Project 2. HMM 적용하여 데이터 모델링 해보기 (자유주제)

Hidden Markov Model

- 은닉마코프모델 계산 및 구현
  - https://ratsgo.github.io/machine%20learning/2017/10/14/computeHMMs/ (https://ratsgo.github.io/machine%20learning/2017/10/14/computeHMMs/)
- https://web.stanford.edu/~jurafsky/slp3/A.pdf (https://web.stanford.edu/~jurafsky/slp3/A.pdf)

~~Taxi Service Trajectory (TST)~~

- Taxi Service Trajectory (TST) Prediction Challenge 2015
  - http://www.geolink.pt/ecmlpkdd2015-challenge/index.html (http://www.geolink.pt/ecmlpkdd2015-challenge/index.html)
  - Artificial Neural Networks Applied to Taxi Destination Prediction
    - https://arxiv.org/pdf/1508.00021.pdf (https://arxiv.org/pdf/1508.00021.pdf)

Human Activity Recognition (HAR)

- ~~Smartphone Dataset for Human Activity Recognition (HAR) in Ambient Assisted Living (AAL) Data Set~~
  - Dataset (https://archive.ics.uci.edu/ml/datasets/Smartphone+Dataset+for+Human+Activity+Recognition+%28HAR%29+in+Ambient+Assisted+Living+%28AAL%2
  - Author
    - *Kadian Alicia Davis, Evans Boateng Owusu*
  - Structure
    - Triaxial acceleration from the accelerometer (total acceleration)
      - `final_acc_train.txt` , `final_acc_test.txt`
    - Triaxial Angular velocity from the gyroscope.
      - `final_gyro_train.txt` , `final_gyro_test.txt`
    - A 561-feature vector with time and frequency domain variables
      - `final_X_train.txt` , `final_X_test.txt`
    - The corresponding activity labels
      - `final_y_train.txt` , `final_y_test.txt`
- **Human Activity Recognition Using Smartphones Data Set**
  - Dataset (https://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones)
  - Author
    - *Jorge L. Reyes-Ortiz, Davide Anguita, Alessandro Ghio, Luca Oneto and Xavier Parra*
  - Structure
    - Raw Data
      - `acc_exp#{1~61}_user{1~30}.txt`
      - `gyro_exp#{1~61}_user{1~30}.txt`
      - `labels.txt`

In [1]:

```
from scipy import io
from hmmlearn import hmm

import matplotlib.pyplot as plt
import numpy as np
import time

labels = np.loadtxt('./HAR/RawData/labels.txt', delimiter=' ', dtype=in
t)
N = len(labels)

acc = np.empty((N), dtype=object)
gyro = np.empty((N), dtype=object)

y = np.empty((N),dtype=int)

actions =['WALKING', 'WALKING_UPSTAIRS', 'WALKING_DOWNSTAIRS', 'SITTING
',
          'STANDING', 'LAYING', 'STAND_TO_SIT', 'SIT_TO_STAND', 'SIT_TO_
LIE',
          'LIE_TO_SIT', 'STAND_TO_LIE', 'LIE_TO_STAND']
num_actions = len(actions)
```

In [2]:

```
prev = ''
acc_file = np.empty([])
gyro_file = np.empty([])
for (i, row) in enumerate(labels):
    filename = 'exp{:02d}_user{:02d}.txt'.format(row[0], row[1])

    if prev != filename:
        acc_file = np.loadtxt('./HAR/RawData/acc_' + filename, delimiter=' ')
        gyro_file = np.loadtxt('./HAR/RawData/gyro_' + filename, delimiter=' ')
        prev = filename
        print(filename)

    acc[i] = acc_file[row[3]:row[4]+1, :]
    gyro[i] = gyro_file[row[3]:row[4]+1, :]
    y[i] = row[2]-1
```

```
exp01_user01.txt
exp02_user01.txt
exp03_user02.txt
exp04_user02.txt
exp05_user03.txt
exp06_user03.txt
exp07_user04.txt
exp08_user04.txt
exp09_user05.txt
exp10_user05.txt
exp11_user06.txt
exp12_user06.txt
exp13_user07.txt
exp14_user07.txt
exp15_user08.txt
exp16_user08.txt
exp17_user09.txt
exp18_user09.txt
exp19_user10.txt
exp20_user10.txt
exp21_user10.txt
exp22_user11.txt
exp23_user11.txt
exp24_user12.txt
exp25_user12.txt
exp26_user13.txt
exp27_user13.txt
exp28_user14.txt
exp29_user14.txt
exp30_user15.txt
exp31_user15.txt
exp32_user16.txt
exp33_user16.txt
exp34_user17.txt
exp35_user17.txt
exp36_user18.txt
exp37_user18.txt
exp38_user19.txt
exp39_user19.txt
exp40_user20.txt
exp41_user20.txt
exp42_user21.txt
exp43_user21.txt
exp44_user22.txt
exp45_user22.txt
exp46_user23.txt
exp47_user23.txt
exp48_user24.txt
exp49_user24.txt
exp50_user25.txt
exp51_user25.txt
exp52_user26.txt
exp53_user26.txt
exp54_user27.txt
exp55_user27.txt
exp56_user28.txt
exp57_user28.txt
exp58_user29.txt
exp59_user29.txt
exp60_user30.txt
exp61_user30.txt
```

