

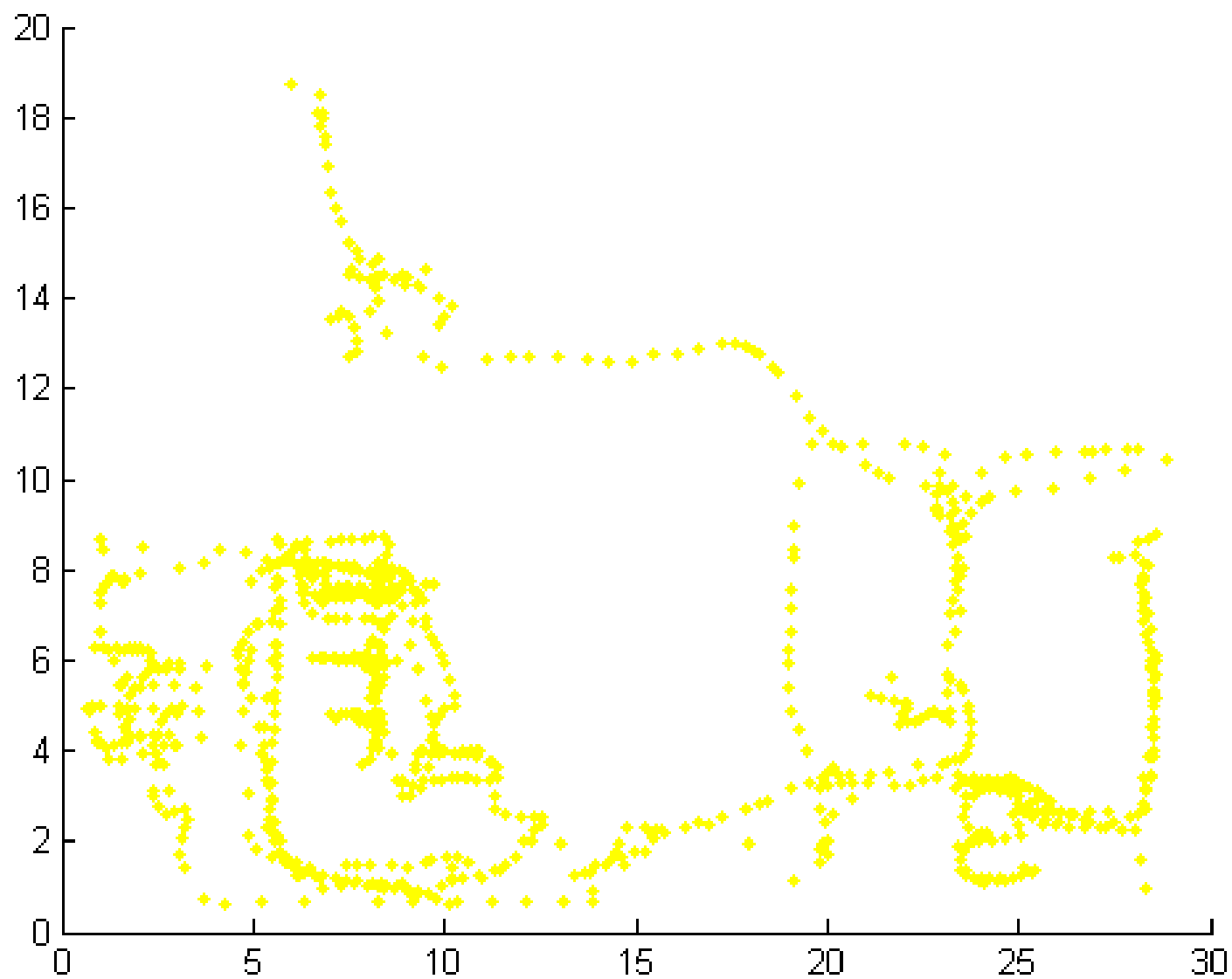
Unsupervised learning to detect teams from individual player trajectories

