Michael O'Meara
ESE650 Project 3: Gesture Recognition
Due date: 2/25/2016 at 1:20pm

Goal:

1. Implement a Hidden Markov Model applied to gesture recognition based on the paper, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", written by Lawrence R. Rabiner
2. Encode IMU data into observation symbols based on K-means
3. Train the HMM for each motion gesture using Expectation-Maximization procedure
4. Evaluate results to improve initialization parameters

Assumptions:

1. IMU data represents unique motion gestures that can be discretized into identifiable categories.
2. K-means algorithm can adequately separate the 6 dimensional IMU data into observation symbols
3. Discretized sequence observations can be learned using an HMM

Strategy:

(1) First, I collected all the IMU data for both Training and Testing sets in order to run k-means on the entire dataset to segment the 6 dimensional data into a universal observation symbol dictionary. In this project, I found that 45 clusters worked well having tried a range between 7 to 200. These 45 clusters became the representation of my observation symbols.

(2) Then, I initialized λ with a simple left-right model, specifically I used values of $A_{i,i} = 1 - 1/(T_{otal\ observations}/N_{umber\ of\ states})$ and $A_{i+1,j} = 1 - A_{i,i}$. B_{ij} was initialized with the values of 1/M, where M is the number of emission and π was given the value of a 1xN vector equal to a one followed by N-1 zeros where N is the number of states in the model.

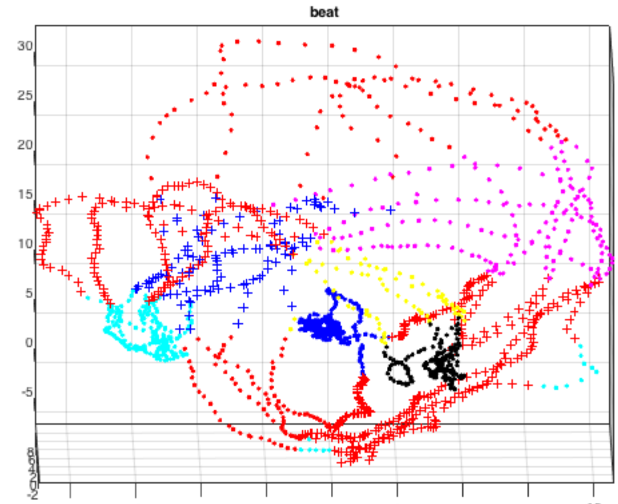
(3) Next, I trained the HMM with this initial λ on the Training set. This training set was composed of roughly 5 gesture examples per training file with 5 training files per gesture and a total of 6 gestures. The label of the gesture was provided via the file name. Training consisted of an expectation-maximization procedure using the forward-backward algorithm and then re-estimating values for A, B and π until the log-likelihood of $P(O|\lambda)$ converged.

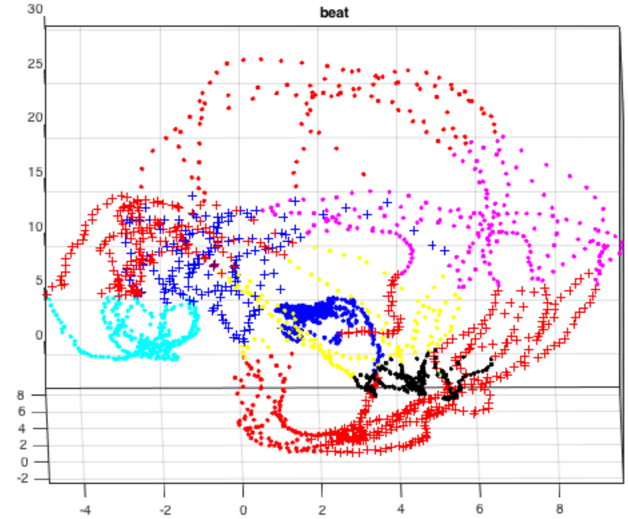
(4) After the model was trained I tested my model with the single instance gesture files to verify the model's performance, which was quite poor initially.