In [3]:

```
for i in range(N):
    print(acc[i].shape, gyro[i].shape, actions[y[i]])
```

```
(983, 3) (983, 3) STANDING
(160, 3) (160, 3) STAND_TO_SIT
(802, 3) (802, 3) SITTING
(165, 3) (165, 3) SIT_TO_STAND
(1015, 3) (1015, 3) STANDING
(288, 3) (288, 3) STAND_TO_LIE
(876, 3) (876, 3) LAYING
(197, 3) (197, 3) LIE_TO_SIT
(932, 3) (932, 3) SITTING
(192, 3) (192, 3) SIT_TO_LIE
(927, 3) (927, 3) LAYING
(191, 3) (191, 3) LIE_TO_STAND
(583, 3) (583, 3) WALKING
(895, 3) (895, 3) WALKING
(911, 3) (911, 3) WALKING
(965, 3) (965, 3) WALKING
(656, 3) (656, 3) WALKING_DOWNSTAIRS
(631, 3) (631, 3) WALKING_UPSTAIRS
(624, 3) (624, 3) WALKING_DOWNSTAIRS
(666, 3) (666, 3) WALKING_UPSTAIRS
(624, 3) (624, 3) WALKING_DOWNSTAIRS
(673, 3) (673, 3) WALKING_UPSTAIRS
(976, 3) (976, 3) STANDING
(206, 3) (206, 3) STAND_TO_SIT
(789, 3) (789, 3) SITTING
(156, 3) (156, 3) SIT_TO_STAND
(927, 3) (927, 3) STANDING
(268, 3) (268, 3) STAND_TO_LIE
(863, 3) (863, 3) LAYING
(184, 3) (184, 3) LIE_TO_SIT
(833, 3) (833, 3) SITTING
(237, 3) (237, 3) SIT_TO_LIE
(778, 3) (778, 3) LAYING
(242, 3) (242, 3) LIE_TO_STAND
(629, 3) (629, 3) WALKING
(959, 3) (959, 3) WALKING
(937, 3) (937, 3) WALKING
(972, 3) (972, 3) WALKING
(251, 3) (251, 3) WALKING_DOWNSTAIRS
(433, 3) (433, 3) WALKING_DOWNSTAIRS
(656, 3) (656, 3) WALKING_UPSTAIRS
(648, 3) (648, 3) WALKING_DOWNSTAIRS
(679, 3) (679, 3) WALKING_UPSTAIRS
(625, 3) (625, 3) WALKING_DOWNSTAIRS
(701, 3) (701, 3) WALKING_UPSTAIRS
(1101, 3) (1101, 3) STANDING
(157, 3) (157, 3) STAND_TO_SIT
(942, 3) (942, 3) SITTING
(142, 3) (142, 3) SIT_TO_STAND
(1135, 3) (1135, 3) STANDING
(418, 3) (418, 3) STAND_TO_LIE
(817, 3) (817, 3) LAYING
(278, 3) (278, 3) LIE_TO_SIT
(773, 3) (773, 3) SITTING
(225, 3) (225, 3) SIT_TO_LIE
(952, 3) (952, 3) LAYING
(181, 3) (181, 3) LIE_TO_STAND
(1068, 3) (1068, 3) WALKING
(1073, 3) (1073, 3) WALKING
(687, 3) (687, 3) WALKING_DOWNSTAIRS
(676, 3) (676, 3) WALKING_UPSTAIRS
(596, 3) (596, 3) WALKING_DOWNSTAIRS
(639, 3) (639, 3) WALKING_UPSTAIRS
(526, 3) (526, 3) WALKING_DOWNSTAIRS
(612, 3) (612, 3) WALKING_UPSTAIRS
(828, 3) (828, 3) STANDING
(160, 3) (160, 3) STAND_TO_SIT
(798, 3) (798, 3) SITTING
```

In [5]:

```
obs = np.empty((N), dtype=object)
origin_data = np.empty((N, 3), dtype=object)

def get_observation(acc, gyro):
    time = acc.shape[0]
    obs = np.empty((time), dtype=int)
    for t in range(time):
        accNum = gyroNum = 0
        for i in range(3):
            if acc[t, i] > 0:
                accNum += (1 << (2-i))
            if gyro[t, i] > 0:
                gyroNum += (1 << (2-i))
        obs[t] = accNum*8 + gyroNum
    return obs

for i in range(N):
    obs[i] = get_observation(acc[i], gyro[i])
    origin_data[i] = [y[i], obs[i], i]

print(origin_data.shape)
```

```
(1214, 3)
```

train_test_split

- `train_size : test_size = 90 : 10`

In [6]:

```
from sklearn.model_selection import train_test_split

# train, test split
train_data, test_data = train_test_split(origin_data, test_size=0.1)

# sort in charlabel
train_data = train_data[train_data[:, 0].argsort()]
test_data = test_data[test_data[:, 0].argsort()]

print('train:', train_data.shape)
print('test:', test_data.shape)
print(test_data)
```

```
train: (1092, 3)
test: (122, 3)
[[0 array([35, 35, 35, ..., 38, 38, 54]) 506]
 [0
  array([39, 39, 39, 47, 47, 47, 47, 46, 42, 58, 58, 58, 58, 42, 43, 43,
41,
       41, 45, 39, 39, 39, 35, 49, 49, 36, 38, 38, 38, 38, 38, 38, 46, 6
0,
       60, 60, 60, 62, 63, 63, 45, 45, 45, 61, 60, 52, 36, 36, 38, 44, 4
5,
       61, 55, 51, 51, 51, 59, 59, 43, 43, 43, 40, 40, 40, 56, 56, 58, 4
3,
       41, 37, 37, 39, 39, 35, 51, 50, 48, 36, 36, 38, 38, 38, 42, 58, 6
0,
       60, 60, 60, 45, 45, 45, 45, 45, 44, 52, 52, 52, 36, 32, 37, 45, 4
7,
       47, 51, 51, 51, 51, 59, 59, 43, 41, 40, 40, 40, 40, 40, 56, 40, 4
1,
       37, 39, 35, 39, 37, 49, 53, 52, 36, 36, 36, 38, 38, 38, 62, 60, 6
0,
       60, 62, 63, 63, 45, 45, 45, 61, 53, 36, 36, 36, 34, 45, 45, 39, 3
9,
       55, 51, 51, 51, 59, 59, 59, 56, 56, 40, 40, 40, 40, 42, 43, 43, 3
3,
       35, 35, 33, 33, 33, 34, 34, 38, 38, 38, 36, 36, 60, 60, 60, 60, 6
2,
       63, 63, 63, 61, 44, 60, 60, 60, 36, 36, 36, 36, 44, 36, 39, 39, 3
9,
       55, 59, 59, 59, 41, 41, 41, 41, 40, 40, 40, 40, 41, 45, 37, 39, 3
5,
       35, 35, 51, 49, 48, 36, 36, 36, 38, 38, 38, 46, 62, 60, 62, 62, 6
3,
       63, 61, 61, 44, 44, 44, 36, 36, 36, 36, 36, 32, 37, 45, 39, 39, 3
5,
       51, 59, 59, 57, 57, 41, 41, 42, 40, 40, 41, 41, 41, 45, 37, 39, 3
9,
       39, 33, 51, 51, 50, 36, 36, 38, 38, 38, 38, 46, 62, 60, 60, 60, 6
1,
       61, 45, 45, 47, 46, 60, 52, 36, 36, 32, 32, 41, 43, 39, 51, 51, 5
1,
       59, 59, 43, 43, 43, 43, 41, 40, 40, 40, 40, 42, 47, 37, 39, 35, 3
9,
       53, 53, 53, 36, 36, 36, 38, 38, 38, 38, 46, 60, 60, 60, 62, 62, 6
3,
       63, 45, 60, 60, 60, 36, 36, 36, 48, 33, 37, 39, 35, 51, 51, 51, 3
5,
       43, 43, 43, 43, 42, 40, 40, 40, 40, 40, 43, 37, 33, 35, 35, 37, 3
5,
       35, 32, 32, 36, 38, 38, 38, 38, 38, 62, 60, 60, 62, 62, 62, 60, 4
5,
       45, 62, 60, 36, 36, 36, 32, 38, 37, 39, 39, 39, 39, 51, 51, 59, 5
9,
       57, 57, 57, 57, 40, 40, 40, 41, 43, 43, 45, 45, 35, 35, 33, 33, 4
9,
       48, 32, 36, 36, 38, 38, 38, 38, 60, 60, 60, 62, 63, 63, 63, 61, 4
4,
       44, 62, 60, 52, 52, 36, 32, 32, 41, 47, 39, 35, 51, 51, 51, 59, 5
9,
       59, 41, 41, 40, 40, 40, 40, 40, 40, 43, 43, 33, 35, 35, 35, 33, 3
5,
       51, 32, 36, 38, 38, 38, 38, 38, 62, 60, 60, 62, 62, 59, 59, 47, 4
5,
       45, 44, 44, 60, 52, 52, 36, 36, 36, 45, 45, 47, 39, 35, 51, 51, 5
9,
       43, 41, 41, 40, 40, 42, 42, 42, 42, 43, 59, 41, 45, 37, 39, 35, 3
3,
       33, 49, 37, 44, 44, 46, 38, 46, 44, 60, 60, 60, 60, 63, 47, 47, 4
7,
```