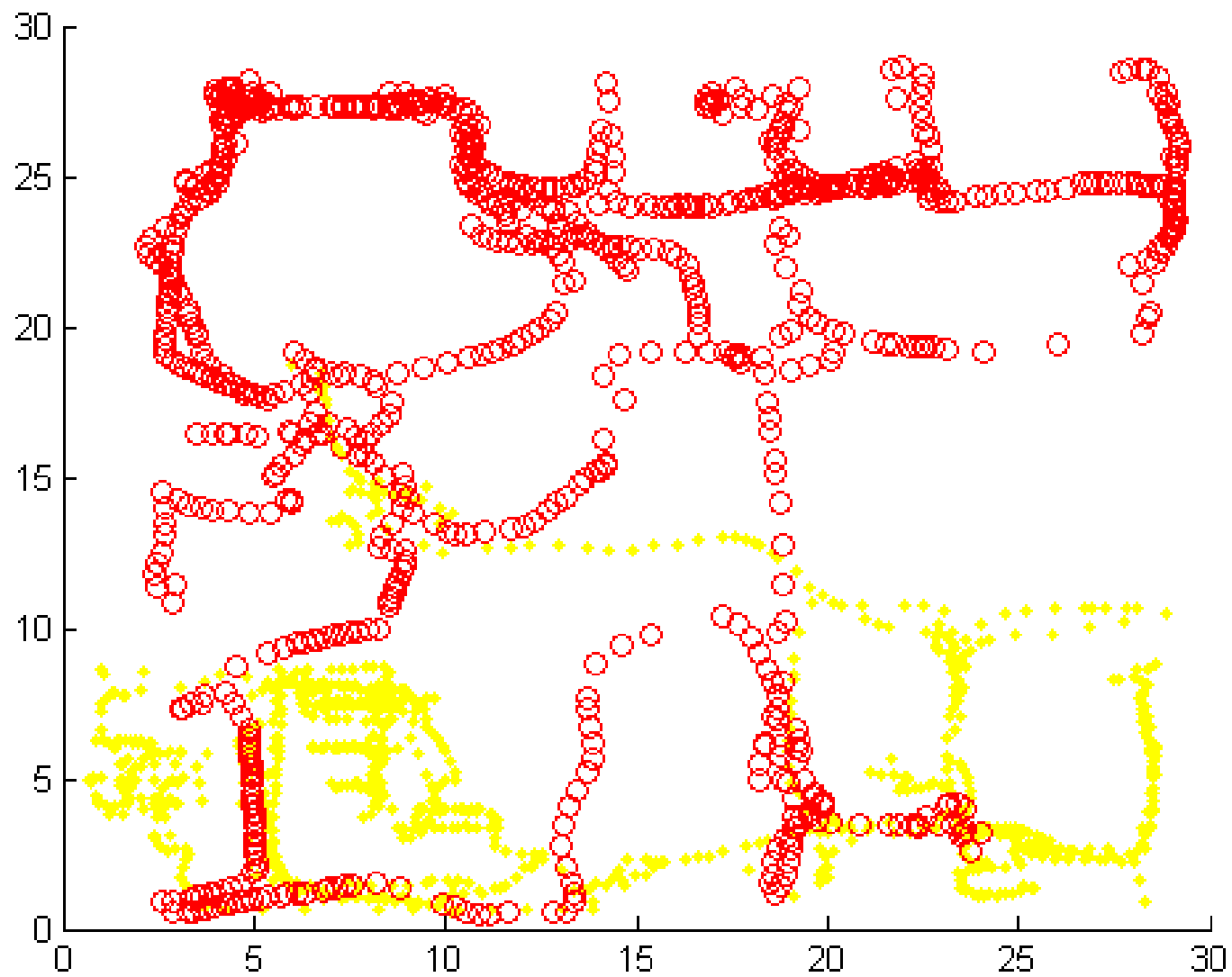
Comp – 652: Machine learning

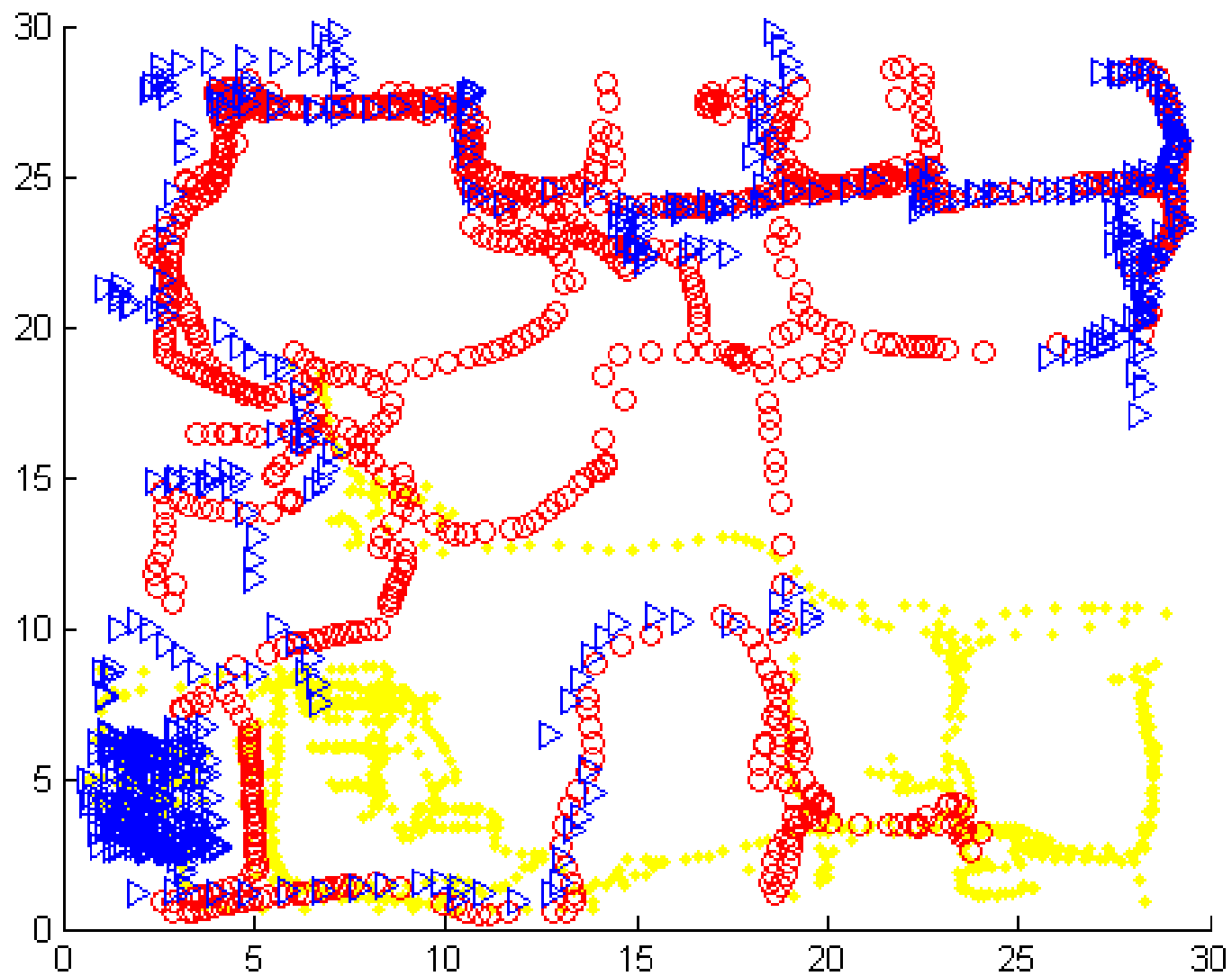
Professor: Doina Precup
Student: Gayane Petrosyan