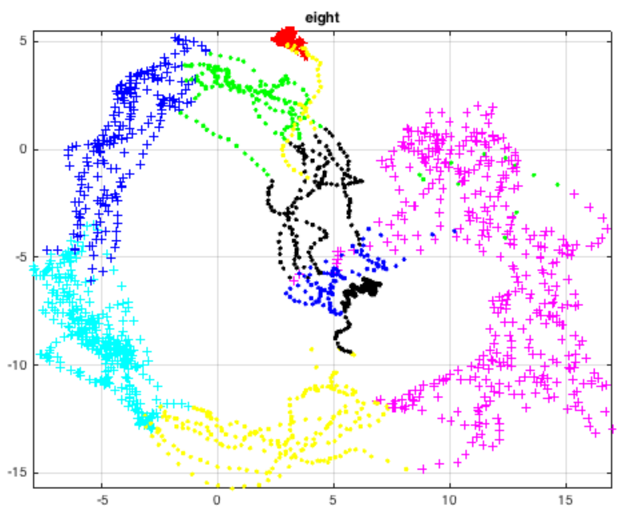
(5) In order to improve performance, I decided to re-examine the initial values for λ and switched to a slightly more connected version the A matrix where instead of zeros on the off diagonals I made the $A_{i,j} = 1 - A_{i,i}$ and $A_{i,i} = 1/N$. I also modified π = [0.5 1/N 1/N 1/N 1/N 1/N 1/N]. After these modifications and a few others to prevent both A and B matrices from having zero values I achieved 100% recognition of each gesture.

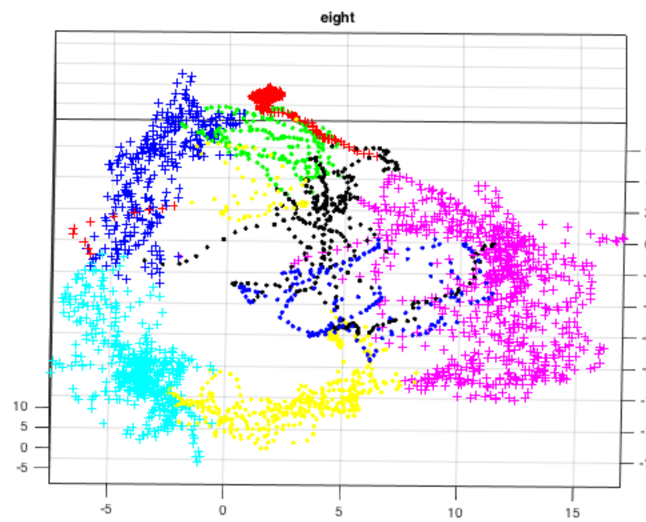
(6) Finally, I retrained the model holding out one of the example training gestures to see if I could identify a gesture that the model was trained to recognize. And what I found was the resulting log probabilities of an unrecognized gesture were much smaller. Recognized gestures had log-likelihoods in the range between -1000 to -5500, whereas, unrecognized gestures had values in the range of -5200 to -7000. And recognized gestures usually had values greater than 1000 log-likelihood points over an unrecognized gesture and was the ultimate determining factor.