In [19]:

```
idx = np.zeros(num_actions + 1, dtype=int)
for i in range(train_data.shape[0]):
    idx[train_data[i][0]+1] = i+1;
```

Human Activity Recognition HMM

HMM Learn

- `hmmlearn` Tutorial
  - https://hmmlearn.readthedocs.io/en/latest/tutorial.html (https://hmmlearn.readthedocs.io/en/latest/tutorial.html)
- `MultinomialHMM` API Reference
  - https://hmmlearn.readthedocs.io/en/latest/api.html#multinomialhmm (https://hmmlearn.readthedocs.io/en/latest/api.html#multinomialhmm)

In [15]:

```
models = np.empty((num_actions), dtype=object)
for i in range(num_actions):
    models[i] = hmm.MultinomialHMM(n_components=64, verbose=True, n_iter
=50)

# multinomial HMM learn
for i in range(num_actions):
    start_time = time.time()
    trainRange = range(idx[i], idx[i+1])
    print('Training', actions[i], 'model w.', idx[i+1]-idx[i], 'examples
', end=' ')

    trainX = np.concatenate([train_data[j][1].reshape(-1, 1) for j in tr
ainRange])
    lengths = [len(train_data[j][1]) for j in trainRange]

    models[i].fit(trainX, lengths)
    print("(elapsed time: {}s).".format(time.time() - start_time))
```

```
Training WALKING model w. 117 examples

         1     -468275.8570             +nan
         2     -356618.6474     +111657.2097
         3     -356393.5111        +225.1362
         4     -356114.9704        +278.5407
         5     -355733.7572        +381.2132
         6     -355157.0342        +576.7230
         7     -354183.0895        +973.9447
         8     -352298.2906       +1884.7989
         9     -347882.4742       +4415.8163
        10     -334774.8760      +13107.5982
        11     -303365.4396      +31409.4364
        12     -271874.5450      +31490.8946
        13     -253797.8182      +18076.7268
        14     -243548.6999      +10249.1183
        15     -236709.3818       +6839.3181
        16     -231679.0704       +5030.3113
        17     -227499.9567       +4179.1137
        18     -223615.8306       +3884.1261
        19     -220030.2184       +3585.6123
        20     -216835.4128       +3194.8055
        21     -214075.9766       +2759.4362
        22     -211813.1879       +2262.7887
        23     -209922.2578       +1890.9301
        24     -208516.8506       +1405.4073
        25     -207579.3689        +937.4816
        26     -206878.1365        +701.2324
        27     -206325.4150        +552.7215
        28     -205888.6151        +436.7998
        29     -205532.6374        +355.9777
        30     -205228.7691        +303.8683
        31     -204960.5947        +268.1744
        32     -204719.8595        +240.7352
        33     -204500.6784        +219.1811
        34     -204298.6735        +202.0049
        35     -204109.9261        +188.7475
        36     -203934.5573        +175.3687
        37     -203772.0468        +162.5105
        38     -203621.9867        +150.0601
        39     -203483.9739        +138.0129
        40     -203356.7152        +127.2587
        41     -203238.1994        +118.5158
        42     -203127.9204        +110.2791
        43     -203025.7660        +102.1543
        44     -202930.7619         +95.0041
        45     -202842.2439         +88.5180
        46     -202760.3519         +81.8920
        47     -202685.1653         +75.1866
        48     -202615.0598         +70.1055
        49     -202546.3511         +68.7087
        50     -202473.7661         +72.5850

(elapsed time: 3876.995486974716s).
Training WALKING_UPSTAIRS model w. 168 examples
```

```
         1     -451213.5401             +nan
         2     -304708.7952     +146504.7450
         3     -304154.6733        +554.1219
         4     -303307.0317        +847.6417
         5     -301695.3602       +1611.6714
         6     -297668.9000       +4026.4602
         7     -283977.0666      +13691.8334
         8     -250568.5421      +33408.5245
         9     -216493.5183      +34075.0239
        10     -193990.1802      +22503.3381
        11     -181948.4424      +12041.7378
        12     -175541.4599       +6406.9824
        13     -170392.8688       +5148.5911
        14     -166282.4128       +4110.4560
        15     -162821.2438       +3461.1690
        16     -159898.8352       +2922.4086
        17     -157461.9260       +2436.9092
        18     -155198.5911       +2263.3349
        19     -152330.4127       +2868.1784
        20     -149921.0525       +2409.3602
        21     -148695.1846       +1225.8679
        22     -147738.9869        +956.1977
        23     -146845.8690        +893.1179
        24     -146249.5486        +596.3204
        25     -145840.4321        +409.1164
        26     -145495.2987        +345.1334
        27     -145071.0519        +424.2468
        28     -144067.9639       +1003.0880
        29     -143121.9511        +946.0129
        30     -142668.1288        +453.8222
        31     -142427.2805        +240.8483
        32     -142268.0800        +159.2005
        33     -142100.2849        +167.7951
        34     -141901.4896        +198.7953
        35     -141742.0119        +159.4778
        36     -141637.2099        +104.8020
        37     -141539.9979         +97.2119
        38     -141455.5542         +84.4438
        39     -141371.3842         +84.1699
        40     -141277.8434         +93.5408
        41     -141187.3308         +90.5127
        42     -141108.8681         +78.4626
        43     -141035.7832         +73.0849
        44     -140966.3517         +69.4316
        45     -140899.4616         +66.8900
        46     -140833.1862         +66.2754
        47     -140765.9233         +67.2629
        48     -140700.3744         +65.5489
        49     -140639.5660         +60.8084
        50     -140582.8274         +56.7387

(elapsed time: 3704.829852581024s).
Training WALKING_DOWNSTAIRS model w. 170 examples
```

```
         1     -412533.5098             +nan
         2     -317224.3599      +95309.1499
         3     -317027.3553        +197.0046
         4     -316779.8301        +247.5253
         5     -316431.8925        +347.9376
         6     -315881.5241        +550.3684
         7     -314880.2219       +1001.3022
         8     -312696.5557       +2183.6662
         9     -307235.9445       +5460.6111
        10     -296543.7446      +10692.1999
        11     -280994.7260      +15549.0186
        12     -262401.9787      +18592.7473
        13     -244727.4347      +17674.5440
        14     -229240.0740      +15487.3607
        15     -218827.1701      +10412.9039
        16     -211056.7455       +7770.4246
        17     -204715.5295       +6341.2160
        18     -200534.6121       +4180.9174
        19     -197649.0961       +2885.5160
        20     -195413.6817       +2235.4145
        21     -193533.3292       +1880.3524
        22     -192097.5184       +1435.8108
        23     -191083.6867       +1013.8317
        24     -190355.8870        +727.7997
        25     -189816.8171        +539.0699
        26     -189340.3178        +476.4993
        27     -188919.1868        +421.1310
        28     -188594.7884        +324.3984
        29     -188336.8372        +257.9512
        30     -188121.3736        +215.4636
        31     -187931.8861        +189.4875
        32     -187760.7927        +171.0934
        33     -187609.0457        +151.7470
        34     -187478.5785        +130.4672
        35     -187361.7524        +116.8261
        36     -187252.0566        +109.6958
        37     -187148.6555        +103.4011
        38     -187047.0105        +101.6450
        39     -186942.4501        +104.5604
        40     -186834.5135        +107.9366
        41     -186727.0732        +107.4403
        42     -186623.6228        +103.4504
        43     -186525.2124         +98.4103
        44     -186432.5084         +92.7041
        45     -186345.0563         +87.4521
        46     -186265.6132         +79.4430
        47     -186196.9449         +68.6683
        48     -186137.5506         +59.3943
        49     -186081.2175         +56.3331
        50     -186022.6938         +58.5236

(elapsed time: 3408.616759777069s).
Training SITTING model w. 102 examples
```

```
         1     -449945.6213             +nan
         2     -315077.0717     +134868.5495
         3     -314651.5004        +425.5713
         4     -313968.6345        +682.8659
         5     -312463.3215       +1505.3129
         6     -307229.0806       +5234.2409
         7     -285783.4396      +21445.6410
         8     -246895.4115      +38888.0281
         9     -214516.8010      +32378.6105
        10     -192649.9914      +21866.8096
        11     -182416.0652      +10233.9261
        12     -176492.4519       +5923.6134
        13     -171143.0090       +5349.4429
        14     -167129.4205       +4013.5885
        15     -164794.4864       +2334.9341
        16     -163598.9500       +1195.5363
        17     -162778.6855        +820.2646
        18     -161957.8032        +820.8823
        19     -161146.8436        +810.9596
        20     -160310.3278        +836.5158
        21     -159667.8022        +642.5256
        22     -159306.7139        +361.0883
        23     -159008.5368        +298.1771
        24     -158722.8630        +285.6738
        25     -158472.7233        +250.1397
        26     -158253.9093        +218.8140
        27     -158073.2202        +180.6891
        28     -157882.9896        +190.2306
        29     -157660.3837        +222.6059
        30     -157483.3169        +177.0668
        31     -157352.7593        +130.5576
        32     -157254.0824         +98.6769
        33     -157182.3233         +71.7591
        34     -157125.6697         +56.6535
        35     -157076.4536         +49.2161
