosis()-Z fBodyGyro-bandsEnergy1()-1,8 fBodyGyro-bandsEnergy1()-9,16 fBodyGyro-bandsEnerg
y1()-17,24 fBodyGyro-bandsEnergy1()-25,32 fBodyGyro-bandsEnergy1()-33,40 fBodyGyro-bandsEnergy1()-41,48 fBodyGyro-bandsEnergy1
()-49,56 fBodyGyro-bandsEnergy1()-57,64 fBodyGyro-bandsEnergy1()-1,16 fBodyGyro-bandsEnergy1()-17,32 fBodyGyro-bandsEnergy1()-3
3,48 fBodyGyro-bandsEnergy1()-49,64 fBodyGyro-bandsEnergy1()-1,24 fBodyGyro-bandsEnergy1()-25,48 fBodyGyro-bandsEnergy2()-1,8 f
BodyGyro-bandsEnergy2()-9,16 fBodyGyro-bandsEnergy2()-17,24 fBodyGyro-bandsEnergy2()-25,32 fBodyGyro-bandsEnergy2()-33,40 fBody
Gyro-bandsEnergy2()-41,48 fBodyGyro-bandsEnergy2()-49,56 fBodyGyro-bandsEnergy2()-57,64 fBodyGyro-bandsEnergy2()-1,16 fBodyGyro
-bandsEnergy2()-17,32 fBodyGyro-bandsEnergy2()-33,48 fBodyGyro-bandsEnergy2()-49,64 fBodyGyro-bandsEnergy2()-1,24 fBodyGyro-ban
dsEnergy2()-25,48 fBodyGyro-bandsEnergy3()-1,8 fBodyGyro-bandsEnergy3()-9,16 fBodyGyro-bandsEnergy3()-17,24 fBodyGyro-bandsEner
gy3()-25,32 fBodyGyro-bandsEnergy3()-33,40 fBodyGyro-bandsEnergy3()-41,48 fBodyGyro-bandsEnergy3()-49,56 fBodyGyro-bandsEnergy3
()-57,64 fBodyGyro-bandsEnergy3()-1,16 fBodyGyro-bandsEnergy3()-17,32 fBodyGyro-bandsEnergy3()-33,48 fBodyGyro-bandsEnergy3()-4
9,64 fBodyGyro-bandsEnergy3()-1,24 fBodyGyro-bandsEnergy3()-25,48 fBodyAccMag-mean() fBodyAccMag-std() fBodyAccMag-mad() fBodyA
ccMag-max() fBodyAccMag-min() fBodyAccMag-sma() fBodyAccMag-energy() fBodyAccMag-iqr() fBodyAccMag-entropy() fBodyAccMag-maxInd
s fBodyAccMag-meanFreq() fBodyAccMag-skewness() fBodyAccMag-kurtosis() fBodyBodyAccJerkMag-mean() fBodyBodyAccJerkMag-std() fBo
dyBodyAccJerkMag-mad() fBodyBodyAccJerkMag-max() fBodyBodyAccJerkMag-min() fBodyBodyAccJerkMag-sma() fBodyBodyAccJerkMag-energy
() fBodyBodyAccJerkMag-iqr() fBodyBodyAccJerkMag-entropy() fBodyBodyAccJerkMag-maxInds fBodyBodyAccJerkMag-meanFreq() fBodyBody
AccJerkMag-skewness() fBodyBodyAccJerkMag-kurtosis() fBodyBodyGyroMag-mean() fBodyBodyGyroMag-std() fBodyBodyGyroMag-mad() fBod
yBodyGyroMag-max() fBodyBodyGyroMag-min() fBodyBodyGyroMag-sma() fBodyBodyGyroMag-energy() fBodyBodyGyroMag-iqr() fBodyBodyGyro
Mag-entropy() fBodyBodyGyroMag-maxInds fBodyBodyGyroMag-meanFreq() fBodyBodyGyroMag-skewness() fBodyBodyGyroMag-kurtosis() fBod
yBodyGyroJerkMag-mean() fBodyBodyGyroJerkMag-std() fBodyBodyGyroJerkMag-mad() fBodyBodyGyroJerkMag-max() fBodyBodyGyroJerkMag-m
in() fBodyBodyGyroJerkMag-sma() fBodyBodyGyroJerkMag-energy() fBodyBodyGyroJerkMag-iqr() fBodyBodyGyroJerkMag-entropy() fBodyBo
dyGyroJerkMag-maxInds fBodyBodyGyroJerkMag-meanFreq() fBodyBodyGyroJerkMag-skewness() fBodyBodyGyroJerkMag-kurtosis() angle(tBo
dyAccMean,gravity) angle(tBodyAccJerkMean),gravityMean) angle(tBodyGyroMean,gravityMean) angle(tBodyGyroJerkMean,gravityMean) a
ngle(X,gravityMean) angle(Y,gravityMean) angle(Z,gravityMean)
```

In [4]:
```python
# Создание тренировочного и тестового датафреймов
train_dataframe = pd.read_csv(path3, sep=" ")
test_dataframe = pd.read_csv(path6, sep=" ")

# Запись меток из файлов
train_labels = pd.read_csv(final_train_labels, header=None)
test_labels = pd.read_csv(final_test_labels, header=None)
```

In [5]:
```python
# Вспомогательный метод для создания tf.data dataset из датафрейма Pandas
def df_to_dataset(dataframe, labels, shuffle=True, batch_size=32):
    dataframe = dataframe.copy()
    ds = tf.data.Dataset.from_tensor_slices((dict(dataframe), labels))
    if shuffle:
        ds = ds.shuffle(buffer_size=len(dataframe))
    ds = ds.batch(batch_size)
    return ds