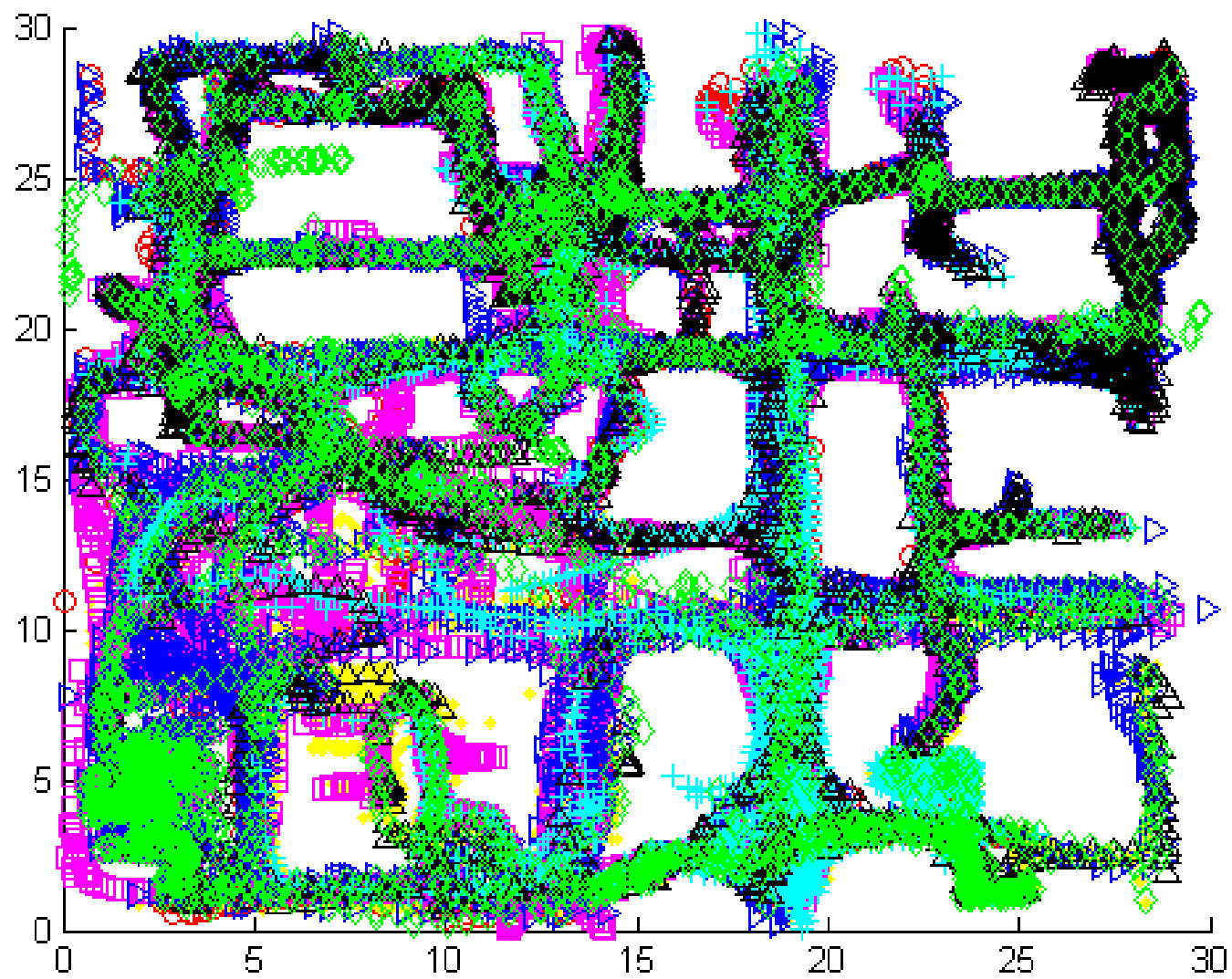
Orbius game and Challenge

- 2d multiplayer game
- players move around and collect some balls
- 3 full game logs
- 28 players and 7 teams