        36     -157032.8838         +43.5699
        37     -156992.6883         +40.1955
        38     -156954.1984         +38.4899
        39     -156915.9951         +38.2033
        40     -156878.0229         +37.9722
        41     -156842.0373         +35.9856
        42     -156806.8524         +35.1849
        43     -156774.5127         +32.3396
        44     -156746.1470         +28.3658
        45     -156719.2761         +26.8709
        46     -156692.7307         +26.5454
        47     -156666.5698         +26.1609
        48     -156641.0777         +25.4921
        49     -156616.1703         +24.9074
        50     -156591.7675         +24.4028

(elapsed time: 3681.860481262207s).
Training STANDING model w. 102 examples
```

```
         1     -485774.9903             +nan
         2     -350501.4511     +135273.5392
         3     -350099.6853        +401.7658
         4     -349521.5934        +578.0920
         5     -348504.9218       +1016.6715
         6     -346223.7060       +2281.2158
         7     -339347.4719       +6876.2341
         8     -316448.4426      +22899.0293
         9     -275548.8345      +40899.6081
        10     -243214.7801      +32334.0545
        11     -220019.2029      +23195.5772
        12     -199838.5703      +20180.6325
        13     -185992.8120      +13845.7583
        14     -178459.2054       +7533.6066
        15     -173626.2538       +4832.9516
        16     -170645.0067       +2981.2471
        17     -168861.4698       +1783.5369
        18     -167616.1455       +1245.3243
        19     -166437.5454       +1178.6001
        20     -165633.5968        +803.9486
        21     -165102.4233        +531.1735
        22     -164703.3186        +399.1047
        23     -164394.0325        +309.2860
        24     -164154.0497        +239.9828
        25     -163951.5199        +202.5298
        26     -163765.6912        +185.8287
        27     -163596.4694        +169.2218
        28     -163441.0617        +155.4076
        29     -163298.4263        +142.6354
        30     -163170.1174        +128.3090
        31     -163055.8820        +114.2354
        32     -162953.0111        +102.8709
        33     -162860.2730         +92.7381
        34     -162778.1502         +82.1228
        35     -162703.9508         +74.1994
        36     -162633.8176         +70.1332
        37     -162564.8469         +68.9707
        38     -162495.1349         +69.7121
        39     -162423.2851         +71.8498
        40     -162346.9291         +76.3560
        41     -162261.2437         +85.6854
        42     -162173.8238         +87.4199
        43     -162093.2947         +80.5291
        44     -162016.0218         +77.2729
        45     -161938.7914         +77.2304
        46     -161863.2584         +75.5330
        47     -161793.6333         +69.6252
        48     -161731.6732         +61.9601
        49     -161673.6191         +58.0541
        50     -161618.9795         +54.6396

(elapsed time: 4011.799980163574s).
Training LAYING model w. 110 examples
```

```
         1     -518544.1476             +nan
         2     -376933.1275     +141611.0201
         3     -376615.8264        +317.3011
         4     -376176.2343        +439.5921
         5     -375460.4417        +715.7926
         6     -374075.8624       +1384.5793
         7     -370833.8714       +3241.9910
         8     -361912.3047       +8921.5667
         9     -339164.0395      +22748.2652
        10     -299433.1371      +39730.9024
        11     -258039.3638      +41393.7732
        12     -231744.6931      +26294.6707
        13     -217237.3893      +14507.3038
        14     -208855.9217       +8381.4675
        15     -204299.4533       +4556.4685
        16     -200290.6315       +4008.8218
        17     -197713.1562       +2577.4753
        18     -196442.4667       +1270.6895
        19     -195428.2029       +1014.2638
        20     -194605.7843        +822.4186
        21     -193995.9765        +609.8078
        22     -193538.0850        +457.8915
        23     -193132.9096        +405.1754
        24     -192687.7190        +445.1906
        25     -192185.8428        +501.8762
        26     -191715.4481        +470.3947
        27     -191216.9238        +498.5242
        28     -190797.6877        +419.2362
        29     -190483.7954        +313.8922
        30     -190231.9318        +251.8636
        31     -190019.0158        +212.9160
        32     -189832.9199        +186.0959
        33     -189666.4263        +166.4936
        34     -189515.7970        +150.6292
        35     -189377.2658        +138.5312
        36     -189249.1251        +128.1407
        37     -189135.0431        +114.0820
        38     -189031.9708        +103.0723
        39     -188938.4791         +93.4917
        40     -188853.9584         +84.5207
        41     -188776.9219         +77.0366
        42     -188706.0169         +70.9050
        43     -188639.9388         +66.0781
        44     -188577.5512         +62.3876
        45     -188517.9842         +59.5671
        46     -188460.5308         +57.4533
        47     -188404.3471         +56.1837
        48     -188349.2986         +55.0485
        49     -188295.8601         +53.4385
        50     -188243.5156         +52.3445

(elapsed time: 3956.0335268974304s).
Training STAND_TO_SIT model w. 55 examples
```

```
         1      -39465.7728             +nan
         2      -29942.6439       +9523.1289
         3      -29886.8054         +55.8385
         4      -29799.5586         +87.2468
         5      -29630.2865        +169.2721
         6      -29212.6816        +417.6050
         7      -27925.1995       +1287.4820
         8      -24342.9175       +3582.2821
         9      -19796.0847       +4546.8328
        10      -16515.6886       +3280.3961
        11      -14185.7344       +2329.9542
        12      -12348.4189       +1837.3156
        13      -11320.6732       +1027.7457
        14      -10530.6040        +790.0691
        15      -10047.8434        +482.7606
        16       -9733.8506        +313.9928
        17       -9581.0143        +152.8363
        18       -9505.9618         +75.0525
        19       -9457.0010         +48.9608
        20       -9417.5690         +39.4319
        21       -9380.1338         +37.4352
        22       -9338.5706         +41.5632
        23       -9294.8700         +43.7005
        24       -9254.5408         +40.3293
        25       -9210.8830         +43.6578
        26       -9172.1774         +38.7056
        27       -9142.9412         +29.2362
        28       -9121.1576         +21.7836
        29       -9102.6408         +18.5169
        30       -9088.0888         +14.5520
        31       -9077.5359         +10.5528
        32       -9068.6965          +8.8395
        33       -9058.8552          +9.8413
        34       -9049.0349          +9.8203
        35       -9042.0396          +6.9953
        36       -9036.1133          +5.9264
        37       -9030.6802          +5.4331
        38       -9025.5508          +5.1293
        39       -9020.5697          +4.9811
        40       -9016.1264          +4.4433
        41       -9012.3212          +3.8052
        42       -9009.0433          +3.2778
        43       -9005.8242          +3.2192
        44       -9002.3894          +3.4348
        45       -8998.4457          +3.9437
        46       -8994.1323          +4.3134
        47       -8990.2021          +3.9302
        48       -8986.7453          +3.4568
        49       -8983.6630          +3.0823
        50       -8980.8293          +2.8338
Fitting a model with 8127 free scalar parameters with only 7481 data poin
ts will result in a degenerate solution.

(elapsed time: 305.054158449173s).
Training SIT_TO_STAND model w. 57 examples
```

```
         1      -31213.1828             +nan
         2      -22000.9081       +9212.2747
         3      -21939.1571         +61.7510
         4      -21829.2595        +109.8977
         5      -21575.4263        +253.8331
         6      -20846.2977        +729.1287
         7      -18780.8196       +2065.4781
         8      -15647.4648       +3133.3548
         9      -12225.2629       +3422.2019
        10       -9624.3064       +2600.9565
        11       -8261.3345       +1362.9719
        12       -7741.1998        +520.1347
        13       -7503.4534        +237.7464
        14       -7294.4824        +208.9710
        15       -7176.6557        +117.8268
        16       -7085.7830         +90.8727
        17       -7011.4139         +74.3691
        18       -6952.8314         +58.5825
        19       -6894.4495         +58.3819
        20       -6844.4184         +50.0311
        21       -6781.8858         +62.5326
        22       -6729.3748         +52.5109
        23       -6709.9931         +19.3817
        24       -6693.5137         +16.4794
        25       -6674.2564         +19.2573
        26       -6659.5213         +14.7351
        27       -6646.8818         +12.6395
        28       -6634.5475         +12.3343
        29       -6623.9365         +10.6109
        30       -6617.0305          +6.9061
        31       -6610.7875          +6.2430
        32       -6604.8651          +5.9224
        33       -6599.3098          +5.5553
        34       -6594.1720          +5.1377
        35       -6589.4568          +4.7152
        36       -6585.1910          +4.2658
        37       -6581.3558          +3.8352
        38       -6577.8827          +3.4731
        39       -6574.7189          +3.1637
        40       -6571.7862          +2.9327
        41       -6568.9223          +2.8639
