

Assignment 2

Neeraj Sujan
Student Id- 03656452
Technische Universität München

August 23, 2015

1 Learning dataset using Gaussian mixture model

This is time for all good men to come to the aid of their party!

Solution The learned GMM parameters are given as follows.
The estimated mean value is

-0.0432	0.0446
-0.0147	-0.0796
0.0262	0.0617
-0.0194	-0.0166

The prior value is given as

0.2400
0.2011
0.2617
0.2972

The covariance matrix is given as

Covariance Matrix		
1	0.000174789634891584 0.000261535184754085	0.000261535184754084 0.000397543592937929
2	0.000394394847071063 0.000216642829136404	0.000216642829136404 0.000127573426379305
3	0.00108304683924854 -0.000424358139781523	-0.000424358139781523 0.000243120264604005
4	0.000743719567290450 -0.000591678530894728	-0.000591678530894728 0.000610265143134276

2 Human gesture recognition using hidden Markov model

Solution The log-likelihood for the test set is given as follows

log-likelihood
-511.406874422073
-570.669709935248
-387.916688342244
-427.306925139780
-437.598861101383
-426.178435051455
-473.303146894623
-400.287976276701
-377.177581003617
-401.061415558705

Since log-likelihood is less than -120, all points are classified as test-data.

3 Learning gait pattern for a humanoid robot using Reinforcement Learning

In this section we describe the results.

For Policy iteration and Q-learning, I have used the same reward matrix. The reward matrix is as follows

2	-2	2	-2
-2	2	-2	-2
2	-2	-2	-2
-2	-2	2	-2
-2	-2	-2	2
2	-2	2	-2
2	-2	2	-2
-2	2	-2	-2
-2	2	2	-2
2	-2	2	-2
2	2	2	2
-2	2	-2	2
2	-2	-2	-2
-2	-2	-2	2
2	2	-2	2
-2	2	-2	2

I have used 0.9 as the value of discount factor. If I increase the value of the discount factor, the algorithm does not converge whereas for lower values of the discount factor, the algorithm takes fewer steps to converge.

Approximately 11-14 steps are required for the algorithm to converge.



Figure 1: initial state = 3



Figure 2: initial state = 10

Q Learning The value of alpha is 0.1 and the value of epsilon is varied between 0.1 to 0.9. I have set it to be 0.1.

If a pure greedy policy is used, then the algorithm does not converge. For epsilon = 0.1 approximately 80 steps are required for the algorithm to converge.



Figure 3: initial state = 5



Figure 4: initial state = 12