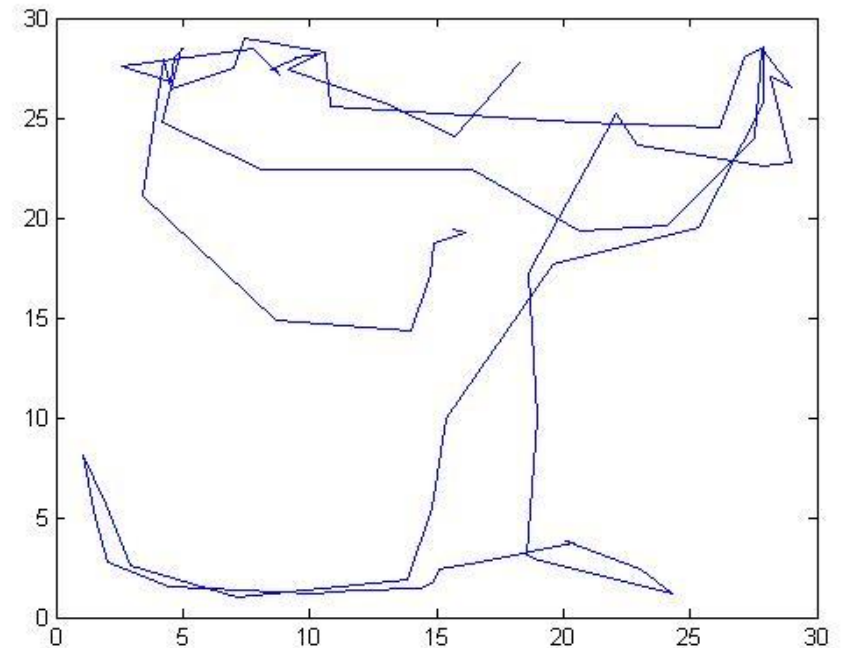
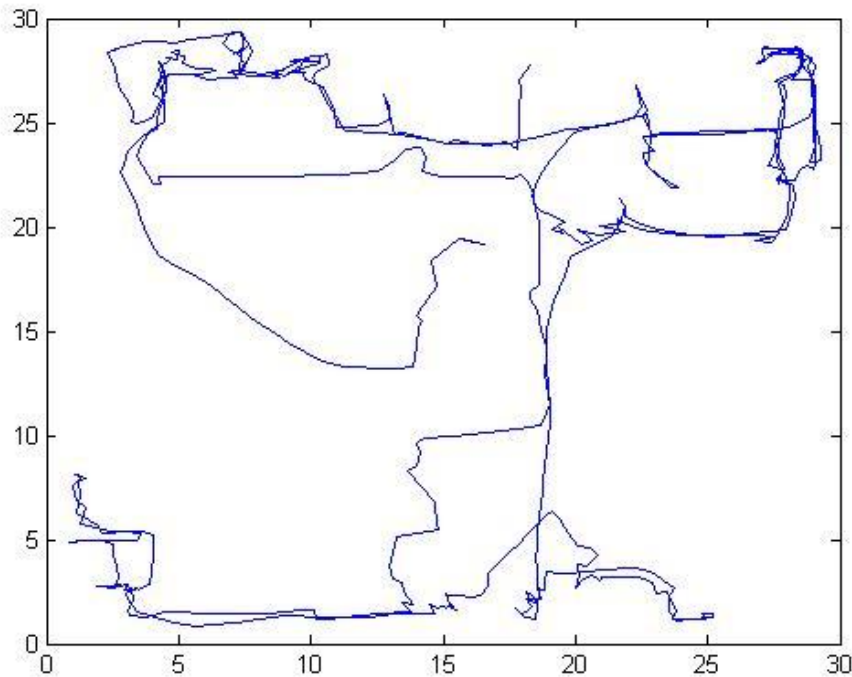






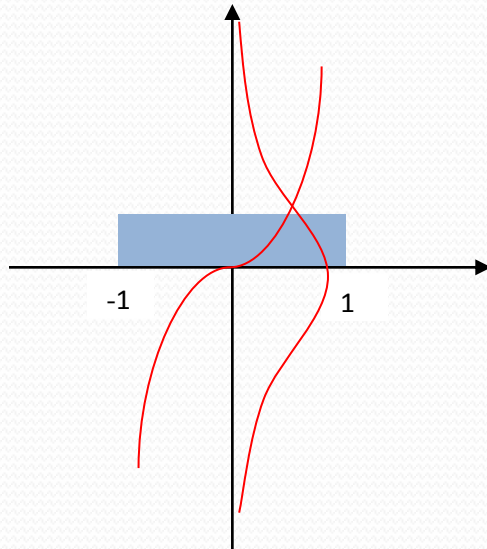
Feature selection and preprocessing

- grouping by intervals of time



Feature selection and preprocessing (2)