        42       -6565.9541          +2.9682
        43       -6563.0251          +2.9290
        44       -6560.3740          +2.6511
        45       -6557.6966          +2.6775
        46       -6554.7599          +2.9367
        47       -6552.1550          +2.6049
        48       -6550.0896          +2.0654
        49       -6548.1704          +1.9192
        50       -6546.1665          +2.0039

(elapsed time: 221.56788611412048s).
Training SIT_TO_LIE model w. 53 examples
```

```
         1      -46038.5699             +nan
         2      -33285.4589      +12753.1109
         3      -33218.6513         +66.8077
         4      -33112.0199        +106.6314
         5      -32903.1253        +208.8946
         6      -32401.4680        +501.6573
         7      -30901.4415       +1500.0265
         8      -27362.0917       +3539.3498
         9      -23019.0333       +4343.0583
        10      -19191.1123       +3827.9210
        11      -16995.7933       +2195.3190
        12      -15491.6024       +1504.1908
        13      -14211.3647       +1280.2377
        14      -13389.9430        +821.4217
        15      -12979.5910        +410.3519
        16      -12637.3835        +342.2075
        17      -12411.8618        +225.5217
        18      -12318.1286         +93.7333
        19      -12263.2394         +54.8892
        20      -12217.8264         +45.4129
        21      -12176.8557         +40.9708
        22      -12138.7419         +38.1137
        23      -12101.2052         +37.5367
        24      -12066.8128         +34.3924
        25      -12038.7902         +28.0226
        26      -12012.9194         +25.8707
        27      -11987.3791         +25.5403
        28      -11962.5153         +24.8638
        29      -11934.1168         +28.3985
        30      -11880.0765         +54.0403
        31      -11823.7552         +56.3213
        32      -11781.4239         +42.3313
        33      -11756.7415         +24.6824
        34      -11716.4211         +40.3203
        35      -11684.0885         +32.3327
        36      -11662.8073         +21.2811
        37      -11639.6331         +23.1742
        38      -11622.8255         +16.8076
        39      -11601.2863         +21.5392
        40      -11586.8453         +14.4410
        41      -11566.4308         +20.4145
        42      -11540.8055         +25.6253
        43      -11527.7690         +13.0364
        44      -11512.0052         +15.7638
        45      -11499.1317         +12.8735
        46      -11486.8023         +12.3294
        47      -11476.6084         +10.1939
        48      -11466.4139         +10.1945
        49      -11458.2433          +8.1707
        50      -11448.7931          +9.4501

(elapsed time: 370.2914688587189s).
Training LIE_TO_SIT model w. 53 examples
```

```
         1      -41455.9138             +nan
         2      -28714.7987      +12741.1152
         3      -28613.3866        +101.4121
         4      -28423.5517        +189.8349
         5      -27964.0691        +459.4826
         6      -26655.4091       +1308.6600
         7      -23429.5847       +3225.8244
         8      -20092.0729       +3337.5118
         9      -17406.6621       +2685.4108
        10      -14926.5629       +2480.0992
        11      -13186.8424       +1739.7205
        12      -12152.4585       +1034.3839
        13      -11520.0867        +632.3718
        14      -11086.2237        +433.8630
        15      -10855.0087        +231.2151
        16      -10685.4617        +169.5470
        17      -10493.2284        +192.2333
        18      -10192.0464        +301.1820
        19      -10033.7674        +158.2790
        20       -9945.2099         +88.5575
        21       -9889.0960         +56.1139
        22       -9837.0644         +52.0316
        23       -9772.9016         +64.1628
        24       -9702.9713         +69.9303
        25       -9593.0928        +109.8785
        26       -9489.7764        +103.3164
        27       -9448.9890         +40.7874
        28       -9416.3554         +32.6336
        29       -9385.2938         +31.0615
        30       -9348.3723         +36.9215
        31       -9319.5704         +28.8019
        32       -9299.4263         +20.1442
        33       -9285.4420         +13.9843
        34       -9272.1076         +13.3344
        35       -9259.0291         +13.0786
        36       -9246.2718         +12.7573
        37       -9228.6050         +17.6668
        38       -9209.7629         +18.8422
        39       -9196.4226         +13.3403
        40       -9184.2347         +12.1879
        41       -9175.3479          +8.8868
        42       -9165.6389          +9.7089
        43       -9158.0100          +7.6290
        44       -9149.6271          +8.3829
        45       -9139.8707          +9.7564
        46       -9116.6760         +23.1947
        47       -9100.0389         +16.6371
        48       -9093.5745          +6.4644
        49       -9079.9471         +13.6274
        50       -9051.1393         +28.8078

(elapsed time: 331.27107095718384s).
Training STAND_TO_LIE model w. 53 examples
```

```
         1     -53783.7322             +nan
         2     -40672.6734      +13111.0588
         3     -40586.4435         +86.2299
         4     -40447.0328        +139.4107
         5     -40167.6545        +279.3783
         6     -39475.9072        +691.7473
         7     -37405.1025       +2070.8047
         8     -33157.5655       +4247.5370
         9     -28400.3678       +4757.1977
        10     -23754.8415       +4645.5262
        11     -20959.4765       +2795.3651
        12     -19160.4877       +1798.9888
        13     -17578.5910       +1581.8967
        14     -16679.8699        +898.7210
        15     -16377.9040        +301.9659
        16     -16136.7718        +241.1322
        17     -15948.0082        +188.7636
        18     -15728.0453        +219.9629
        19     -15472.7298        +255.3155
        20     -15301.9401        +170.7897
        21     -15177.7980        +124.1421
        22     -15061.5836        +116.2144
        23     -14937.8373        +123.7463
        24     -14820.7751        +117.0622
        25     -14750.2566         +70.5185
        26     -14712.3361         +37.9204
        27     -14685.8764         +26.4597
        28     -14660.0742         +25.8022
        29     -14632.2477         +27.8265
        30     -14604.2652         +27.9826
        31     -14558.8294         +45.4357
        32     -14463.6664         +95.1630
        33     -14352.9935        +110.6729
        34     -14311.7124         +41.2811
        35     -14291.8804         +19.8320
        36     -14272.4509         +19.4295
        37     -14257.1393         +15.3116
        38     -14246.7396         +10.3997
        39     -14238.0093          +8.7303
        40     -14226.8201         +11.1891
        41     -14213.3027         +13.5175
        42     -14182.3237         +30.9790
        43     -14138.9225         +43.4012
        44     -14116.5401         +22.3824
        45     -14101.8976         +14.6425
        46     -14083.4590         +18.4386
        47     -14058.1015         +25.3575
        48     -14051.2071          +6.8944
        49     -14046.6539          +4.5533
        50     -14043.5583          +3.0956

(elapsed time: 428.47260451316833s).
Training LIE_TO_STAND model w. 52 examples
```

```
         1      -40293.4739             +nan
         2      -30245.6996      +10047.7743
         3      -30173.8722         +71.8274
         4      -30055.9238        +117.9484
         5      -29817.9818        +237.9419
         6      -29230.8076        +587.1742
         7      -27471.7321       +1759.0756
         8      -24087.3634       +3384.3687
         9      -20831.5653       +3255.7981
        10      -17512.1184       +3319.4468
        11      -14993.8745       +2518.2439
        12      -13434.7846       +1559.0898
        13      -12528.9874        +905.7973
        14      -11854.7071        +674.2803
        15      -11486.6336        +368.0734
        16      -11285.6799        +200.9537
        17      -11143.5182        +142.1617
        18      -10980.7300        +162.7882
        19      -10845.3609        +135.3691
        20      -10767.6346         +77.7263
        21      -10698.5536         +69.0810
        22      -10634.2357         +64.3178
        23      -10582.9439         +51.2918
        24      -10465.3652        +117.5787
        25      -10407.8662         +57.4990
        26      -10357.7230         +50.1432
        27      -10331.0574         +26.6656
        28      -10292.9230         +38.1344
        29      -10249.2177         +43.7052
        30      -10221.6223         +27.5955
        31      -10196.4785         +25.1438
        32      -10179.7664         +16.7121
        33      -10171.0531          +8.7133
        34      -10163.4593          +7.5939
        35      -10155.7546          +7.7047
        36      -10148.2758          +7.4787
        37      -10137.4227         +10.8531
        38      -10126.0187         +11.4040
        39      -10107.4278         +18.5909
        40      -10093.2673         +14.1604
        41      -10086.6562          +6.6111
        42      -10082.1822          +4.4740
        43      -10078.6717          +3.5105
        44      -10075.6830          +2.9887
        45      -10072.5752          +3.1078
        46      -10068.0222          +4.5530
        47      -10061.1133          +6.9089
        48      -10053.9566          +7.1567
        49      -10048.0924          +5.8642

(elapsed time: 326.9073598384857s).

        50      -10045.4167          +2.6757
```

In [37]:

```
def test_hmm_models(hmm_models):
    testSize = test_data.shape[0]
    wrongCases = 0

    print('=================== Wrong Cases ===================')
    for i in range(testSize):
        testX = np.concatenate([test_data[i][1].reshape(-1, 1)])

        maxScore = hmm_models[0].score(testX)
        maxAction = 0

        for action in range(num_actions):
            score = hmm_models[action].score(testX)
            if maxScore < score:
                maxScore = score
                maxAction = action

        if not test_data[i][0] == maxAction:
            print(actions[test_data[i][0]], actions[maxAction], maxScor
e, test_data[i][0] == maxAction)
            wrongCases += 1

            expect = origin_data[test_data[i][2]][1]
            result = origin_data[train_data[idx[maxAction]][2]][1]

            ax1 = plt.subplot('211')
            ax1.set_ylabel('state')
            ax1.set_title('{} vs. {}'.format(actions[test_data[i][0]], a
ctions[maxAction]))
            plt.plot(expect)

            ax2 = plt.subplot('212', sharex=ax1, sharey=ax1)
            ax2.set_ylabel('state')
            ax2.set_xlabel('time')
            plt.plot(result)

            plt.show()

    print('(Wrong, Total)', (wrongCases, testSize))
    print('Accuracy: {}%'.format((1 - wrongCases/testSize) * 100))
```

In [17]:

```
import pickle
import os
def save_hmm_models(models, name):
    if os.path.exists(name):
        os.system("rm -rf " + name)
    os.mkdir(name)
    for i in range(num_actions):
        filename = './{}/{}_{:02d}_{}.pkl'.format(name, name, i, actions
[i])
        with open(filename, 'wb') as file: pickle.dump(models[i], file)
        print('Save', filename)
```

In [53]:

```
save_hmm_models(models, 'acc8vel8iter50')
```

```
Save ./acc8vel8iter50/acc8vel8iter50_00_WALKING.pkl
Save ./acc8vel8iter50/acc8vel8iter50_01_WALKING_UPSTAIRS.pkl
Save ./acc8vel8iter50/acc8vel8iter50_02_WALKING_DOWNSTAIRS.pkl
Save ./acc8vel8iter50/acc8vel8iter50_03_SITTING.pkl
Save ./acc8vel8iter50/acc8vel8iter50_04_STANDING.pkl
Save ./acc8vel8iter50/acc8vel8iter50_05_LAYING.pkl
Save ./acc8vel8iter50/acc8vel8iter50_06_STAND_TO_SIT.pkl
Save ./acc8vel8iter50/acc8vel8iter50_07_SIT_TO_STAND.pkl
Save ./acc8vel8iter50/acc8vel8iter50_08_SIT_TO_LIE.pkl
Save ./acc8vel8iter50/acc8vel8iter50_09_LIE_TO_SIT.pkl
Save ./acc8vel8iter50/acc8vel8iter50_10_STAND_TO_LIE.pkl
Save ./acc8vel8iter50/acc8vel8iter50_11_LIE_TO_STAND.pkl
```

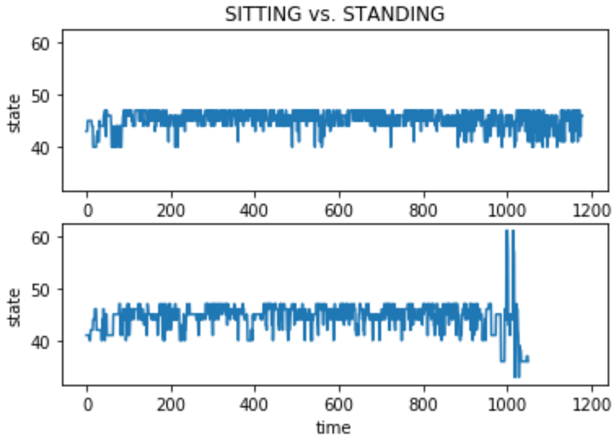
In [9]:

```
import pickle
def load_hmm_models(name):
    hmm_models = []
    for i in range(num_actions):
        filename = './{}/{}_{:02d}_{}.pkl'.format(name, name, i, actions
[i])
        with open(filename, 'rb') as file: hmm_models.append(pickle.load
(file))
        print('Open', filename)
    return hmm_models
```

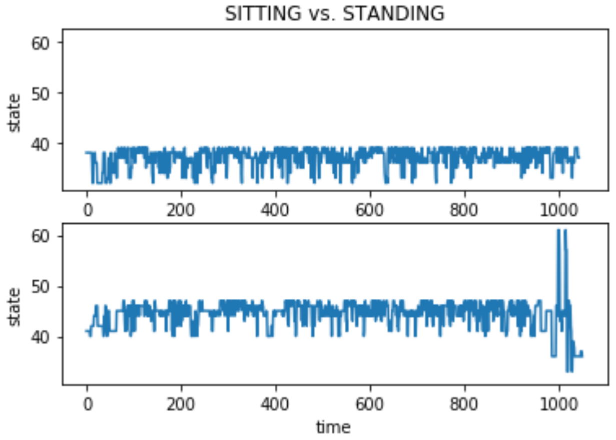
In [38]:

```
new_models = load_hmm_models('acc8vel8iter50')
# new_models = load_hmm_models('jerk8acc8iter1')
test_hmm_models(new_models)
```

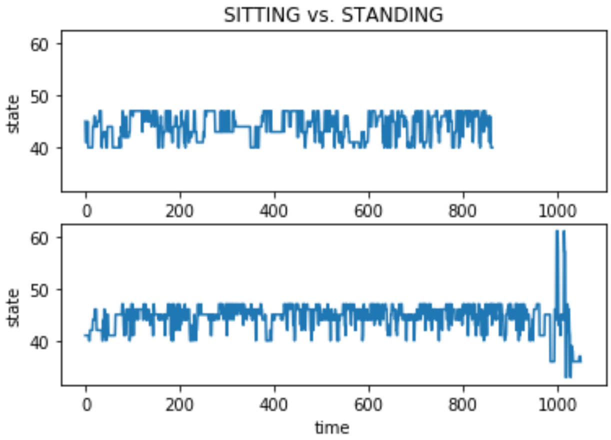
```
Open ./acc8vel8iter50/acc8vel8iter50_00_WALKING.pkl
Open ./acc8vel8iter50/acc8vel8iter50_01_WALKING_UPSTAIRS.pkl
Open ./acc8vel8iter50/acc8vel8iter50_02_WALKING_DOWNSTAIRS.pkl
Open ./acc8vel8iter50/acc8vel8iter50_03_SITTING.pkl
Open ./acc8vel8iter50/acc8vel8iter50_04_STANDING.pkl
Open ./acc8vel8iter50/acc8vel8iter50_05_LAYING.pkl
Open ./acc8vel8iter50/acc8vel8iter50_06_STAND_TO_SIT.pkl
Open ./acc8vel8iter50/acc8vel8iter50_07_SIT_TO_STAND.pkl
Open ./acc8vel8iter50/acc8vel8iter50_08_SIT_TO_LIE.pkl
Open ./acc8vel8iter50/acc8vel8iter50_09_LIE_TO_SIT.pkl
Open ./acc8vel8iter50/acc8vel8iter50_10_STAND_TO_LIE.pkl
Open ./acc8vel8iter50/acc8vel8iter50_11_LIE_TO_STAND.pkl
=================== Wrong Cases ===================
SITTING STANDING -1831.2377137911633 False
```