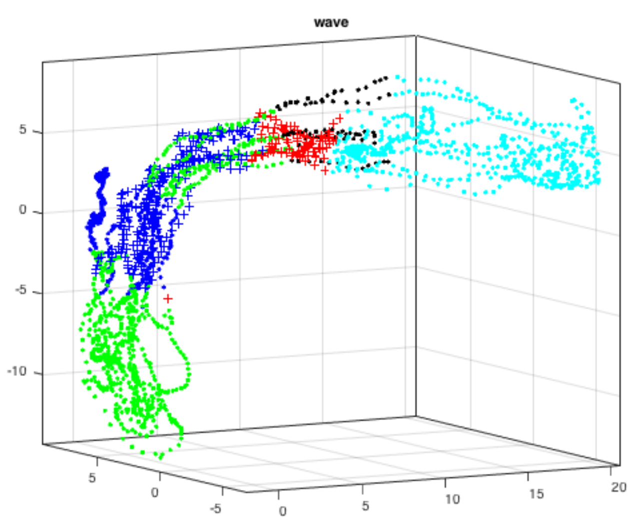
Understanding the data

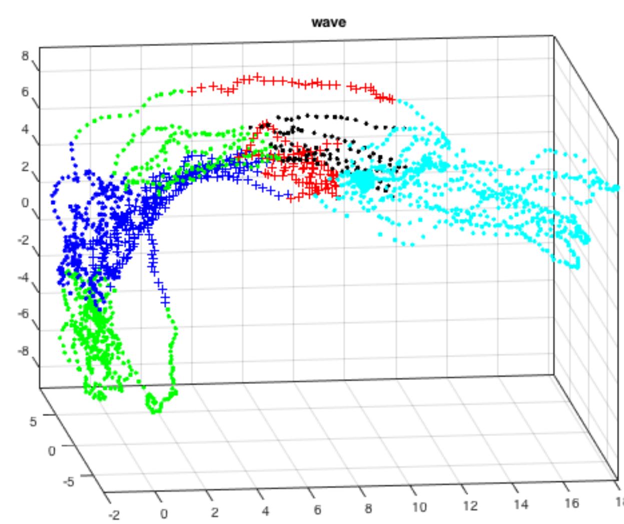












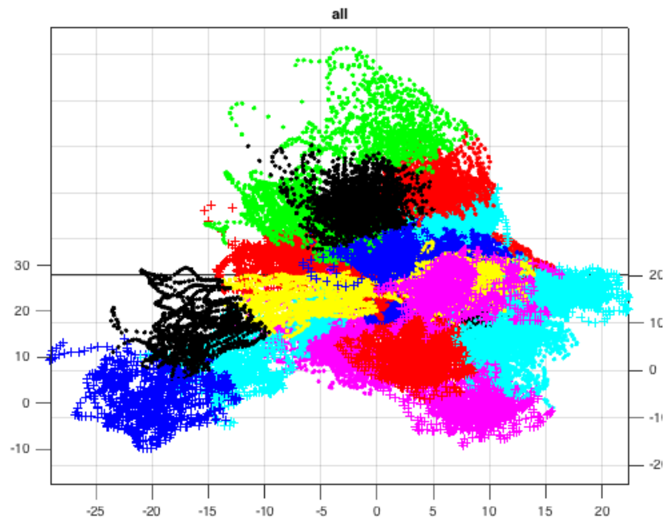
On the previous page are a few examples of the training data representing two of each “beat”, “eight” and “wave” gestures. Since the 6 dimensional data can't be graphed I am only showing the gyro data in these 3D plots.

As we can see each different gesture looks quite different from other gestures and so I felt confident the HMM would be able to discriminate between them.

In the beginning of this project I experimented with different approaches of dimensionality reduction (e.g. adding accelerometer$_{x,y,z}$ to gyro$_{x,y,z}$ then dividing by 2, FFT), but ultimately the raw data proved enough to achieve excellent results. I did not, however, use the duration data since including it resulted in poorer results.

K-means Observation Symbols Representation (full data set)

A couple challenges worth mentioning was the choice of clusters and the choice of the number of states. The number of clusters was chosen based on experimental results whereas the number of states was based on the number of gestures plus one to account for a gesture that could not be recognized (i.e. N = 7, where N is the number of states).



Testing on Single Instance Training Data

Testing gesture: beat3 guess #1: beat3 log-likelihood = -3766.924760 guess #2: beat3 log-likelihood = -4497.832409 guess #3: beat3 log-likelihood = -4750.758906	Testing gesture: circle guess #1: circle log-likelihood = -2944.745373 guess #2: circle log-likelihood = -3324.212276 guess #3: circle log-likelihood = -3689.075353	Testing gesture: inf guess #1: inf log-likelihood = -1043.536984 guess #2: inf log-likelihood = -3569.688786 guess #3: inf log-likelihood = -4237.417969
Testing gesture: beat3 guess #1: beat3 log-likelihood = -3892.187845 guess #2: beat3 log-likelihood = -4605.170398 guess #3: beat3 log-likelihood = -4869.261040	Testing gesture: circle guess #1: circle log-likelihood = -3431.485243 guess #2: circle log-likelihood = -3607.922649 guess #3: circle log-likelihood = -3891.333690	Testing gesture: inf guess #1: inf log-likelihood = -1399.780131 guess #2: inf log-likelihood = -4299.815403 guess #3: inf log-likelihood = -4938.748914
Testing gesture: beat4 guess #1: beat4 log-likelihood = -4726.176653 guess #2: beat3 log-likelihood = -5673.898230 guess #3: beat3 log-likelihood = -6393.749987	Testing gesture: eight guess #1: eight log-likelihood = -5379.544626 guess #2: eight log-likelihood = -5676.230206 guess #3: eight log-likelihood = -5844.808501	Testing gesture: wave guess #1: wave log-likelihood = -2959.171214 guess #2: wave log-likelihood = -5059.317341 guess #3: inf log-likelihood = -6619.443383
Testing gesture: beat4 guess #1: beat4 log-likelihood = -4598.806372 guess #2: beat3 log-likelihood = -5756.564040 guess #3: beat4 log-likelihood = -6378.098532	Testing gesture: eight guess #1: eight log-likelihood = -5078.436176 guess #2: eight log-likelihood = -5187.627274 guess #3: eight log-likelihood = -5398.098077	Testing gesture: wave guess #1: wave log-likelihood = -1830.097469 guess #2: wave log-likelihood = -4209.226536 guess #3: wave log-likelihood = -4887.344214