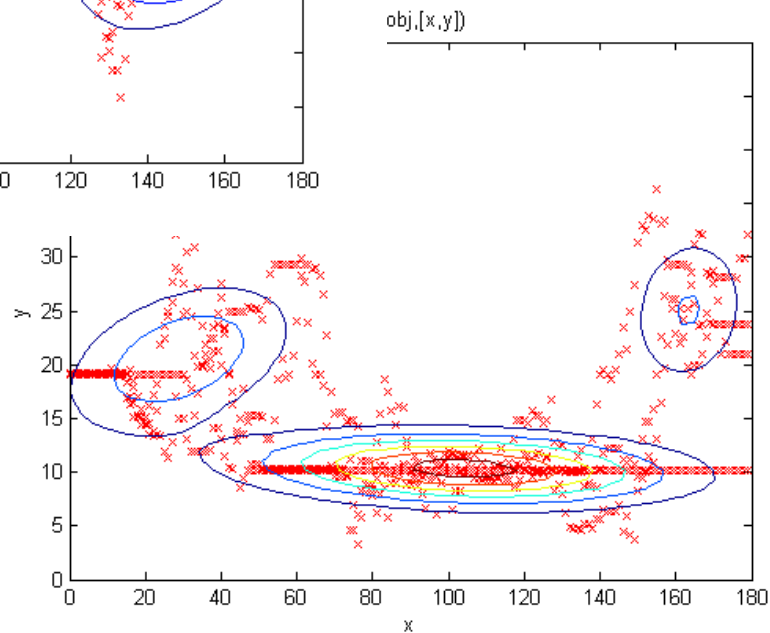
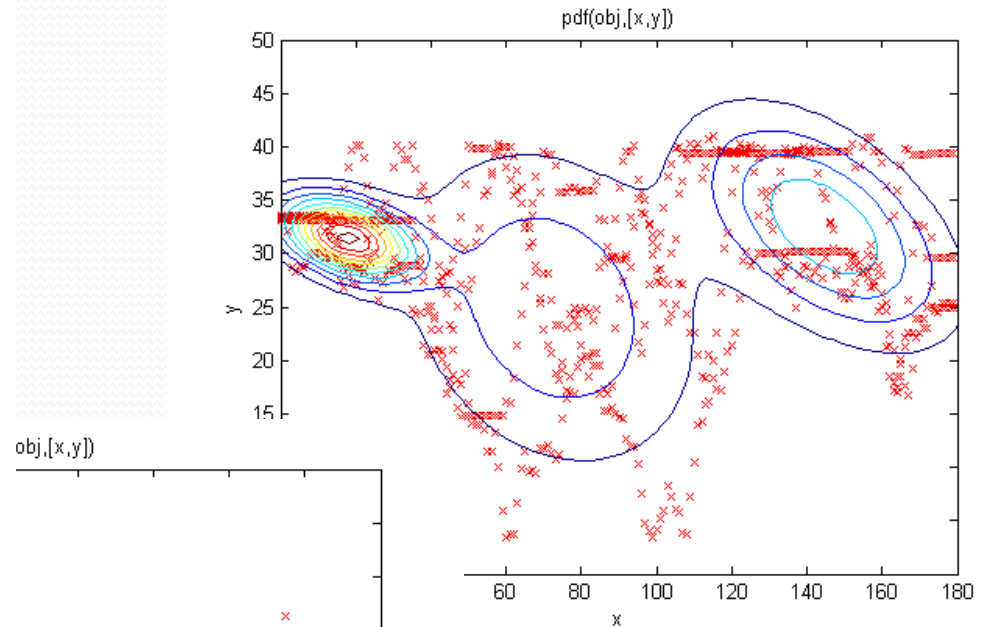
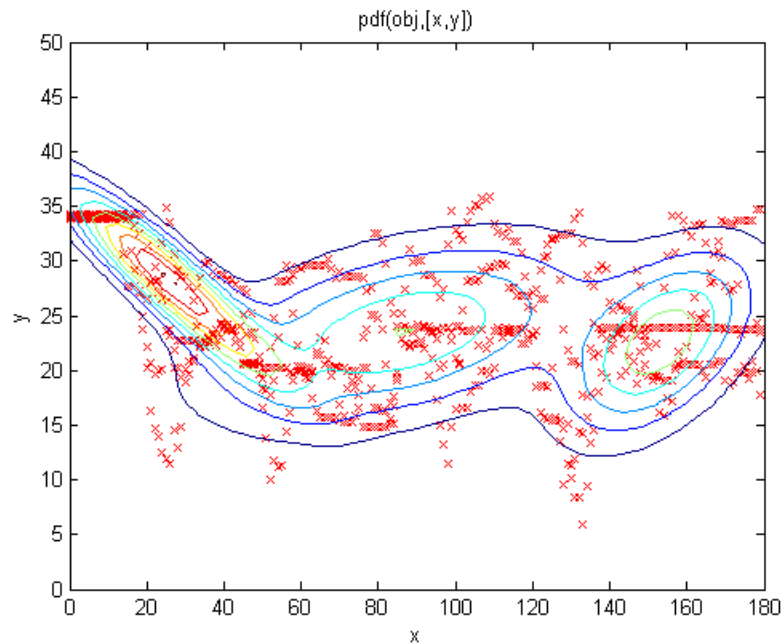
- Calculating a distance from origin
- Making uniform distribution Gaussian(inverse error function)