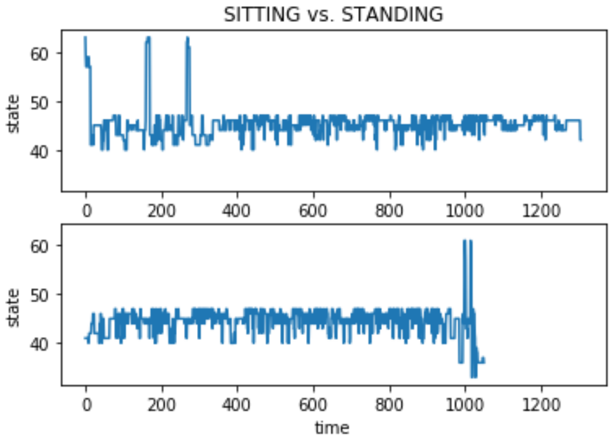
```
SITTING STANDING -1642.0398878264227 False
```



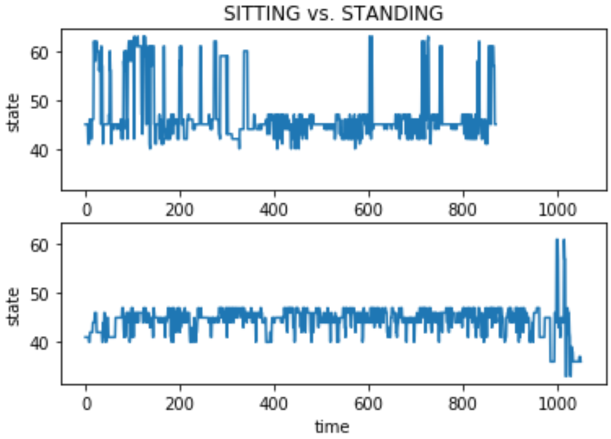
```
SITTING STANDING -1076.062357540761 False
```



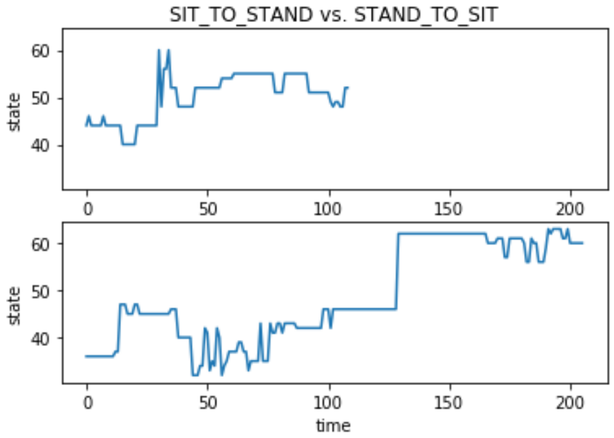
```
SITTING STANDING -1548.3217601864283 False
```



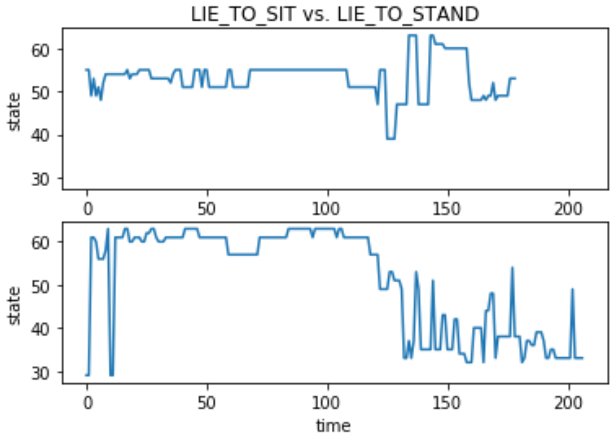
```
SITTING STANDING -1428.5890072836985 False
```



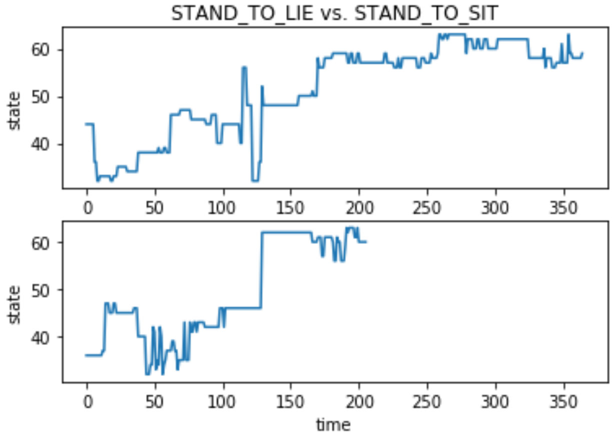
```
SIT_TO_STAND STAND_TO_SIT -105.8773443034097 False
```



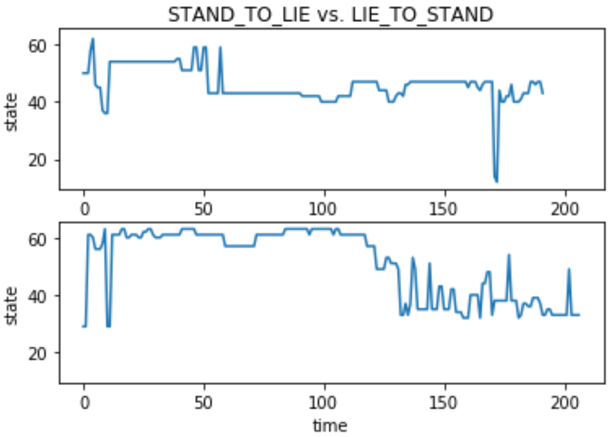
```
LIE_TO_SIT LIE_TO_STAND -212.67516426968723 False
```



```
STAND_TO_LIE STAND_TO_SIT -376.3769555622246 False
```



```
STAND_TO_LIE LIE_TO_STAND -249.59297042464544 False
```



```
(Wrong, Total) (9, 122)
Accuracy: 92.62295081967213%
```

```
In [ ]:
```