Final initial λ parameters

A$_{7x7}$ =

0.8571	0.1429	0.1429	0.1429	0.1429	0.1429	0.1429
0.1429	0.8571	0.1429	0.1429	0.1429	0.1429	0.1429
0.1429	0.1429	0.8571	0.1429	0.1429	0.1429	0.1429
0.1429	0.1429	0.1429	0.8571	0.1429	0.1429	0.1429
0.1429	0.1429	0.1429	0.1429	0.8571	0.1429	0.1429
0.1429	0.1429	0.1429	0.1429	0.1429	0.8571	0.1429
0.1429	0.1429	0.1429	0.1429	0.1429	0.1429	0.8571

π$_{1x7}$ = [0.5000 0.1429 0.1429 0.1429 0.1429 0.1429 0.1429]

I did try normalized version of the above initializations and they didn't work as well.

Test Results

Testing gesture: test1 guess #1: wave log-likelihood = -1515.396046 guess #2: wave log-likelihood = -3793.539551 guess #3: wave log-likelihood = -4009.699501 Testing gesture: test2 guess #1: beat4 log-likelihood = -2415.957074 guess #2: beat3 log-likelihood = -4465.182229 guess #3: beat3 log-likelihood = -5250.961563 Testing gesture: test3 guess #1: inf log-likelihood = -751.439986 guess #2: inf log-likelihood = -3565.385297 guess #3: inf log-likelihood = -4317.557212 Testing gesture: test4 guess #1: beat4 log-likelihood = -3092.426794 guess #2: beat4 log-likelihood = -5068.025264 guess #3: beat3 log-likelihood = -5172.046285	Testing gesture: test5 guess #1: circle log-likelihood = -2851.018042 guess #2: circle log-likelihood = -3665.658189 guess #3: circle log-likelihood = -3696.273926 Testing gesture: test6 guess #1: inf log-likelihood = -933.930514 guess #2: inf log-likelihood = -3664.066745 guess #3: inf log-likelihood = -4454.198139 Testing gesture: test7 guess #1: eight log-likelihood = -1458.674876 guess #2: eight log-likelihood = -4896.761034 guess #3: eight log-likelihood = -5203.317076 Testing gesture: test8 guess #1: beat4 log-likelihood = -2389.167020 guess #2: beat3 log-likelihood = -4456.883778 guess #3: beat3 log-likelihood = -5239.975002

As we can see from the results above, on the Test Set, log-likelihood values greater than -2500 show a fairly high confidence of the particular gesture. For example, test2's best guess is “beat4” with a log-likelihood value of -2416, but test4's best guess is “beat4” with a log-likelihood value of -3092, which indicates that it's not as confident in this guess. Another indicator of the confidence in the guess is when looking at the difference between the guesses #1 through #3. For most of the test gestures there's a large difference showing high confidence in the guess. The exception being test4 where the log-likelihood values are closer together showing an overall lower confidence in that guess. And finally, when the top guesses don't match then there's obviously lower confidence in the guess. This is especially true of the beat gestures since they're all fairly similar.

Lastly, I found my results varied depending greatly on how kmeans segmented the gesture space. Sometimes guessing “beat3”, sometimes “beat4” on the same test. As I proposed in class a solution to this issue might be to use another technique to segment the space, like for example, an octree or kd-tree.