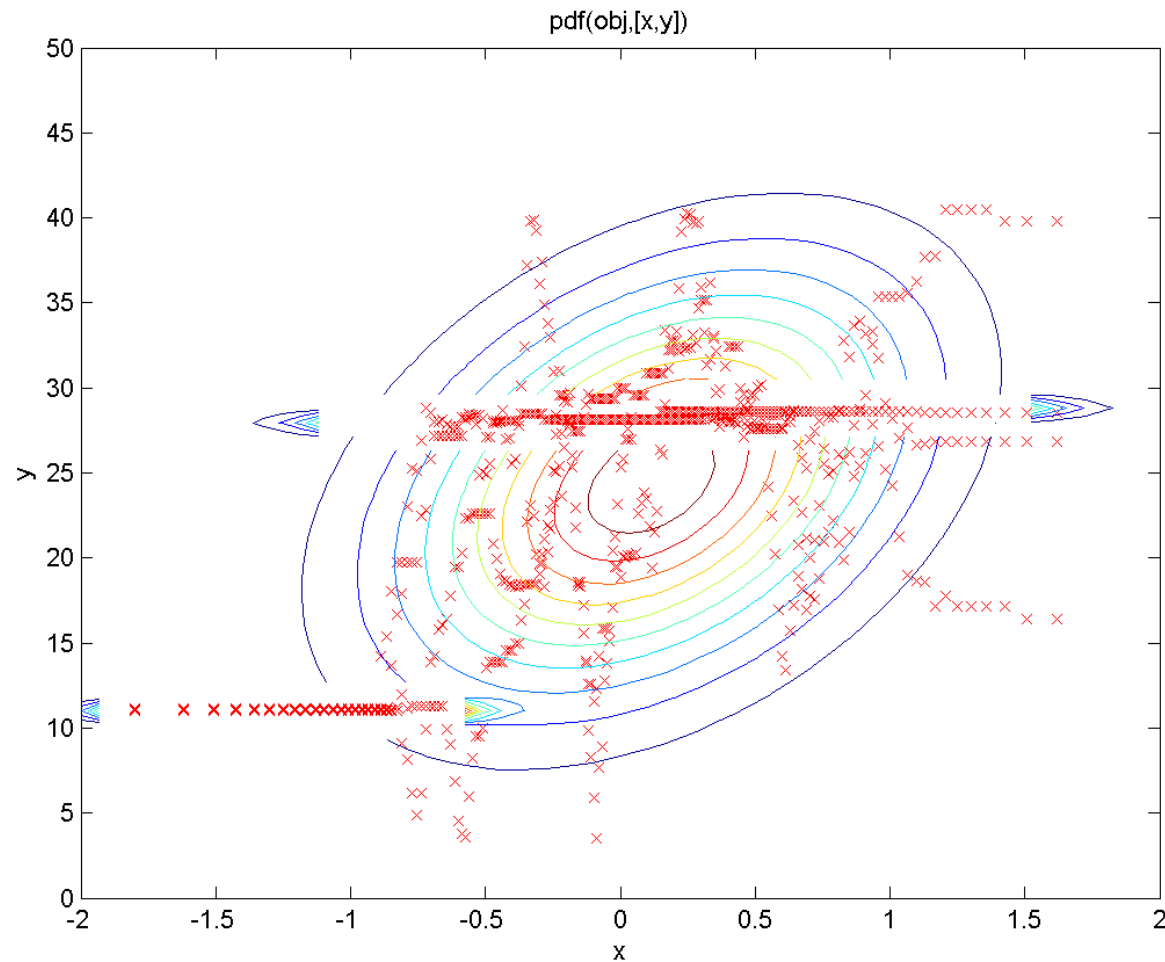
General approach

- Mixture of Gaussian. Find parameters with EM
- For each player find a probability of being in each group
- For each team picking most probable players

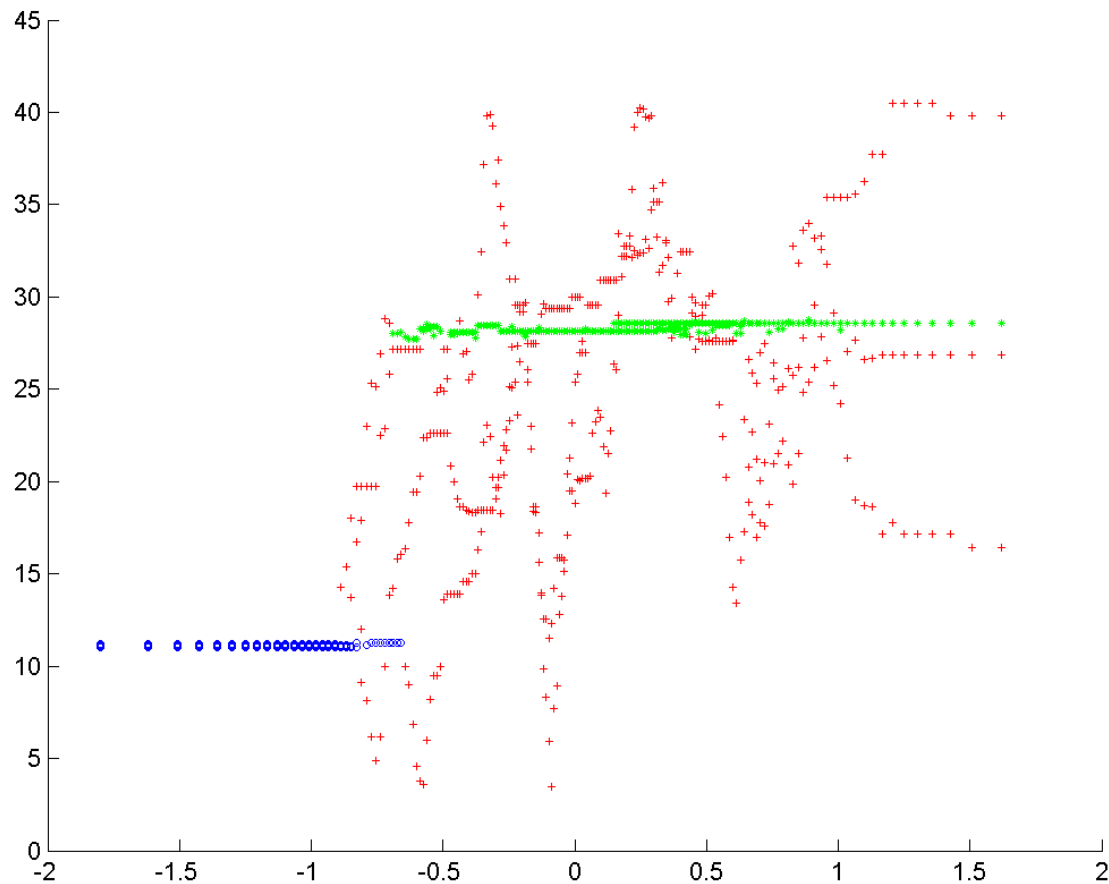
Mixture of Gaussian



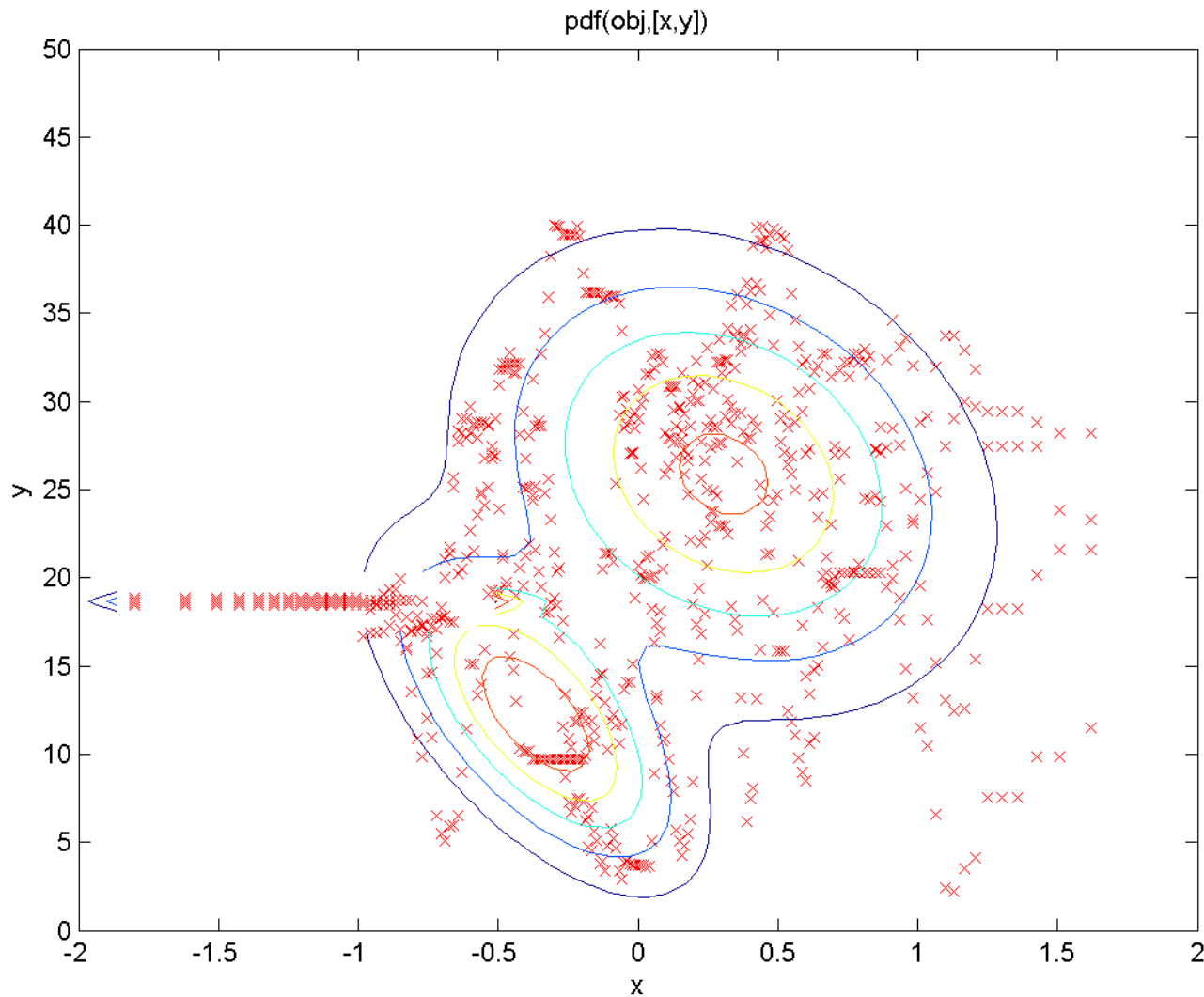
Mixture of Gaussian (parameter estimation)



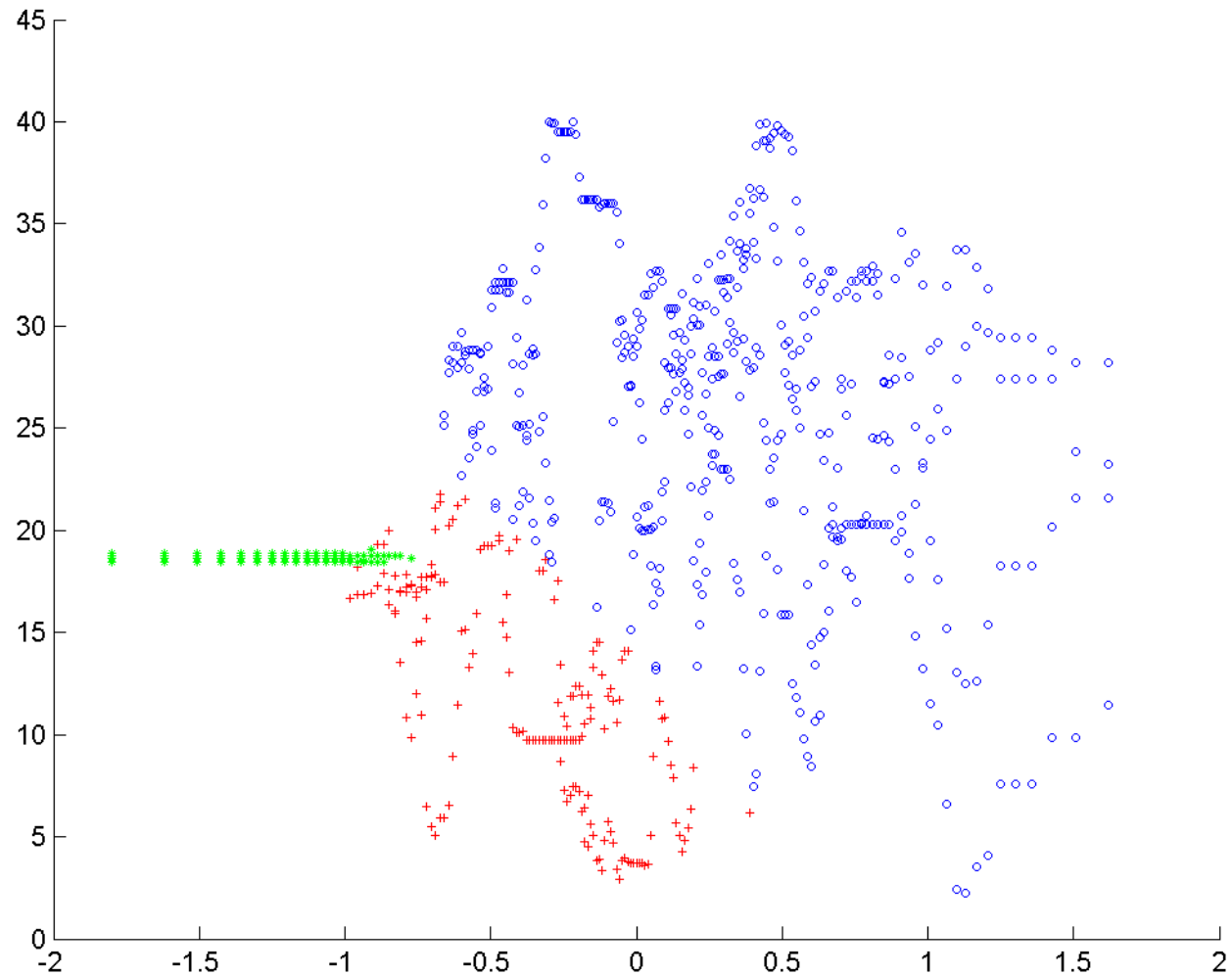
Mixture of Gaussian (clustering)



Mixture of Gaussian (parameter estimation)



Mixture of Gaussian (clustering)



Grouping

- For each player calculate log likelihood of being in particular team

$$\arg \max_c (\log P(x_i/y_i = k; \mu_{kc}, \Sigma_{kc}) + \log P(y_i = k, \phi_{kc}))$$

- Repeatedly
 - Assign each team most probable players
 - Assign players for their highest probability team.



Results... could be better

Training vs. Testing

2	4	1	3
5	7	8	6
9	12	10	25
16	26	11	15
18	19	20	14
24	21	23	22
13	17	28	27

Error percentage: 0.214286

2	1	3	10
14	8	12	5
11	6	9	16
4	13	26	21
18	17	15	28
19	7	22	24
27	20	25	23

Error percentage: 0.535714

Future work directions

- More group examples
- Elimination of similar groups



Questions ?