# Создание пайплайнов (из датафреймов в датасет)
batch_size = 32 # Размер пакета
train_ds = df_to_dataset(train_dataframe, train_labels, batch_size=batch_size)
test_ds = df_to_dataset(test_dataframe, test_labels, shuffle=False, batch_size=batch_size)
```

In [6]:
```python
# Создание списка признаков, если они заданы в файле датасета
for feature_batch, label_batch in train_ds.take(1):
    features = list(feature_batch.keys())

# Создание словаря признаков, если данные хранятся в отдельном файле
features_dict = {}
with open(path_features_file) as file:
    for line in file:
        key, *value = line.split()
        features_dict[key] = value
```

In [7]:
```python
# Создание слоя признаков
feature_columns = [] # Столбцы признаков
for header in range(0, 561):
    feature_columns.append(feature_column.numeric_column((features[header])))
feature_layer = tf.keras.layers.DenseFeatures(feature_columns) # Слой признаков
```

In [8]:
```python
# Для теста
example_batch = next(iter(train_ds))[0]
# Служебный метод для создания столбца признаков и преобразования пакета данных
def demo(feature_column):
    feature_layer = layers.DenseFeatures(feature_column)
    print(feature_column)
    print(feature_layer(example_batch).numpy())
f_test = feature_column.numeric_column("tBodyAcc-mean()-X")
demo(f_test)
```

```
NumericColumn(key='tBodyAcc-mean()-X', shape=(1,), default_value=None, dtype=tf.float32, normalizer_fn=None)
WARNING:tensorflow:Layer dense_features_1 is casting an input tensor from dtype float64 to the layer's dtype of float32, which
is new behavior in TensorFlow 2.  The layer has dtype float32 because it's dtype defaults to floatx.

If you intended to run this layer in float32, you can safely ignore this warning. If in doubt, this warning is likely only an i
ssue if you are porting a TensorFlow 1.X model to TensorFlow 2.

To change all layers to have dtype float64 by default, call `tf.keras.backend.set_floatx('float64')`. To change just this laye
r, pass dtype='float64' to the layer constructor. If you are the author of this layer, you can disable autocasting by passing a
utocast=False to the base Layer constructor.

[[0.27296337]
 [0.27346438]
 [0.27752978]
 [0.24910697]
 [0.2711866 ]
 [0.14872362]
 [0.28300244]
 [0.28024137]
 [0.2782633 ]
 [0.25836396]
 [0.3308318 ]
 [0.27466685]
 [0.24445261]
 [0.23489885]
 [0.1561557 ]
 [0.2793797 ]
 [0.27451178]
 [0.25200006]
 [0.24723145]
 [0.2783501 ]
 [0.2748853 ]
 [0.27311456]
 [0.27727452]
 [0.27683604]
 [0.28504223]
 [0.1280471 ]
 [0.22559084]
 [0.35935047]
 [0.27190846]
 [0.28265026]
 [0.22506213]
 [0.19526114]]
```

In [9]:
```python
# Создание модели
model = tf.keras.Sequential([
    feature_layer,
    layers.Dense(128, activation=tf.nn.relu),
    layers.Dense(128, activation=tf.nn.relu),
    layers.Dense(6, activation=tf.nn.softmax)
])

# Компиляция модели
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy', metrics=['accuracy'], run_eagerly=True)
```

In [10]:
```python
# Для сохранения весов модели
checkpoint_path = r"./checkpoint/cp.ckpt"
cp_callback = tf.keras.callbacks.ModelCheckpoint(filepath=checkpoint_path, save_weights_only=True, verbose=1)

# Тренировка модели
history = model.fit(train_ds, validation_data=test_ds, epochs=3, callbacks=[cp_callback])
```

```
Epoch 1/3
WARNING:tensorflow:Layer dense_features is casting an input tensor from dtype float64 to the layer's dtype of float32, which is
new behavior in TensorFlow 2.  The layer has dtype float32 because it's dtype defaults to floatx.

If you intended to run this layer in float32, you can safely ignore this warning. If in doubt, this warning is likely only an i
ssue if you are porting a TensorFlow 1.X model to TensorFlow 2.

To change all layers to have dtype float64 by default, call `tf.keras.backend.set_floatx('float64')`. To change just this laye
r, pass dtype='float64' to the layer constructor. If you are the author of this layer, you can disable autocasting by passing a
utocast=False to the base Layer constructor.

230/230 [==============================] - ETA: 0s - loss: 0.4035 - accuracy: 0.8379
Epoch 00001: saving model to ./checkpoint/cp.ckpt
230/230 [==============================] - 115s 498ms/step - loss: 0.4035 - accuracy: 0.8379 - val_loss: 0.2689 - val_accuracy:
0.9016
Epoch 2/3
230/230 [==============================] - ETA: 0s - loss: 0.1357 - accuracy: 0.9470
Epoch 00002: saving model to ./checkpoint/cp.ckpt
230/230 [==============================] - 113s 490ms/step - loss: 0.1357 - accuracy: 0.9470 - val_loss: 0.2529 - val_accuracy:
0.8928
Epoch 3/3
230/230 [==============================] - ETA: 0s - loss: 0.1048 - accuracy: 0.9589
Epoch 00003: saving model to ./checkpoint/cp.ckpt
230/230 [==============================] - 114s 495ms/step - loss: 0.1048 - accuracy: 0.9589 - val_loss: 0.2030 - val_accuracy:
0.9213
```

In [11]:
```python
# Визуализация результатов тренировки
import matplotlib.pyplot as plt
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']
loss = history.history['loss']
val_loss = history.history['val_loss']
epochs_range = range(3)
plt.figure(figsize=(8, 8))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy')
plt.plot(epochs_range, val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')
plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss')
plt.plot(epochs_range, val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()
```

In [ ]: